Giampiero Mastinu
Massimiliano Gobbi
Carlo Miano

Optimal Design of Complex Mechanical Systems

With Applications to Vehicle Engineering

Springer

Optimal Design of Complex Mechanical Systems

G. Mastinu M. Gobbi C. Miano

Optimal Design of Complex Mechanical Systems

With Applications to Vehicle Engineering

With 161 Figures 62 Tables

Springer

Professor Dr-Ing. Giampiero Mastinu
Professor Dr-Ing. Massimiliano Gobbi
Department of Mechanical Engineering
Politecnico di Milano (Technical University)
Via La Masa, 34
20156 Milan, Italy

Dr-Ing. Carlo Miano
formerly with:
– Politecnico di Milano
Department of Mechanical Engineering, Italy
– FORD (FFA), Aachen, Germany
now with:
– Ferrari Automobili S.p.A., Maranello, Italy

Library of Congress Control Number: 2006925849

ISBN-10 3-540-34354-7 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-34354-7 Springer Berlin Heidelberg New York

Typesetting: by the authors and techbooks using a Springer LaTeX macro package
Cover design: *design & production* GmbH, Heidelberg

Printed on acid-free paper SPIN: 11607786 89/techbooks 5 4 3 2 1 0

To my beloved aunt and teacher, Tata. Giampiero
To my family. Max
To Carolina and my family. Carlo

Preface

The book is devoted both to researchers wishing to acquire basic or advanced knowledge on the optimisation of complex mechanical systems (with special reference to multi-objective programming) and to engineers specialist in the field of vehicle design. In the book, reference will be made exclusively to those (complex) mechanical systems which can be described by means of mathematical models. The optimal design of these complex systems implies the proper tuning of many design variables (up to 50 or more) in order to find a preferred compromise among conflicting system's performances. The number of these performances is also usually very high (up to 20 or more).

In the first part of the book, an informative tutorial (Chap. 1) is produced to introduce to the reader the main issues on the optimisation of complex mechanical systems. The subject is explained plainly with the aim to present the key definitions and the basic considerations on the topic. Additionally, the main techniques to solve optimisation problems are briefly presented (without resorting to involved mathematics) to make the reader aware of the actual problem-solving capabilities in the field. All the topics which are mentioned in the informative tutorial (Chap. 1) are dealt with in mathematical form in Chaps. 2–4.

In the second part of the book, all the mathematical theories introduced in the first part are applied to solve a number of optimisation problems, many of which refer to ground vehicles.

Authors do think that the design or (which is the same) the optimisation of complex mechanical systems is only at its very beginning. Actually, the power of computers is just now becoming nearly sufficient to attempt the solution of such optimisation problems. Possibly, in the future new knowledge will be acquired on the dynamical behaviour of complex systems and consequently new algorithms for optimising them will be developed. Authors envisage that the optimisation of complex mechanical systems will be continuously updated in

the near future. The book will remain anyway at least informative on the basic issues on the topic, both with reference to the theory, and with reference to a number of presented optimisation examples.

Milan
December 2005

G. Mastinu
M. Gobbi
C. Miano

Acknowledgements

First and foremost, we wish to thank Professor Carlo Doniselli for having stimulated, with his valuable advice and constructive criticism, our interest in optimisation theory.

We wish to express our gratitude to Paolo Guarneri, Francesco Levi and Gianpiero Rocca for their effort in reading the manuscript and reviewing it.

Milan
December 2005

G. Mastinu
M. Gobbi
C. Miano

Contents

List of Figures

List of Tables

Acronyms

1S-PSD	one-slope PSD (Eq. (5.7), Chap. 5)
2S-PSD	two-slope PSD (Eq. (5.8), Chap. 5)
ANN	artificial neural network
BP	back-propagating
DM	decision maker
DACE	design and analysis of computer experiments
DOF	degree of freedom
GA	genetic algorithm
GCV	generalized cross-validation
GSA	global sensitivity analysis
KKT	Karush–Kuhn–Tucker
LCC	life cycle cost
LDN	low discrepancy nets
LDS	low discrepancy sequences
MCO	multi-criteria optimisation
MCDA	multi-criteria decision aid
MCDM	multi-criteria decision making
MLPNN	multi-layer perceptron neural network
MOGA	multi-objective genetic algorithm
MOP	multi-objective programming
MOO	multi-objective optimisation
NBI	normal boundary intersection
NLP	non-linear programming
NN	neural network
OR	operational research
PCC	Pearson correlation coefficient
PO	Pareto-optimal
QMM	quasi-Monte Carlo method
RBF	radial basis function
RBFNN	radial basis function neural network

RSM	response surface methodologies
RMS	root mean-square
SQP	sequential quadratic programming
SRCA	Spearman rank correlation analysis
SSE	Sum of squared errors
TSS	Total squared sum

List of Symbols

$\mathcal{C}$	objective function space
c_i	probability of selecting the individual i for genetic algorithms
$\mathbf{d}$	small increment of the design variables
D	total model variance using Sobol method
D_N	discrepancy of a set of N points
D_N^*	discrepancy of a set of N points (alternative definition)
$d(\mathcal{H})$	the distance between the first and the last specific digit on the string in genetic algorithms
d_N	dispersion of the sample of size N
$\mathcal{F}$	design variable feasible domain
$\mathbf{f}$	objective function vector
$\mathbf{g}$	constraint vector
$H(\mathbf{x})$	Hessian matrix
$\mathcal{O}^{n_{dv}}$	n_{dv}-dimensional unit cube design variable domain of definition
N	number of function evaluations
N_r	highest rank of the population considered in genetic algorithms
N_t	size of the training set
N_{si}	population of size i in genetic algorithms
n_b	number of bits
n_{cdv}	number of combinations of design variables
n_{ec}	number of equality constraints
n_c	number of constraint
n_{dv}	number of design variables
n_l	total number of layers in a neural network
n_{of}	number of objective functions
n_v	number of interval divisions
OA	orthogonal array
$O(\mathcal{H})$	order of a schema in genetic algorithms
$\mathcal{P}$	generic set of Pareto-optimal points

P_c	crossover rate
P_m	mutation rate
Q	quadratic form of the Hessian ($\mathbf{d}^T H(\mathbf{x})\mathbf{d}$)
r_s	Spearman correlation indices
R	coefficient of determination
$\Re^n$	n-dimensional set of real numbers
s	Laplace variable
S	Sobol sensitivity index
S_{SE}	sum of squared errors
$\hat{\mathbf{T}}$	transition matrix
T_{SS}	total square sum
T_S	total Sobol sensitivity index
t_s	simulation time to perform a single simulation
t_t	total simulation time
U	utility function
$\mathbf{w}$	weight vector
$\mathbf{W}$	matrix of interconnection weights
$\mathbf{x}$	design variable vector
$\mathcal{X}$	domain of definition of design variables
$\mathbf{y}$	objective function vector
$\boldsymbol{\beta}$	free parameters of a global approximation method/regression method
$\boldsymbol{\varepsilon}$	constraint vector (in constraints method)
η	safety coefficient ($\geq$1)
$\boldsymbol{\eta}$	Lagrange multipliers of constraints (in KKT condition)
Φ	penalty function
Λ_{ij}	trade-off rate
$\boldsymbol{\lambda}$	Lagrange multipliers of objective functions (in KKT condition)
λ_r	regularisation parameter
Σ_x	standard deviation of the variable x

Notations

$\mathbf{x}$	vector (small bold character)
x_{ij}	i=component of a vector for ($i = 1, \ldots, n_{dv}$); j=number of sample for ($j = 1, \ldots, N$)
$\mathbf{H}$	matrix (capital bold character)
h_{ij}	i=row of a matrix; j=column of a matrix
$\mathcal{C}$	set (capital cursive character)
$\mathrm{Cov}(x, y)$	covariance of the variables x and y
$\mathrm{Var}(x)$	variance of x
$\nabla f_i(\mathbf{x})$	gradient of f_i estimated at $\mathbf{x}$
$\hat{}$	estimated quantity

$\bar{x}$	mean of x
$\tilde{}$	goal vector (e.g. $\tilde{\mathbf{y}}$ goal vector of objective function vector $\mathbf{y}$)

Operators

^{T}	transpose
$\in$	included
$\cap$	intersection
$\cup$	union
:	such that
$\wedge$	logical AND

Chapter 1

b	beam cross-section width (m)
E	material modulus of elasticity (Young's modulus) (MPa)
F	force applied at the cantilever free end (N)
F_{cr}	cantilever critical load (N)
G	material modulus of tangential elasticity (MPa)
h	beam cross-section height (m)
J	flexural moment of inertia of the section (m^4)
k_1	cantilever beam coefficient for the calculation of the critical load
k_2	cantilever beam coefficient for the calculation of the critical load
l	cantilever length (m)
m	cantilever mass (kg)
y	deflection at the cantilever free end due to load (m)
ρ	material density (kg/m^3)
σ_E	material yield stress (MPa)
σ_{adm}	admissible stress at the cantilever fixed end (MPa)
σ_{max}	maximum stress located at the top of the cross-section at the fixed end of the cantilever (MPa)

Chapter 3

h	cylinder height (m)
r	cylinder radius (m)

Chapter 5

a	break wave number (2S-PSD) (rad/m)
A_b	road irregularity parameter (1S-PSD) (m)
A_v	road irregularity parameter (2S-PSD) (m^2)
c_o	$c_o = A_v a v$ (m^2 rad/s)
c_{rv}	$c_{rv} = \frac{1}{2} c_o$ (m^2 rad/s)
G_1	active suspension gain G_1 (N/m)
G_2	active suspension gain G_2 (N s/m)
H_1	transfer function between ξ and $\ddot{z_2}$ ($1/s^2$)
H_2	transfer function between ξ and F_{z1} (N/m)
H_3	transfer function between ξ and $z_2 - z_1$ (1)
j	$j = \sqrt{-1}$
k_1	tyre radial stiffness (N/m)
k_2	suspension stiffness (N/m)
m_1	unsprung mass (kg)
m_2	sprung mass (kg)
P_{SD}	power spectral density (PSD)
r_2	suspension damping (N s/m)
v	vehicle speed (m/s)
z_1	mass m_1 absolute vertical displacement (m)
z_2	mass m_2 absolute vertical displacement (m)
σ_{F_z}	standard deviation of the road/wheel vertical force (road holding) (N)
$\sigma_{\ddot{z}_2}$	standard deviation of the body acceleration (discomfort) (m/s^2)
$\sigma_{z_2 - z_1}$	standard deviation of the suspension stroke (working space) (m)
ξ	imposed displacement (m)
ω	circular frequency (rad/s)
ω_c	reference circular frequency $\omega_c = av$ (2S-PSD) (rad/s)

Subscript

r	reference value

Chapter 6

a_c	centripetal acceleration (m/s^2)
$C_{\alpha F/R}$	front (F)/rear (R) cornering stiffness ($(\partial F_y / \partial \alpha)_{\alpha=0}$) (N/rad)
CC	centre of curvature of centre of gravity trajectory
$F_{yF/R}$	front (F)/rear (R) axle lateral force (N)

$F_{yF/Rl}$	front (F)/rear (R) axle limit lateral force (N)
$F_{zF/R}$	front (F)/rear (R) axle vertical load (N)
G	position of the vehicle centre of gravity
g	acceleration of gravity (m/s^2)
J_z	inertia moment along the vertical axis (kg m^2)
K_β	sideslip angle gain
$K_{\dot\psi}$	yaw velocity gain
$l_{F/R}$	relative distances of G from front (F)/rear (R) axles (m)
l	wheelbase (m)
m	total mass of the vehicle (driver included) (kg)
$O_{\dot\psi}$	yaw velocity overshoot
$T_{\dot\psi}$	yaw velocity peak response time (s)
T_{z1}	time constant (s)
T_{z2}	time constant (s)
v	vehicle speed module (m/s)
v_{cr}	critical vehicle speed at which the vehicle motion becomes unstable (m/s)
$\alpha_{F/R}$	lateral slip angles of front (F)/rear (R) axles (rad)
$\alpha_{Fl/Rl}$	limit lateral slip angles of front (F)/rear (R) axles (rad)
β	sideslip angle at G (rad)
γ	boundary curve of the Pareto-optimal domain
δ	steering angle (rad)
δ_{steady}	steering angle input amplitude (rad)
$\tilde\delta$	steering angle input amplitude defined by Eq. (6.28) (rad)
κ	curvature of centre of gravity trajectory (1/m)
$\mu_{F/R}$	front (F)/rear (R) tyre–road friction coefficient
ν	cut off frequency (1/s)
ψ	yaw angle (rad)
ρ	radius of curvature of centre of gravity trajectory (m)
ξ	traveled distance of the centre of gravity (m)
ς	damping factor (1/s)

Chapter 7

a_c	centripetal acceleration at vehicle centre of gravity (m/s^2)
$C_{k,F/R}$	Pacejka's magic formulæ *coefficient (front/rear)*
F_d	force transmitted by shock absorber (N)
F_y	tyre lateral force (N)
$F_{y,stat}$	tyre lateral force (steady-state condition) (N)
F_z	tyre vertical load (N)
$k_{s,F/R}$	suspension spring stiffness (front/rear) (N/m)
$k_{r,F/R}$	anti-roll bar stiffness (front/rear) (N/m)
P	required engine power (W)

$p_{1,F/R}$	maximum cornering stiffness as a function of vertical load (front/rear) (N/rad)
$p_{2,F/R}$	vertical load corresponding to maximum cornering stiffness (front/rear) (N)
p_3	Pacejka's Magic Formulæ *coefficient (1/rad)*
$r_{b,h}$	bump damping at high speed $v_d \leq v_{0,b}$ (N s/m)
$r_{b,l,F/R}$	Bump damping at low speed $v_d > v_{0,b}$, $v_d \leq 0$ (front/rear) (N s/m)
$r_{r,h}$	rebound damping at high speed $v_d > v_{0,r}$ (N s/m)
$r_{r,l,F/R}$	rebound damping at low speed $v_d > 0$, $v_d \leq v_{0,r}$ (front/rear) (N s/m)
TB	TB factor (ISO 7401) (s rad)
v_w	wheel absolute speed (m/s)
v_d	shock absorber elongation speed (m/s)
$v_{0,b}$	shock absorber transition speed (bump) (m/s)
$v_{0,r}$	shock absorber transition speed (rebound) (m/s)
y_{path}	deviation from straight path during J-turn manoeuvre (m)
α	tyre slip angle (rad)
β	sideslip angle at vehicle centre of gravity (rad)
γ	tyre camber angle (rad)
$\gamma_{st,F/R}$	static camber angle (front/rear) (rad)
δ	steering angle at wheel (rad)
$\delta_{st,F/R}$	static toe angle (front/rear) (rad)
ϑ	pitch angle (rad)
χ	relaxation length (m)
φ	roll angle (rad)
ψ	yaw angle (rad)
ρ	path radius (m)

Chapter 8

a_x	vehicle longitudinal acceleration (m/s^2)
a_y	vehicle lateral acceleration (m/s^2)
$c_{F/R}$	half track front/rear (m)
d_w	horizontal distance between vehicle centre of gravity and wind pressure centre (m)
d	distance from reference path (m)
D	discomfort index
F	vector of forces (N)
F_w	wind force (N)
$G_{1F/R}$	front/rear active suspension gain (N/m)
$G_{2F/R}$	front/rear active suspension gain (N s/m)
h	height of centre of gravity (m)

$J_{1,2}$	steering step input objective function, see Table 8.1
J_w	polar moment of inertia of wheels (kg m)
$J_x/J_y/J_z$	moments of inertia of the vehicle body around $x-y-z$ directions (kg m)
$k_{rF/R}$	suspension anti–roll bar stiffness front/rear (N m/rad)
$k_{sF/R}$	suspension spring stiffness front/rear (m)
$k_{wR/F}$	tyre radial stiffness front/rear (N/m)
K_R/K_S	see Table 8.1
K_{dfr}/K_{dfr2}	see Table 8.1
$\mathbf{K}$	stiffness matrix
$l_{F/R}$	horizontal distance between vehicle centre of gravity and front/rear axle (m)
L_1/L_2	see Table 8.1
m	vehicle body mass (kg)
$m_{F/R}$	wheel mass front/rear (kg)
$\mathbf{M}$	mass matrix
P_1/P_2	see Table 8.1
P_{SDi}	generic power spectral density as function of circular frequency
$\mathbf{q}$	vector of coordinates
$r_{R/F}$	suspension damping front/rear (N s/m)
$\mathbf{R}$	damping matrix
$R_{F/R}$	roll moment at front/rear axle (N m)
R_m	roll moment ratio
s	number of combination of $n_{ac}-1$ element extracted from a set of n_{ac} elements
T_o/T_i	torque at outer/inner wheel with respect to a bend (N m)
v	vehicle speed (m/s)
W	see Table 8.1
β	sideslip at centre of gravity (rad)
δ/δ_R	wheel angle front/rear (rad)
Ω_o/Ω_i	angular velocity at outer/inner wheel with respect to a bend (rad/s)
$\varkappa$	driver's gain (see Sect. 8.1)
$\dot{\psi}$	yaw velocity (rad/s)
$\dot{\psi}^\star$	reference yaw velocity (rad/s)
ϱ	wheel radius (m)
$\sigma_{F_zF/R}$	standard deviation of vertical force tyre-ground front/rear (N)
$\sigma_{wsF/R}$	standard deviation of relative vertical displacement wheel–body front/rear (m/s)
τ	driver preview time (s)

Chapter 10

A_b	track irregularity (1S-PSD) (m^{-1})
A_v	track irregularity (2S-PSD) (m)
B	track irregularity factor (1S-PSD) function of vehicle speed $B = 2\pi^2 v^{3/2}\sqrt{A_b}$
W_{qS}	white noise power spectral density ($q = 1$ for the 1S–PSD, $q = 2$ for the 2S-PSD)
k_1	primary suspension stiffness (referring to one single axle-box) (N/m)
k_2	secondary suspension stiffness (referring to one single axle-box) (N/m)
m_1	1/4 of bogie mass (kg)
m_2	1/8 of body mass (kg)
r_1	primary suspension damping (referring to one single axle–box) (N s/m)
r_2	secondary suspension damping (referring to one single axle–box) (N s/m)
ω	circular frequency (rad/s)
s	$s{=}j\omega$
ω_c	reference circular frequency (2S-PSD) $\omega_c = v\Omega_c$ (rad/s)
$P_{SD\xi}(\omega)$	power spectral density of the vertical track irregularity (ξ) ($\mathrm{m}^2/(\mathrm{rad\,s})$)
v	vehicle speed (m/s)
F_z	vertical force on the axle-box (N)
ξ	absolute vertical displacement of axle-box (m) (equal, by hypothesis, to vertical track irregularity)
z_1	absolute vertical displacement of mass m_1 (m)
z_2	absolute vertical displacement of mass m_2 (m)
β	stiffness ratio $\beta = k_2/k_1$
μ	mass ratio $\mu{=}\ m_2/m_1$
σ_{F_z}	standard deviation of the vertical force on the axle-box (N)
$\sigma_{\ddot{x}_2}$	standard deviation of vertical body acceleration ($\mathrm{m/s}^2$)
$\sigma_{z_2-z_1}$	standard deviation of the stroke of the secondary suspension (m)
$\sigma_{z_1-\xi}$	standard deviation of the stroke of the primary suspension (m)
Ω_C	break wave number (2S-PSD) (rad/m)

Subscript

r	reference value

Chapter 11

A	track irregularity coefficient
A_{seat}	room for one seat (IC train) (m)
C_{LCC}, C_C	life cycle cost and purchase costs
C_M, C_E	maintenance and operating costs
ϵ_{Ni}	track settlement (damage)
f_1, f_2	lateral displacements of the vehicle body into a bend (m)
H_i	generic transfer function
$\mathbf{K}$	stiffness matrix
k_0, k_1, k_2	ballast, primary and secondary suspension stiffness (N/m)
l	car length (m)
$\mathbf{M}$	mass matrix
m_b, m_c	bogie and body masses (3 d.o.f. system model) (kg)
m_w, m_{tr}	wheelset and track masses (3 d.o.f. system model) (kg)
m_r	ratio of the tare mass and the total mass of the vehicle (kg)
N	number of load cycles acting on a track section
n_{pass}	number of passengers in a car
n_{spr}	number of seats per row
n_w	number of wheelsets per car
Ω	circular frequency (rad/s)
p	payload (kg)
$\mathbf{R}$	damping matrix
r_0, r_1, r_2	ballast, primary and secondary suspension damping (N s/m)
S_i	generic power spectral density
σ_i	generic standard deviation
St_{pass}	number of standing passengers per car
v	vehicle speed (m/s)
V_r	residual value of the vehicle
W_s	gauge cross-width (m)
w_s	seat cross-width (m)
w_v	vehicle cross-width (m)
z, h, u, y, x	vertical displacements (3 d.o.f. system model)

Chapter 12

a	$a = r_i/r_o$
a	projected dimension of the warp and fill tow (m)
a_0	non-undulated length of a (m)
a_u	undulated length of a (m)
c	$c = R/r_o$

C_f	coefficient that account curvature effects in the computation of maximum torsional stress
d_i	inner diameter of wire circular section of tubular spring (m)
d_o	outer diameter of wire circular section of spring (m)
D	diameter of spring coil measured from spring axis to centre of section (m)
E	material modulus of elasticity (Young's modulus) (MPa)
f	spring deflection (m)
f_s	maximum spring deflection (m)
f_{s_0}	reference maximum deflection (m)
G	material modulus of tangential elasticity (MPa)
h	$h = r_o - r_i$ (m)
h_h	thickness of each ply (m)
h_t	thickness of warp and fill tows (m)
i_c	spring number of coils
J_b	bending moment of inertia of the tubular section of the spring (m^4)
J_n	bending moment of inertia of the tubular section of the spring (m^4)
J_p	polar moment of inertia of the tubular section of the spring (m^4)
k	axial spring stiffness (N/m)
k_a	$k_a = k$ (N/m)
k_r	bending spring stiffness (N/°)
k_c	coupling spring stiffness (N/°)
L	length of the bar (m)
l	spring lenght (m)
l_r	$2\pi r$ (m)
M_{byi}	bending moment for the generic lamina i (Nm)
M_{ti}	torsion moment for the generic lamina i (Nm)
m	spring mass (kg)
m_t	tubular spring mass (kg)
N_{xi}	axial load in x-direction for the generic lamina i (N)
$\mathbf{n}_i^*$	vector of principal load on a single lamina i
$\mathbf{n}^*$	vector of total principal load
n_l	number of laminae
n_r	number of resonance torsional mode
P	spring external load (N)
P_c	spring critical load (N)
P_s	maximum spring load (N)
p_d	packing density of fibres (m^3)
Q_{mi}	'Mariotte' load per unit length of the tubular bar on the generic lamina i (N/m)
R	$D/2$ (m)
S	shear limit stress (MPa)

r, θ	polar local coordinate system
r_i	inner radius of circular section of tubular spring (m)
r_{ii}	inner radius of the single lamina i (m)
r_m	mean radius of circular section of spring (m)
r_o	outer radius of circular section of spring (m)
r_{oi}	outer radius of the single lamina i (m)
$\mathbf{u}^*$	vector of generalised local displacements
v_f	fibres volume fraction ($1/m^3$)
V_1	velocity of propagation of torsional mode (m/s)
V_2	velocity of propagation of flexural mode (m/s)
u, v, w	displacements along $x - y - z$ axes
X_t, Y_t	tension limit stresses (MPa)
X_c, Y_c	compression limit stresses (MPa)
x, y, z	local coordinate system (see Fig. 12.1)
α	spring helix angle (°)
β	orientation of composite laminae (°)
δ_a	axial deflection of the spring (m)
δ_c	spring deflection at critical load (m)
η	integer which accounts for the way the spring ends are constrained for the computation of spring resonances
φ	rotation of the hollow circular section (°)
$\boldsymbol{\epsilon}$	strain vector
ϵ_x	longitudinal strain – x-axis
ϵ_y	longitudinal strain – y-axis
γ	shear strain
ν	Poissons's ratio
$\boldsymbol{\sigma}$	stress vector
σ_x	longitudinal stress – x-axis (MPa)
σ_y	longitudinal stress – y-axis (MPa)
θ	inclination angle of the fill (°)
τ	shear stress (MPa)
τ_l	spring maximum shear stress produced (MPa)
τ_{max}	maximum torsional stress (MPa)
ρ	spring material density (kg/m^3)
ω_{1n_r}	n_rth torsional mode (rad/s)
ω_{2n_r}	n_rth flexural mode (rad/s)
ω_{max}	max allowable lower resonance frequency (rad/s)

Subscripts

$_{cr}$	critical
$_i$	inner

$_O$	quantity related to orthotropy planes
$_o$	outer

Chapter 13

E	material modulus of elasticity (Young's modulus) (MPa)
F_r	infinitesimal volume radial force (N)
F_θ	infinitesimal volume circumferential force (N)
F_m	infinitesimal volume centrifugal force (N)
h	flywheel thickness (mm)
K	flywheel kinetic energy stored (J)
M	flywheel total mass (kg)
r, θ, z	flywheel principal coordinates
u	radial displacement
$\varepsilon_r, \varepsilon_z, \varepsilon_\theta$	flywheel principal strains (MPa)
Γ	stress function
ν	Poisson's ratio
ρ	flywheel material density (kg/m^3)
$\sigma_r, \sigma_z, \sigma_\theta$	flywheel principal stresses (MPa)
σ_{VMM}	Von Mises reference stress (MPa)

Part I

Theory

1

Introduction to the Optimal Design of Complex Mechanical Systems

When a designer is able to simulate the physical behaviour of a system by means of a validated mathematical model, the subsequent task is that of defining the system model parameters (also called design variables, see Chap. 2) in order to obtain the desired system performances (also called objective functions, see Chap. 2).

Often such performances are conflicting, i.e. improving one implies the worsening of another, so a compromise has to be reached.

When the system model parameters are more than four or five and the system's objective functions are more than five or six, both the definition of parameters and the balancing of conflicting performances may become cumbersome or even impossible unless a special approach is adopted.

This Chapter will deal with the above-stated problem, i.e. the definition of system model parameters of complex systems, when conflicting performances have to be balanced. The design of complex systems will be accomplished by exploiting multi-objective programming (MOP), an optimisation theory pertaining to operational research (OR).

1.1 On the Optimal Design of Complex Systems

In order to introduce to the reader what is meant exactly by optimisation of a complex system, let us resort to an example. Let us imagine that a cantilever has to be optimally designed[1]. The example actually deals with a simple system, however, in this apparently simple problem, the main features pertaining to the optimisation of complex systems are present.

The cantilever, shown in Fig. 1.1, has a rectangular cross-section and a force acts at the free end. Let us assume that the optimisation problem to be solved is

[1]An effective *design process* should always produce an optimal solution, so *optimisation* and *design* are reputed to be the *same process* by the authors

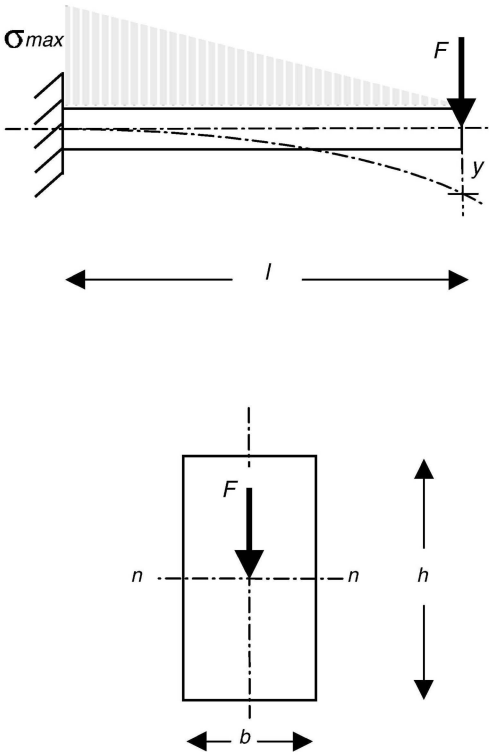


Fig. 1.1. Cantilever whose rectangular cross-section is to be defined in order to minimise both the cantilever mass and the cantilever deflection at the free end

- *finding the values of the design variables (length b and length h) defining the cantilever cross-section*[2]

in order to

- *minimise the cantilever mass*
- *minimise the cantilever deflection at its free end,*

subject to the following conditions

- *the maximum stress at the fixed end must be little than (or equal to) the admissible stress*
- *elastic stability must be guaranteed (i.e. buckling must be avoided).*

A designer should chose the values defining the cross-section of the cantilever in order to get it as light and stiff as possible, avoiding both a failure (due to too high stress) and elastic instability.

[2] b and h may vary, respectively, within two well-defined ranges

In mathematical form, the above optimisation problem may be stated as follows:

- *Given*
  - l the cantilever length (m)
  - b the beam cross-section width (m)
  - h the beam cross-section height (m)
  - J $= \frac{1}{12}bh^3$, the flexural moment of inertia of the section (n–n axis) (m^4)
  - F the force applied at the cantilever free end (see Fig. 1.1) (N)
  - σ_E the material yield stress (MPa)
  - η safety coefficient ($\geq$1)
  - $\sigma_{adm} = \sigma_E/\eta$ the admissible stress at the cantilever fixed end (MPa)
  - E the material modulus of elasticity (Young's modulus) (MPa)
  - G the material modulus of tangential elasticity (MPa)
  - ρ the material density (kg/m^3)
- *and defining*
  - $m = \rho bhl$ the cantilever mass (kg)
  - $y = \frac{1}{3}\frac{Fl^3}{EJ} = 4\frac{Fl^3}{Ebh^3}$ the deflection at the free end due to load F (m)
  - $\sigma_{max} = 6\frac{Fl}{bh^2}$ the maximum stress located at the top of the cross-section at the fixed end of the cantilever (MPa)
  - $F_{cr} = \frac{k_1 b^3 h}{l^2}\sqrt{\left(1 - k_2\frac{b}{h}\right)EG}$ the critical load (see [270]) (N)
- *find b and h such that*

$$b_{min} \leq b \leq b_{max}$$
$$h_{min} \leq h \leq h_{max}$$

- *and such that*

$$\min \begin{pmatrix} m(b,h) \\ y(b,h) \end{pmatrix} = \begin{pmatrix} \rho bhl \\ 4\frac{Fl^3}{Ebh^3} \end{pmatrix} \tag{1.1}$$

- *subject to*

$$\sigma_{max} = 6\frac{Fl}{bh^2} \leq \sigma_{adm} = \sigma_E/\eta \tag{1.2}$$

$$F < F_{cr} = \frac{k_1 b^3 h}{l^2}\sqrt{\left(1 - k_2\frac{b}{h}\right)EG} \tag{1.3}$$

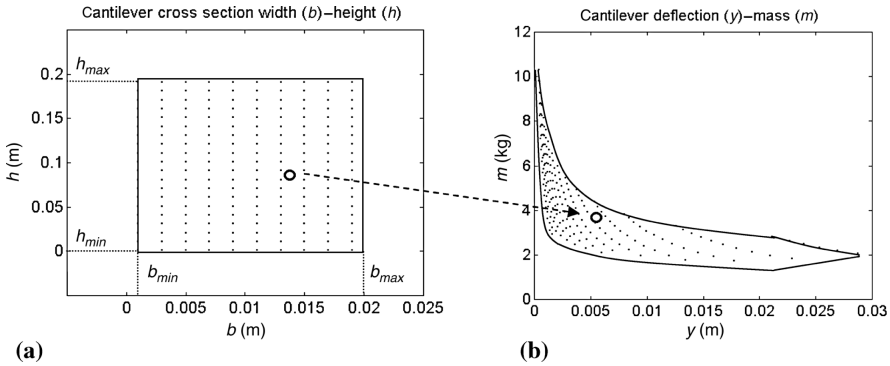


Fig. 1.2. (**a**) The cantilever cross-section width b and height h that are considered for the optimisation. (**b**) Cantilever mass $m(b,h)$ and cantilever deflection at the free end $y(b,h)$. The manifold in b accounts for the constraints on the maximum stress (1.2) and on the critical load (1.3). Data (symbols referring to (1.2) , (1.3), and Fig. 1.1): $b_{min} = 0.001\,\mathrm{m}$, $b_{max} = 0.020\,\mathrm{m}$, $h_{min} = 0.001\,\mathrm{m}$, $h_{max} = 0.200\,\mathrm{m}$, $\rho = 2,700\,\mathrm{kg/m^3}$ $E = 70,000\,\mathrm{MPa}$, $G = 27,000\,\mathrm{MPa}$, $\sigma_E = 160\,\mathrm{MPa}$, $\eta = 1$, $F = 1,000\,\mathrm{N}$, $l = 1\,\mathrm{m}$, $k_1 = 0.669$, $k_2 = 0.63$

The optimisation problem aims to minimise[3] the *objective functions* $(m(b,h),\ y(b,h))$ by selecting properly the *design variables*[4] (b,h) and by satisfying the constraints on F_{cr} and σ_{adm}.

In order to solve the problem, one may compute $m(b,h)$ and $y(b,h)$ as function of all possible combinations of b and h.

This is a naive but reasonable approach. In other words, if a couple of values b,h are selected, then, correspondingly, a couple of values $m(b,h)$, $y(b,h)$ can be computed. The values of b and h have to be varied, respectively, within the prescribed ranges $b_{min} \le b \le b_{max}$ and $h_{min} \le h \le h_{max}$. The inequalities (1.2) and (1.3) have to be evaluated for all possible combinations of b and h: if the inequalities are satisfied, the couple of values $m(b,h)$, $y(b,h)$ is kept, otherwise it is discarded.

In Fig. 1.2 the results of such a computation are shown. At one point in the rectangle on the plane b–h corresponds a point in the manifold on the plane y–m. The rectangular manifold on the plane b–h is transformed into the

[3] If, for a generic optimisation problem, the objective functions were to be *maximised*, changing their signs would transform the maximisation problem into a minimisation one, so the presented example dealing with minimisation is quite general

[4] b and h are precisely *parameters*, with the property that they have to be varied, thus they are *variable parameters*, which is a nonsense as parameters do have fixed values. To avoid misunderstanding these variable parameters are named *design variables* to distinguish them from actual parameters having fixed values

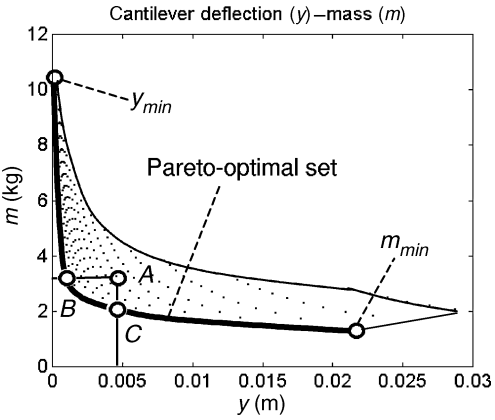


Fig. 1.3. Definition of the Pareto-optimal set in the objective functions domain. Data in Fig. 1.2

manifold on the plane y–m. In other words, a transformation is established between the domain of design variables and the domain of objective functions.

As all the possible combinations of design variables b, h have been used to generate $m(b,h)$ $y(b,h)$ (having verified that $m(b,h)$, $y(b,h)$ do satisfy the constraints (1.2) and (1.3)), the question is now how to find the values of b, h which minimise concurrently the mass m and the deflection y at the free end of the cantilever.

The definition of the solution of the addressed optimisation problem is not straightforward and requires a special reasoning.

Let us consider the couple of values $m_A(b_A, h_A)$, $y_A(b_A, h_A)$ referring to point A in Fig. 1.3. The couple of values b_A, h_A defines a cantilever cross-section which *should not* be considered by a designer, that is, the cantilever A is a wrong solution for the addressed optimisation problem.

The reason for this is readily explained. Let us consider point $B(m_B(b_B, h_B), y_B(b_B, h_B))$. It is $m_A(b_A, h_A) = m_B(b_B, h_B)$ but $y_B(b_B, h_B) < y_A(b_A, h_A)$: the two cantilevers have the same mass m but the deflection of the cantilever B is smaller than that of the cantilever A. Thus, the cantilever B is better than cantilever A. Similarly, for cantilever C, $m_C(b_C, h_C) < m_A(b_A, h_A)$ and $y_A(b_A, h_A) = y_C(b_C, h_C)$. Also the cantilever C is better than cantilever A because they have the same deflection but the mass of the cantilever C is smaller than that of the cantilever A. Point A is said to be *dominated* by B and C, i.e. B and C *dominate* A. By inspection of Fig. 1.3, A is also dominated by all the design solutions between B and C on the bold line, i.e. the solutions on the bold line between B and C are better at least in one objective function than A.

Given a (wrong) solution A, there exist (right) solutions which are better than A at least in one objective function. All the solutions corresponding to points in Fig. 1.3 which do not lie on the bold line (defined by the two end

points y_{min} and m_{min}) are *wrong solutions* to be discarded by a designer. Conversely, the good or *optimal solutions* are those and only those which are represented by points lying on the said bold line, the set of these solutions is called *Pareto-optimal set*[5].

The task of the designer is that of choosing a solution only from the Pareto-optimal set.

It is evident that the designer can choose among an unlimited number of solutions, i.e. the solution is not unique.

This is due to the fact that the addressed optimisation problem required to minimise concurrently $m(b,h)$ and $y(b,h)$, i.e. *the minimisation of a vector function minimisation of a vector function* had to be performed.

The minimisation of a vector function is an issue typical in the optimisation of complex systems. Presently, it does not seem to be very well known by designers, even if the theory was developed more than one century ago. On the contrary, the minimisation of a *scalar function*[6] of one or more variables is generally reputed as a relatively simple task.

In order to explain what does it mean 'minimising a vector function', let us start to minimise a scalar function, i.e. let us separately minimise $m(b,h)$ and $y(b,h)$. These two scalar functions are obviously to be minimised by taking into account that constraints (1.2) and (1.3), i.e. a *constrained minimisation* has to be performed.

If only one objective function was considered, e.g. $m(b,h)$, the problem could be formulated mathematically as follows:

- *Find* b *and* h
- *such that*

$$b_{min} \leq b \leq b_{max}$$
$$h_{min} \leq h \leq h_{max}$$

- *and such that*

$$\min m(b,h) = \min \rho bhl$$

- *subject to*

$$\sigma_{max} = 6\frac{Fl}{bh^2} \leq \sigma_{adm} = \sigma_E/\eta$$
$$F < F_{cr} = \frac{k_1 b^3 h}{l^2}\sqrt{\left(1 - k_2\frac{b}{h}\right)EG}$$

[5] Vilfredo Pareto (1848–1923) was an Italian engineer, who later became a famous sociologist and economist at the University of Lousanne

[6] for example, minimising only $m(b,h)$ or only $y(b,h)$)

The solution to this optimisation problem in which only one objective function is to be minimised can be denoted by

$$m_{m_{min}}(b_{m_{min}}, h_{m_{min}}), y_{m_{min}}(b_{m_{min}}, h_{m_{min}})$$

Similarly, by minimising $y(b, h)$ the solution can be written as

$$m_{y_{min}}(b_{y_{min}}, h_{y_{min}}), y_{y_{min}}(b_{y_{min}}, h_{y_{min}})$$

It has to be noticed that if a scalar function has to be minimised, the solution is unique

$$\min m(b, h) \rightarrow m_{m_{min}}(b_{m_{min}}, h_{m_{min}}), y_{m_{min}}(b_{m_{min}}, h_{m_{min}}) \rightarrow 1\,\text{solution}$$

$$\min y(b, h) \rightarrow m_{y_{min}}(b_{y_{min}}, h_{y_{min}}), y_{y_{min}}(b_{y_{min}}, h_{y_{min}}) \rightarrow 1\,\text{solution}$$

but, if a vector function has to be minimised, the solution is not unique

$$\min \begin{pmatrix} m(b, h) \\ y(b, h) \end{pmatrix} \rightarrow \text{Pareto-optimal set} \rightarrow \infty^1 \text{optimal solutions}$$

The minimisation of a vector function involves the definition of solutions. So the concurrent optimisation of more than two objective functions involves finding infinite solutions.

This occurrence is very important and may be reputed as the key issue of multi-objective programming.

The Pareto-optimal set is the set which contains all the infinite solutions coming from the minimisation of a vector function.

The designer has to select one solution from the Pareto-optimal set, and this selection is inherently somewhat subjective. The Pareto-optimal set does contain (by definition) all the best compromise solutions between conflicting objective functions. All of the Pareto-optimal solutions do have the status of the best compromise solutions. The degree or level of the compromise varies in a fuzzy way among solutions. Selecting the desired level of the compromise is the primary or ultimate task of the designer.

The designer has to act as a *decision maker*, actually, by selecting a solution from the Pareto-optimal set, he decides his desired best compromise between conflicting objective functions.

In our example, the designer has to chose (from the Pareto-optimal set, and from this set only) the preferred couple of values $m(b, h), y(b, h)$. It is clear that the designer will disregard all of the so-called dominated solutions (the solutions corresponding to points lying inside the manifold in Fig. 1.2 (b)), and will chose one solution from the dominating ones (belonging to the Pareto-set).

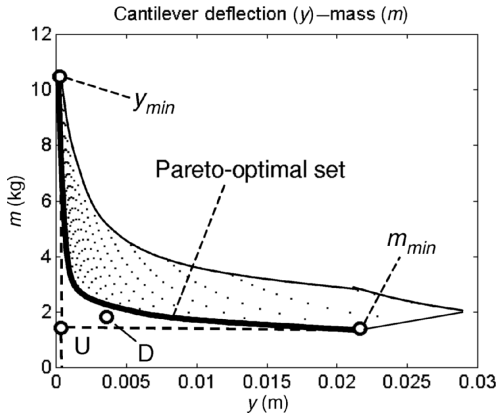


Fig. 1.4. Definition of the utopia or ideal point U (coordinates are m_{min}, y_{min}). The objective functions (m, y) referring to point D cannot be attained by the considered system. Data in Fig. 1.2

In the literature [59, 238], many attempts have been made to help the designer to make a choice. The results are often questionable, mainly because it is assumed that the designer knows perfectly the desired compromise to be reached among conflicting objective functions. This is not always the case, or better, in the design of complex systems, it is *never* the case.

For example, even in our case, it is not easy to answer the questions *how small the deflection should be* and *how light the cantilever should be*? It is a matter which might involve additional considerations which were not taken into account at the stage of problem formulation.

What actually happens during the optimisation of a complex system is that the designer, before making his choice, acquires an in–depth knowledge of the system performances under investigation. Particularly, he understands what are the limit performances of the system.

For example, in our case, he will realise that an objective function defined by point D in Fig. 1.4 will never be attained, unless the system is changed. Also y_{min} and m_{min} are important boundaries to the performances (i.e. objective functions). These boundaries, depending on the problem under investigation, could be changed by varying the design variable ranges (i.e. the ranges into which the design variables are varied during the optimisation process).

An important reference point is the *utopia or ideal point*. In our example, it is the point whose coordinates are (m_{min}, y_{min}). Obviously, a cantilever having such performances (i.e. objective functions) does not exist, however if it existed, it would be the best solution. The utopia point can be used to choose Pareto-optimal solutions. Empirical reasoning suggests to choose those Pareto-optimal solutions which are closer to the utopia point.

If the system is complex (tenth of objective functions and tenth of design variables) the knowledge and understanding process (necessary for an effective

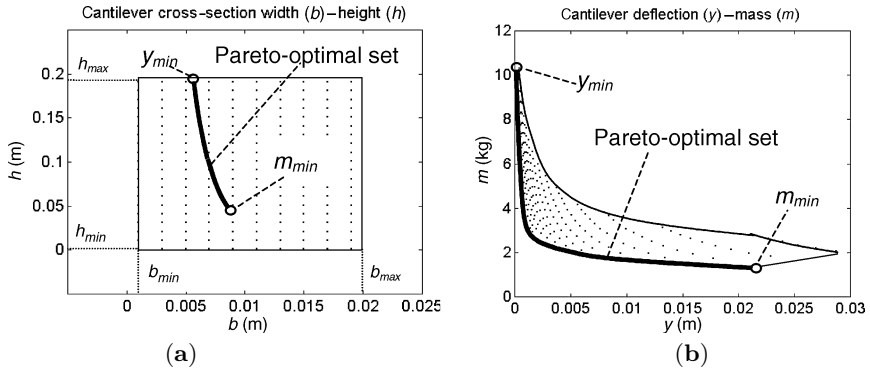


Fig. 1.5. The two Pareto-optimal sets defined, respectively, in the design variable domain (a) and in the objective function domain (b). Data in Fig. 1.2

optimisation) may be heavy and time-consuming. Sometimes the time for optimising a complex system may require as much time as that which was spent to develop the validated mathematical model of the system under investigation. When a designer has at his disposal a validated mathematical model of a system, he might be only half way (or less!) from the complete (optimal) design of the system.

As the designer has chosen the preferred compromise among conflicting objective functions by selecting a solution from the Pareto-optimal set, then the problem is to define or identify the values of the design variables related to those objective functions. In other words, the designer has to define the design variables that allow the system to perform according to his wishes.
So we have to consider that, having a set of Pareto-optimal solutions, what we really want are the values of design variables corresponding to those Pareto-optimal solutions. It can be useful to represent the Pareto-optimal solutions both in the objective functions domain or conversely, in the design variables domain (see Fig. 1.5).

With reference to our optimisation problem, in Fig. 1.5, the Pareto-optimal set defined in the design variable domain is shown. As it could be expected, the stiffer cantilever has the highest cross-section height h.
There is a direct relationship between the points of the two Pareto-optimal sets represented in Fig. 1.5. It is obvious that the designer will always select the design variables from the Pareto-optimal set represented in Fig. 1.5(a).

The shapes of the cross-sections of three Pareto-optimal cantilevers are shown in Fig. 1.6. Two of them refer respectively to points m_{min} and y_{min}. The intermediate cantilever in Fig. 1.6 has both the minimum mass m and the minimum deflection y, according to the definition of the best compromise addressed before.

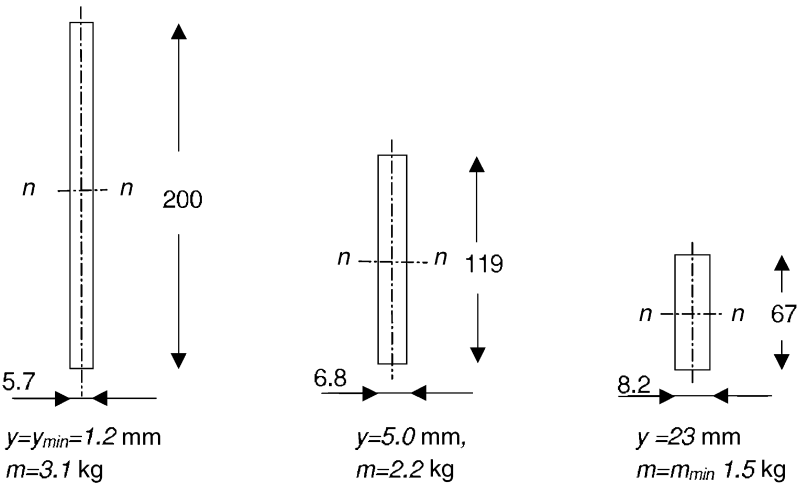


Fig. 1.6. Cross-sections of cantilevers belonging to the Pareto-optimal set. Dimensions in mm. *Left*: Minimum deflection cantilever (point y_{min} in Fig. 1.5). *Centre*: A generic Pareto-optimal cantilever. *Right*: Minimum mass cantilever (point m_{min} in Fig. 1.5). Data are shown in Fig. 1.2

1.2 Finding the Pareto-optimal Sets

From the above considerations, it is clear that a designer should always try to find the Pareto-optimal sets (both in the design variable space and in the objective function space) to obtain the best solutions. The existence of the Pareto-optimal set is stated by a theorem which guarantees non-empty Pareto-optimal sets under broad conditions[7], holding for the majority of engineering problems.

In the engineering practice, very rarely problems are solved by resorting to the computation of the Pareto-optimal sets (Fig. 1.7). In fact, the computations to find the Pareto-optimal sets are rather involved and time-consuming[8]. However, in the last years, the ever-increasing power of computers has allowed to attempt the optimal design of complex systems on the basis of Pareto theory.

1.2.1 Exhaustive Method

There are many methods to find Pareto-optimal sets (see Sect. 3.4.1). The one, called exhaustive method, which has been used in the presented example referring to the cantilever, is the simplest, but unmanageable due to the huge

[7]The assumptions are that the design variable space is closed and the objective functions are continuous functions of the design variables [166]

[8]This seems the main reason why, presently, very few designers are instructed to apply the optimisation theory based on Pareto-optimality

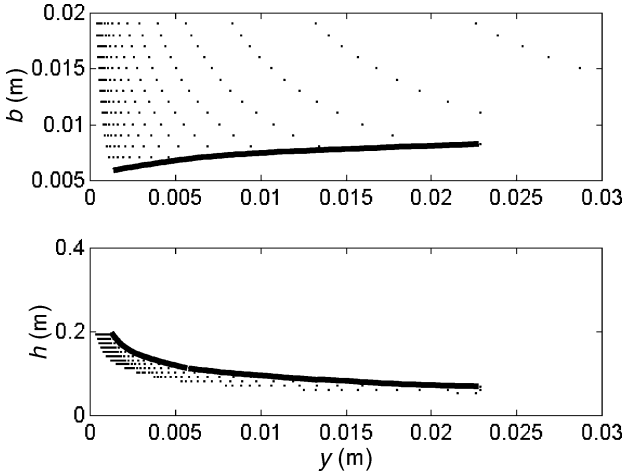


Fig. 1.7. Pareto-optimal cross-section width b and Pareto-optimal cross-section height h as function of Pareto-optimal deflection y. By these two graphs, the direct relationship between the points of the two Pareto-optimal sets in Fig. 1.5 can be obtained

amount of computations required. To explain why the exhaustive method is unmanageable, let us resort to an example.

Let us imagine that a complex system is defined by 10 design variables n_{dv} and that 30 objective functions n_{of} have to be taken into account. Each design variable may assume $n_v = 10$ different values within its definition range. Let us assume that, given a combination of design variables, the time t_s for performing a simulation to obtain the value of one single objective function is 1 s. By the exhaustive method, the number of all possible combinations of design variables values is

$$n_{cdv} = n_v^{n_{dv}} \tag{1.4}$$

and the total time t_t for computing all of the objective functions as function of all possible combinations of design variables is

$$t_t = t_s n_{of} n_v^{n_{dv}} \tag{1.5}$$

Substituting the numerical values

$$t_t = t_s n_{of} n_v^{n_{dv}} = 1 \times 30 \times 10^{10}\,\mathrm{s} = 3 \times 10^{11}\,\mathrm{s} = 9,645\,\mathrm{years} \tag{1.6}$$

the total time t_t for computing all the objective functions is clearly a time too long to solve an engineering problem (10,000 years seems to be the age of *homo sapiens sapiens*). Additionally if, for some practical reason, the number of design variables should be increased by two (from 10 to 12), the total time requested for simulations would be 100 times longer

$$t_t = t_s n_{of} n_v^{n_{dv}} = 1 \times 30 \times 10^{10+2}\,\mathrm{s} = 100 \times 9{,}645\,\mathrm{years} = 96{,}4500\,\mathrm{years} \tag{1.7}$$

These figures are startling and state evidently that an exhaustive construction[9] of the Pareto-optimal set is, in general, not possible, unless the system under consideration is very simple.

The critical factors in (1.5) are the number of design variables n_{dv}, the number of values the design variables may assume within their respective definition ranges n_v, and the simulation time for computing one single objective function t_s. Often, the number of objective functions n_{of} is not critical.

1.2.2 Uniformly Distributed Sequences and Random Search

There have been conceived many methods (see Sect. 3.4.2) to reduce the above-addressed total number of simulations ${n_v}^{n_{dv}}$, in fact, it is not absolutely necessary to explore all the possible design configurations to construct the Pareto-optimal sets.

Sometimes it is sufficient to reduce (to a considerable extent) the number of simulations to estimate properly the Pareto-optimal set both in the space of objective functions and in the space of design variables.

For example, in Fig. 1.8(a) a regular grid representing an ordered combination of design variables values is compared with two different combinations (centre and right) which are still ordered (in the sense that points are not randomly distributed) but the number of points is greatly reduced with respect to the previous combination. Points related to the combination in Fig. 1.8 (centre) are somehow 'equally' distributed along vertical planes. Every square of the three vertical planes contains one point. Points along these planes are somehow 'well distributed'. For each vertical plane, one out of the three coordinates of a point is fixed. So – in order to reconstruct the relationships between the design variables and the objective functions – the informative contribution given by the design variable, which is kept fixed, might be nearly the same for all points lying on the same plane. This causes a computational inefficiency as many simulations will be performed at a given (fixed) value of a design variable.

The distribution in Fig. 1.8(c) overcomes this problem, the information on the relationships between the objective functions and the design variables is complete and less redundant. From a mathematical point of view, the above explanation is very rough, anyway it gives some hint on how and why simulations can be reduced to estimate the Pareto-optimal sets.

Orthogonal arrays and *low discrepancy sequences*[10] are particularly suited to be used to reduce as much as possible the number of combinations of design variables used as input data for simulations. These sequences are called

[9] When all the simulations have been made on the basis of all possible combinations of design variables, the construction of all Pareto-optimal sets is performed by selecting the so-called dominating solutions from the dominated ones. This is mathematically performed by applying a proper definition that will be presented in Chap. 2

[10] Proper information on this will be given in Sect. 3.4.2

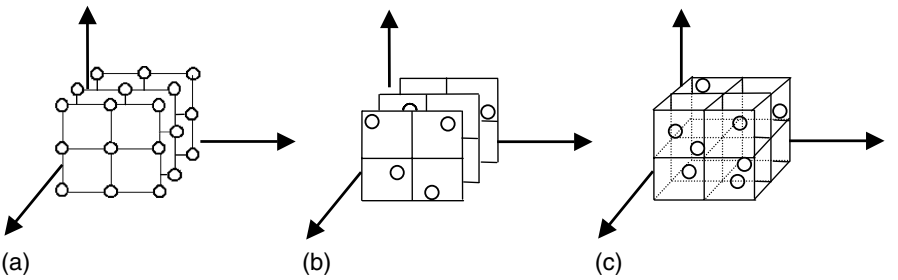


Fig. 1.8. Three different combinations of design variables values. Number of design variables: $n_{dv} = 3$. (**a**) Regular grid, number of values the design variables may assume within their respective definition ranges: $n_v = 3$, total number of design variable combinations: 27. (**b**) Low-discrepancy grid, total number of design variable combinations: 12. (**c**) Low discrepancy grid, total number of design variable combinations: 8

uniformly distributed sequences and are said to have *low discrepancy* as the points are uniformly distributed in the space. In multi-dimensional spaces (>3 up to 30 or more dimensions), the low discrepancy placement of points is a peculiar mathematical matter which requires a special theoretical background.

1.2.3 Genetic Algorithms

Another well-known method to optimise complex systems is based on Genetic algorithms (GAs)(see Sect. 3.4.4). This approach mimics what has happened during the evolution of living creatures. Living creatures evolve by adapting as much as possible to the environment. It is assumed that

- individuals have a chance to reproduce themselves according to their fitness (reproduction step),
- an individual cross its genes with the ones of its partner to generate a new individual (crossover step),
- some mutation of the genes always acts during the previous crossover process (mutation step)

The generated individuals start a new three-step generation process (reproduction, crossover, mutation). A number of generations are necessary to select fit individuals belonging to Pareto-optimal sets.

A design project evolves by adapting as much as possible both to the designer aims and to the design constraints.

An individual corresponds to a design solution, the genes correspond to a design variable combination which identifies a single design solution. A direct mathematical relationship is established between genes and design variables: the design variables values are expressed in binary form and these binary strings correspond to the genes. Each design variable can be converted into

its binary equivalent, and thereby mapped into a fixed length of zeros and ones.

An individual/design solution is selected for crossover according to its fitness. The fitness is related to the values of the objective functions. If the fitness is high, the individual/design solution will have more chances to reproduce, i.e. crossing its genes with those of a partner. The crossover process just mixes the binary strings pertaining to genes/design variables of two parents to generate one new individual. The genes/design variables of the new individual/design solution may slightly vary due to mutation. GAs do have many variants which are often developed to solve particular optimisation problems.

GAs are very useful especially when the design variables values take discrete values. Unfortunately, they are not very simple to be used (with respect to other methods such as global approximation, see next subsection) especially when dealing with the optimisation of (very) complex systems. In fact they often require the exact definition of some algorithm parameters that influence the efficiency of the search.

1.2.4 Comparison of Broadly Applicable Methods to Solve Optimisation Problems

In Table 1.1 four general and broadly applicable methods to solve optimisation problems are presented together with their optimisation properties and performances. All the four methods have been introduced in the preceding subsections. They allow a vectorial formulation, i.e. they permit the optimisation of all the objective functions concurrently (direct derivation of Pareto-optimal sets).

The computational efficiency refers to how fast an optimisation can be. For simple problems (number of design variables less than 5 or 6) all the four presented methods work well. Complex problems are handled well by GAs; on the contrary, the exhaustive method is absolutely unsuited for this kind of problems.

The accuracy in the definition of the Pareto-sets is best for the exhaustive method in which design variables may vary in non-continuous ranges (discrete values). All the four methods in Table 1.1 do have the following important properties:

- they can deal with design variables which vary within non-continuous ranges (discrete values)
- objective functions can be non-continuous functions of the design variables.

1.2.5 Global Approximation

The multi-objective optimisation of very complex systems may require still a prohibitive simulation effort, even reducing the number of combinations

Table 1.1. General and broadly applicable methods for solving optimisation problems

	Computational efficiency (time)		*Accuracy*		*Discrete design variables values allowed*	*Objective functions need to be continuous functions*
	Simple problem	*Complex problem*				
Exhaustive	−	− −	++ (*discrete design variables*)	− (*continuous design variables values*)	yes	no
Uniformly distributed Sequences	+	−	−		yes	no
Genetic algorithms	+	+	+		yes	no

Other methods, very efficient for specific cases, are reported in Table 1.3 (− −: very bad; −: bad; +: good; + +: very good).

of design variables by means of the mentioned techniques. To optimise the performances of complex systems, the relationship between design variables and objective functions can be *approximated* by means of a pure mathematical model (see Chap. 4). The parameters of the purely mathematical model (which can be either an artificial neural network (ANN) or a piecewise quadratic function, or others) are defined on the basis of a limited number of simulations performed by means of the originally validated mathematical model of the system under consideration.

In other words, there are two models, the first which is the original validated mathematical model that the designer uses to simulate the actual physical behaviour of the system to be optimised, and the second mathematical model which is a purely mathematical model, able to approximate the outputs of the first model (Fig. 1.9).

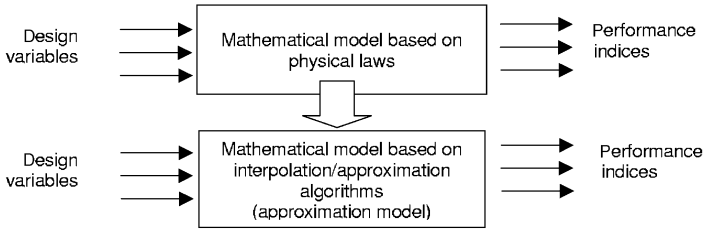


Fig. 1.9. Global approximation approach to solve optimisation problems. The original model based on physical laws is substituted by another purely mathematical model based on interpolation/approximation algorithms

Obviously it is supposed, as it always happens in actual applications, that the first model requires time-consuming simulations and the second model provides quick simulation outputs. Typically the simulation time of the second model is a small fraction (1/10 – 1/10,000) of the simulation time requested by the first model.

The accuracy in the approximation of the outputs depends on the approximation model employed. In Table 1.2, known methods for global approximation are presented. Linear and quadratic interpolation are well suited to approximate locally, i.e. in the neighbourhood of a single point, the objective functions. They are suited for very simple optimisation problems, and the tuning (i.e. the process for defining the parameters of the pure mathematical model) is relatively easy. The evaluation accuracy, i.e. the ability to reproduce the original values of objective functions given the values of design variables, is not very high. Response surface methodologies use linear and quadratic approximation models.

Radial basis functions neural networks seem to be accurate, easy to be tuned with appropriate algorithms (as will be exposed in Chap. 4) and efficient in the evaluation of objective functions. Multi-layer perceptron neural

Table 1.2. Approximation methods for solving optimization problems (–: bad; +: good; + +: very good)

	Approximation domain	*Evaluation accuracy*		*Tuning effort*
		Simple model	*Complex model*	
Linear interpolation	local	+	−	++
Quadratic interpolation	local	+	−	++
Radial basis functions Neural networks	local/global	++	+	+
Multi–layer perceptron neural networks	global	++	++	−
Statistical approximation	global	++	+	−

networks perform better than previous radial basis functions neural networks but require more effort in tuning. These approximation methods are particulary suited for complex design optimisation problems.

1.2.6 Multi-objective Programming via Non-linear Programming

Historically, multi-objective optimisation problems were numerically solved resorting to non-linear programming (NLP), i.e. by transforming the original vector problem into a scalar one (see Sect. 3.4.5). Obviously this transformation is still effective but it is not recommended any longer for solving general complex optimisation problems, i.e. finding Pareto-optimal sets.

A well-known method for transforming a vector optimisation problem into a scalar one is the *constraints method* (see Sect. 3.4.10). Given the multi-objective optimisation problem as a vector function to be minimised

$$\min \mathbf{f}(\mathbf{x}) = \min \begin{Bmatrix} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \\ \dots \\ f_{n_{of}}(\mathbf{x}) \end{Bmatrix} \qquad \mathbf{x} = \begin{Bmatrix} x_1 \\ x_2 \\ \dots \\ x_{n_{dv}} \end{Bmatrix} \tag{1.8}$$

subject to n_c constraints

$$\mathbf{g}(\mathbf{x}) = \begin{Bmatrix} g_1(\mathbf{x}) \\ g_2(\mathbf{x}) \\ \dots \\ g_{n_c}(\mathbf{x}) \end{Bmatrix} \leq 0 \tag{1.9}$$

the original vector problem is transformed into the new scalar one

$$\min f_1(\mathbf{x}) \qquad \mathbf{x} = \begin{Bmatrix} x_1 \\ x_2 \\ \dots \\ x_{n_{dv}} \end{Bmatrix} \tag{1.10}$$

subject to the $n_c + (n_{of} - 1)$ constraints

$$\mathbf{g}(\mathbf{x}) = \left\{ \begin{array}{c} f_2(\mathbf{x}) - \varepsilon_2 \\ \dots \\ f_{n_{of}}(\mathbf{x}) - \varepsilon_{n_{of}} \\ g_1(\mathbf{x}) \\ g_2(\mathbf{x}) \\ \dots \\ g_{n_c}(\mathbf{x}) \end{array} \right\} \leq 0 \tag{1.11}$$

By varying the values of the elements of the vector

$$\varepsilon = \left\{ \begin{array}{c} \varepsilon_2 \\ \dots \\ \varepsilon_{n_{of}} \end{array} \right\} \leq 0 \tag{1.12}$$

the Pareto-optimal set can be found.

This scalarisation can be applied to any kind of optimisation problem. It is not straightforward how to vary ε, so this method is only used to refine a Pareto-optimal solution. In other words, if a Pareto-optimal solution is known approximately, by applying the constraints method in the neighbourhood of the approximately known solution, the approximation can be significantly improved to the desired extent.

Another widespread method for scalarisation is the weight method (see Sect. 3.4.9). Given the multi-objective optimisation problem as a vector function to be minimised

$$\min \mathbf{f}(\mathbf{x}) = \min \left\{ \begin{array}{c} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \\ \dots \\ f_{n_{of}}(\mathbf{x}) \end{array} \right\} \qquad \mathbf{x} = \left\{ \begin{array}{c} x_1 \\ x_2 \\ \dots \\ x_{n_{dv}} \end{array} \right\} \tag{1.13}$$

subject to n_c constraints

$$\mathbf{g}(\mathbf{x}) = \left\{ \begin{array}{c} g_1(\mathbf{x}) \\ g_2(\mathbf{x}) \\ \dots \\ g_{n_c}(\mathbf{x}) \end{array} \right\} \leq 0 \tag{1.14}$$

the original vector problem is transformed into the new scalar one

$$\min(w_1 f_1(\mathbf{x}) + w_2 f_2(\mathbf{x}) + \ldots + w_{n_{of}} f_{n_{of}}(\mathbf{x})) \qquad \mathbf{x} = \left\{ \begin{array}{c} x_1 \\ x_2 \\ \dots \\ x_{n_{dv}} \end{array} \right\} \tag{1.15}$$

subject to the $n_c + (n_{of} - 1)$ constraints

$$\mathbf{g}(\mathbf{x}) = \left\{ \begin{array}{c} g_1(\mathbf{x}) \\ g_2(\mathbf{x}) \\ \dots \\ g_{n_c}(\mathbf{x}) \end{array} \right\} \leq 0 \tag{1.16}$$

by varying the elements of the vector

$$\mathbf{w} = \left\{ \begin{array}{c} w_1 \\ w_2 \\ \dots \\ w_{n_{of}} \end{array} \right\} \tag{1.17}$$

the Pareto-optimal set can be found both in the space of objective functions and in the space of design variables, provided that the Hessian of $w_1 f_1(\mathbf{x}) + w_2 f_2(\mathbf{x}) + \ldots + w_{n_{of}} f_{n_{of}}(\mathbf{x})$ is a semi-definite positive matrix in the design variable space (a sufficient condition [166] for this is that $w_1 f_1(\mathbf{x}) + w_2 f_2(\mathbf{x}) + \ldots + w_{n_{of}} f_{n_{of}}(\mathbf{x})$ is globally convex in the design variable space).

This scalarisation can be applied to any kind of convex optimisation problem, for other problems it might fail in finding all Pareto-optimal solutions.

1.2.7 Algorithms to Solve Optimisation Problems in Scalar Form

In Table 1.3, the more used algorithms for solving multi-objective optimisation problems via scalar formulation are presented and compared. Solving the optimisation problem in scalar form deals with the minimisation of one function of many design variables subject to constraints (on design variables) (see Chap. 3).

Obviously, the algorithms that could solve the optimisation problem in vector form are still available for solving the scalar problem.

The *simplex algorithm* performs well and does not require the computation of the derivatives of the function.

Sequential unconstrained minimisation technique (SUMT) is a kind of gradient method and requires the function to be continuous together with its first derivative. Discrete values are not permitted because the search is based on gradient evaluation.

Sequential quadratic programming (SQP) performs better than SUMT, and requires the same conditions on the functions to be minimised.

The main disadvantage of these methods are that they are all local methods. So the finding of global minima are all influenced by the starting point that must be close to the global solution.

1.3 Understanding Pareto-optimal Solutions

After Pareto-optimal sets have been computed, the designer can make a choice and select from these sets the preferred solution featuring the desired compromise among objective functions.

Table 1.3. Methods for solving multi-objective optimisation problems via non-linear programming (scalar formulation) (– –: very bad; –: bad; +: good; + +: very good)

	Computational efficiency		*Accuracy*		*Discrete design variables values allowed*	*Objective functions continuous functions*
	Simple problem	*Complex problem*				
Exhaustive	–	– –	++ (*discrete design variables values*)	– (*continuous design variables values*)	yes	no
Uniformly distributed sequences	+	–	–		yes	no
Genetic algorithms	+	+	+		yes	no
Simplex	+	–	+		yes	no
Sequential unconstrained minimisation technique	+	–	+		no	yes
Sequential quadratic programming	+	+	+		no	yes

This process can be preceded and even accelerated by special analyses allowing the designer to have an insight into the physical phenomena he is trying to analyse (see Sect. 3.3.3). In other words there are some analyses by which the designer may understand the reason why a Pareto-optimal solution requires such a design variable combination. In particular, these analyses may show which is

- the relationship between two Pareto-optimal objective functions,
- the relationship between two Pareto-optimal design variables,
- the relationship between a Pareto-optimal design variable and a Pareto-optimal objective function.

In Fig. 1.10 these different relationships are shown. The Pareto-optimal set is defined in an n_{of}-dimensional domain, so the Pareto-optimal values projected onto a bidimensional domain (f_i, f_j) appear in a picture like the one in Fig. 1.10. Thus, the addressed analyses can be performed on the basis of a statistical approach possibly combined with the plotting of graphs like those represented in Fig. 1.10. The *Spearman rank–order correlation coefficient* can be used to assess the above-introduced relationships and thus is particularly suited to discover significant relationships. This coefficient is able to identify monotonic non-linear relationships and thus is particularly suited to discover significant relationships that may suggest important engineering solutions. If

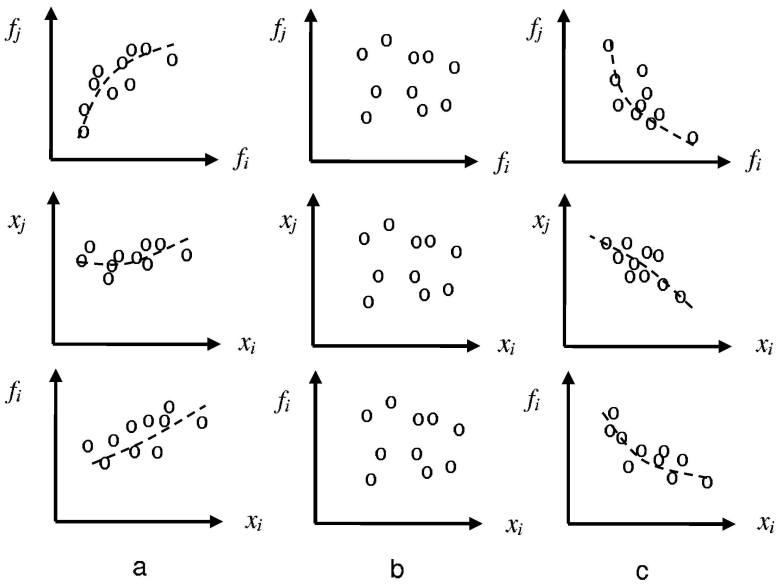


Fig. 1.10. Relationships between Pareto-optimal objective functions (f_i–f_j), between Pareto-optimal design variables (x_i–x_j), and between Pareto-optimal objective functions and Pareto-optimal design variable (f_i–x_i). The values can be either directly correlated (**a**), uncorrelated (**b**), or indirectly correlated (**c**)

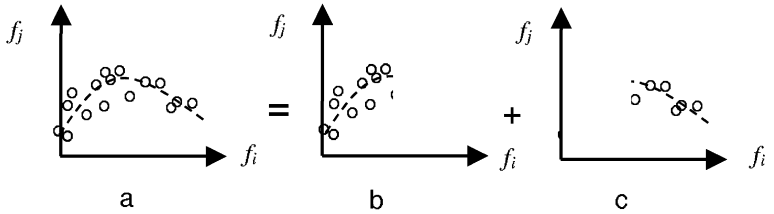


Fig. 1.11. Partition of the f_i domain to obtain two monotonic interpolation functions f_i. The Spearman rank–order correlation coefficient can be computed separately for cases (**b**) and (**c**)

the relationships are not monotonic a partition of the domain could be performed as shown in Fig. 1.11.

By compusting the Spearman rank–order correlation coefficient, very useful information on the system under consideration can be gained. In fact, a kind of sensitivity analysis focused on the Pareto-optimal solutions can be performed. In conventional sensitivity analysis, the relationships between objective functions and design variables are computed in the neighbourhood of one simple reference design solution. The analysis does not distinguish dominated from non-dominated solutions, i.e. the analysis is *not* restricted to Pareto-optimal solutions. So very poor information can be gained with conventional parameter sensitivity analysis. An analysis restricted to the very special Pareto-optimal values only may give significant hints to the designer. Actually, the relationships between Pareto-optimal objective functions and Pareto-optimal design variables can be highlighted.

2

Engineering Design and Optimal Design of Complex Mechanical Systems: Definitions

2.1 Engineering Design

In this section, only a very brief introduction on what could be meant by engineering design is given. The aim is just to introduce to the reader, in the subsequent part of the chapter, some definitions pertaining to the optimal design of complex mechanical systems.

The objective of engineering design, a major part of research and development (R&D) activity, is to produce drawings, specifications and other relevant information needed to manufacture products that meet customer requirements.

The main task of engineers is to apply their scientific and engineering knowledge to find one (or more) solutions to technical problems, and possibly to optimise those solutions within the requirements and constraints set by material, technological, economical, legal, environmental and human-related considerations. Design problems[1] become concrete tasks after the clarification and definition of the criteria which engineers have to adopt and to apply in order to create new technical products. The mental creation of a new product is the task of design or development engineers, whereas its physical realisation is the responsibility of manufacturing engineers.

2.1.1 Stages of the Design Process

According to [197] there are roughly four stages of the design process (see Fig. 2.1):

1. *Conceptual design* determines the principle of a solution. Conceptual design is that part of the design process in which, by the identification of the essential problems, by the establishment of function and by the search for

[1] A *design problem* (in engineering) may be defined as a set of requirements to solve a technical problem

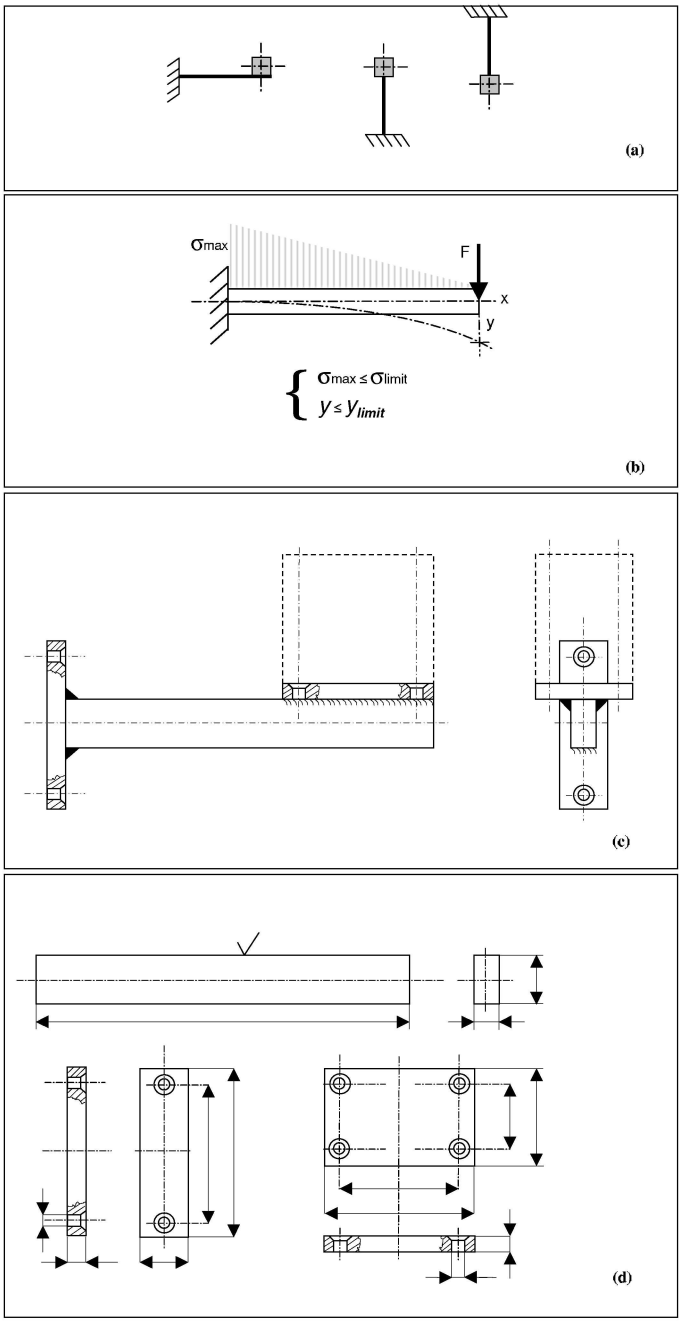


Fig. 2.1. Example of an engineering design process which develops in four stages. First stage **(a)** Three different *concept* design solutions to carry a body. Second stage **(b)** *Feasibility* study. Third stage **(c)** *Embodiment* of the design solution. Fourth stage **(d)** *Details* of the design solution

appropriate working principles and their combination, the basic solution is obtained through the elaboration of a solution principle.
2. *Preliminary design*. This stage (feasibility design) can be considered as a part of conceptual design. The preliminary layout is obtained by refining the conceptual designs and ranking them according to the design specifications, and choosing the best as the preliminary design.
3. *Embodiment design* is that part of the design process in which, starting from the working structure or concept of a technical product, the design is developed, in accordance with technical and economic criteria and in the light of further information, to the point where subsequent detail design can lead directly to production [197].
4. *Detail design* is that part of the design process which completes the embodiment of technical products with final instructions about the layout, forms, dimensions and surface properties of all individual components, the definitive selection of materials, operating procedures and costs [197].

The above classification of engineering design in four phases is not the unique one[2], anyway it is rather general, at least for what concerns complex mechanical systems.

On the more general level, design consists of a loop: product design ↔ manufacturing ↔ marketing improvement ↔ product design [7].

2.1.2 Creativity

Creativity is a very important component of some design phases. According to the level of creativity involved, engineering design can be classified into the following four categories [37] (see also Sect. 2.6):

> *Creative Design*: A priori plan for the solution of the problem does not exist. Design is an abstract decomposition of the problem into a set of levels that represents choices for the components of the problem. The key element in this design type is the transformation from the subconscious to conscious.
> *Innovative Design*: The decomposition of the problem is known, but the alternatives for each of its subparts do not exist and must be synthesised. Design might be an original or unique combination of existing components. It can be argued that a certain amount of creativity comes into play in the innovative design process.
> *Redesign*: An existing design is modified to meet required changes in the original functional requirements;
> *Routine Design*: A priori plan of the solution exists. The subparts and alternatives are known in advance, perhaps as a result of either a creative or innovative design process. Routine design involves finding the appropriate alternatives for each subpart that satisfy the given constraints.

[2] Actually it depends on the product developed

At the creative stage the design is very fuzzy. As it moves to routine design, it gets precise and predetermined.

2.2 Optimal Design of Complex Mechanical Systems

2.2.1 Fundamental Hypothesis

Referring to the classifications introduced in Sects. 2.1.1 and 2.1.2, let us assume that, given a design problem, a *conceptual* design solution has been found by a *creative* design activity.

At this stage of the design process the *optimal design of complex (mechanical) systems* can take place.

2.2.2 Single– and Multi-criteria Optimisation

When one thinks of optimisation in general, or about well–known optimisation problems, one usually thinks of problems of minimising or maximising a single quantity or objective. Indeed, with many optimisation problems, one begins with the implicit assumption that all candidate solutions can be ranked unambiguously according to their cost or utility.

The goal of the optimisation process is then well defined: one must find the highest ranked solution(s) possible.

But in real–world applications, problems with a single, well–defined objective to be optimised tend to be the exception rather than the rule. In engineering, (but also in finance, operational research, medicine, design, planning, scheduling, timetabling and many other domains), it is common to have problems with multiple requirements on system performances. Often, problems with a single objective may express what is most important or fundamental about a task in these domains, and they are mathematically simple to be treated, but they are not a faithful model of the real world. Unfortunately, in a problem with multiple objectives, it is generally impossible to obtain a total ordering (a ranking) of all of the alternative solutions, without invoking further rules or assumptions. This means that 'pure' optimisation, in which an unambiguously best solution is sought, may not be possible. This problem of ranking solutions arises whenever we must compare two solutions that offer a different compromise of the different performances. In this situation, the decision of the better solution may become somewhat subjective, or must rely on additional information, such as the 'importance' of each performance.

2.2.3 Multi-criteria Optimisation (MCO)

Multi-criteria optimisation (MCO) is the discipline that deals with the optimisation of (engineering design) problems in which many conflicting citeria have to be accounted for. Criteria can be expressed either mathematically or

not. In any case, the solutions cannot be ranked alone from their evaluation. Thus, in a broad sense, MCO really entails two different tasks, namely, search of a set of solutions and decision making.

- Search is needed to find solutions,
- Decision making is needed for ranking them.

This last task or activity of MCO is often called *multi-criteria decision making* (MCDM). MCDM refers to the methods for making choices between solutions that offer a different compromise of criteria. It is a scientific and mathematical discipline in itself, separate from search. MCDM essentially entails methods for scalarising the vector of objective functions (for the definition of 'objective function' see Sect. 2.5), so that a total ordering of solutions can be obtained, from which the 'best' can be chosen. Scalarising methods in turn involve techniques for equalising the ranges of different criteria, and for mathematically modeling the 'preferences' that (human) expert decision makers (DMs) have, considering compromise choices between solutions. For a concise but extensive overview of methods for performing MCDM, see [166].

Some more problems of conceptual design are described in [22] where MCDM is considered in a framework called multiple criteria decision aid (MCDA). Managers (i.e. designers acting as DMs) are often not satisfied with the optimal solution found in a bare MCDM process, and they require something more, an effective *aid*. Actually

- preferences are formed in a learning process;
- often there is a set of designers with different opinions and preferences;
- an optimal solution is created, not found;
- imprecision, interaction, flexibility, ... are needed;
- support for ambiguity handling and uncertainty;
- there is a need to move from a rational to pro-active approach.

2.2.4 multi-objective Optimisation (MOO)

Multi-objective optimisation (MOO) is the discipline that deals with the optimisation of (engineering design) problems in which many criteria are expressed mathematically.

2.2.5 Multi-objective Programming (MOP)

Multi-objective programming (MOP) is a theory belonging to multi-objective optimisation (Fig. 2.2). By means of MOP, an engineering design problem (in which the many criteria are expressed mathematically) can be formulated and solved. MOP will be introduced in Sect. 2.10, after the following definitions relating to complex systems, system models, objective functions, design variables, are given.

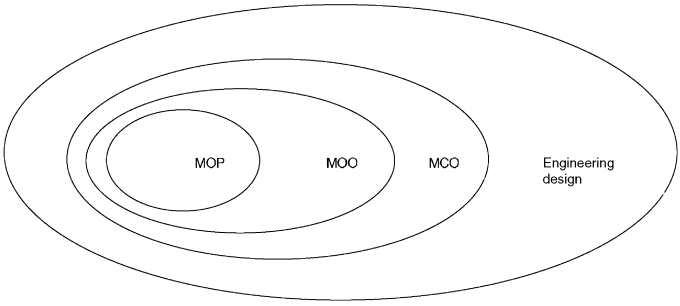


Fig. 2.2. Multi-objective programming (MOP) is a part of multi-objective optimisation (MOO) which is a part of multi-criteria optimisation, which can be exploited in the engineering design process

2.3 Complex Systems

Scientists and engineers have not yet accepted a common definition of what is meant by complexity. According to the definition given in [31], a *complex system* is taken to be the one whose properties are not fully explained by an understanding of its component parts. In a complex system multiple interactions between many different components exist.

The complexity of a system has also been related with the complexity of the process to define it. For instance, how much effort would be taken to solve a problem. According to designers' point of view, complexity should be defined relative to what they are trying to achieve [7]. Simply, when a designer has to define the values of four (or more) design variables and a preferred compromise has to be found among four (or more) conflicting system performances, then we can say that the designer has to deal with a complex system.

Complexity can be also defined as a measure of uncertainty. Uncertainty arises because of many factors: lack of knowledge about a system, the interaction among its multiple components.

Complex systems are often mathematical structures and processes that involve non-linearity. Complex systems are frequently composed by both many interacting objects and non-linear dynamical subsystems. The studies on complex systems include a wide variety of topics, such as

- cellular automata,
- chaos,
- evolutionary computation,
- fractals,
- genetic algorithms,
- artificial neural networks,
- parallel computing.

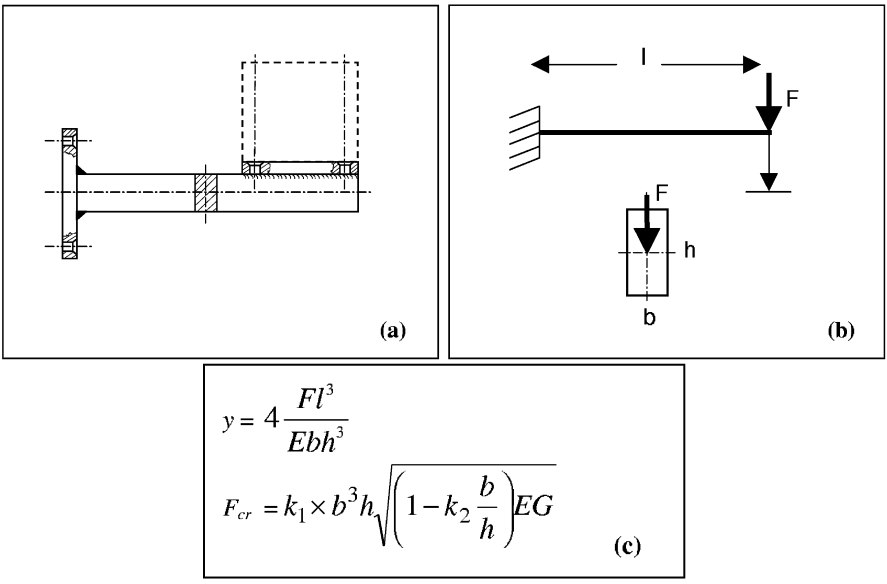


Fig. 2.3. Different models of an actual physical system. **(a)** Actual system. **(b)** Mechanical system model. **(c)** Mathematical system model (symbols and equations are reported in Chap. 1)

2.4 System Models

Given a *physical system* (an actual or virtual set of physical objects), we will assume that we will be always able to derive a *mathematical system model* (or, briefly, a *mathematical model*) capable to describe the behaviour of the original physical system[3].

If the mathematical system model is derived by applying physical laws, it will be called, in short, *physical model*. The mathematical system model can be composed of a set of algebraic or differential equations or a combination of them.

Often, given an actual mechanical system, it is convenient, in order to derive efficiently the corresponding physical model, to construct a *mechanical system model*. In this case we would have (see Fig. 2.3) the actual mechanical system, the mechanical system model, the mathematical system model (i.e. a physical model of the original actual system).

In a global approximation approach (see Chap. 4) the physical model is substituted by another mathematical model capable to compute the system physical behaviour very quickly.

[3] We will consider mostly physical systems, however all of the theoretical topics dealt with in the book may refer also to other systems (economic, social, etc.) that can be described by a mathematical model

The mathematical system model can be stochastic if some parameters and/or variables in it are defined by one or more stochastic processes.

2.5 System Performances, Criteria, Objective Functions

By definition, the physical behaviour of an actual physical system is described by a number of *system performances*.

System performances can be judged by a set of standards or principles strictly related to the design problem formulation. This set is called the set of *criteria*. Given a design problem, all the pertinent aspects must be represented by a complete set of *criteria*. The set of criteria must be minimal, there is no smaller set of criteria capable of representing all the aspects of the problem.

If the system performances can be quantified and expressed in mathematical form, then the mathematical expressions are called *objective functions*. Objective functions are expressed as computable functions of a set of parameters called design variables (see Sect. 2.6). Objective functions are the quantities that the designer wishes to optimise.

Objective functions will be identified as

$$\mathbf{f} = \{f_1(\mathbf{x}), \ldots, f_i(\mathbf{x}), \ldots, f_{n_{of}}(\mathbf{x})\}^T$$
$$\mathbf{x} = \{x_1, \ldots, x_j, \ldots, x_{n_{dv}}\}^T$$

where $\mathbf{x}$ is a vector of design variables (design variables are defined in Sect. 2.6). Objective functions measure the goodness of the system being designed. In almost all applications it is advisable to re–scale the values of the objective functions, that is, normalise the objective functions

$$f_{i\,norm} = \frac{f_i - f_{i\ \min}}{f_{i\ \max} - f_{i\ \min}} \quad i = 1, \ldots, n_{of} \tag{2.1}$$

With this transformation the range of each new objective function is $[0, 1]$. The normalised objective functions used in calculations need to be restored to original scale for analysing and displaying.

The need to take into account of subjective as well as objective aspects of the design is a complicated issue. For example the handling of a road vehicle is easily described in terms of linguistic descriptors rather than numerical values.

A common method to deal with subjective data is to convert it to objective values using a numerical scale.

Hierarchical decomposition can help (see Sect. 2.11) [163, 164]. Subjective evaluations can be handled by decomposing a complex quantitative attribute into lower level sub-attributes to make assessments more meaningful and manageable for the designer.

2.6 System Parameters, Design Variables

In general, a mathematical system model embodies a set of parameters. Usually designers are interested in finding one (or more) set of parameters that define one (or more) preferred configurations of the mathematical system. During the optimisation procedure the said set of parameters is changed to find their preferred values. So, during the optimisation process the mathematical system model parameters do vary. As parameters, by definition, do not vary, it is advisable to define these 'variable parameters' as *design variables*. Design variables are the variables that pertain to the mathematical system model during the optimisation process. Design variables can be grouped into a vector and will be identified as

$$\mathbf{x} = \{x_1, \ldots, x_i, \ldots, x_{n_dv}\}^T \tag{2.2}$$

Sometimes, the designer can choose to hold some design variable fixed in order to simplify the optimisation problem. These fixed quantities will be re–classified (obviously) as parameters.

The selection of the design variables is not unique, but it is very important to select these variables to be independent of each other.

Depending on how design variables are involved in the design process, the design can be reputed to be more or less creative. Referring to Sect. 2.1.2 the following classification of creativity in design is given [80]:

- if the number of the design variables and the ranges of values they can take remain fixed during design processing, then the process is routine design;
- if the number of the design variables remains fixed but their ranges change, then it is innovative design;
- if the number of design variables changes too, then it is creative design.

2.7 Constraints

Constraints are conditions which must occur for the design, in order to function as intended. When constraints can be expressed in mathematical form, they take the form of inequalities and/or equalities. Constraints will be identified as

$$\mathbf{g} = \{g_1(\mathbf{x}), \ldots, g_i(\mathbf{x}), \ldots, g_{n_c}(\mathbf{x})\}^T$$
$$g_i(\mathbf{x}) \leq 0 \quad (i = 1, \ldots, n_c)$$

where $\mathbf{x}$ is the vector of design variables[4]. Constraints on design variables are generally direct limitations on the variables themselves (side constraints, called lower bounds and upper bounds)

[4] This is a general definition of constraints. If a constraint should be expressed by $g_i(\mathbf{x}) \geq 0$ setting $g_i'(\mathbf{x}) = -g_i(\mathbf{x}) \leq 0$ would restore the inequality. Also, an equality constraint $g_i = 0$ can be expressed by using two additional inequality constraints $g_i'(\mathbf{x}) = g_(\mathbf{x}) \leq 0$ and $g_i''(\mathbf{x}) = -g_(\mathbf{x}) \leq 0$

$$x_{i_{low}} \leq x_i \leq x_{i_{up}} \quad (i = 1, \ldots, n_{dv}) \tag{2.3}$$

Constraints on system model behaviour are limits on the system output (objective function values). Sometimes these limits are given by the physical laws governing the system.

2.8 Space of Design Variables, Space of Objective Functions

We have already defined the design variables vector $\mathbf{x}$ and the objective functions vector $\mathbf{f}(\mathbf{x})$, we have to consider two spaces: the n_{dv}-dimensional space of the design variables ($\mathcal{X}$) and the n_{of}-dimensional space of the objective functions ($\mathcal{C}$).

2.9 Feasible Design Variables Domain, Design Solution

The whole set of constraints define the boundaries of the *feasible design variables domain*. Obviously, the feasible design variables domain is contained in the space of design variables

$$\mathcal{F} \subset \mathcal{X} \tag{2.4}$$

Often it is not easy to find $\mathcal{F}$. A *design solution* is represented by a point in the feasible design variable domain.

2.9.1 Conflict

Given a design problem, a number of criteria are formulated. As a rule in multicriteria optimisation (MCO) *conflicting criteria* are managed. A 'criterium' is a standard or a principle that is used to judge something. Criteria are considered conflicting when the satisfaction of one criterium does not imply the satisfaction of the others.

2.10 Multi-objective Programming (MOP)

2.10.1 Non-linear Programming (NLP) and Constrained Minimisation

Before treating multi-objective programming problems some relevant definitions for scalar optimisation will be introduced.

The aim of a scalar optimisation is to select the values of a vector of design variables $\mathbf{x}$ subject to the effect of some constraints $\mathbf{g}(\mathbf{x})$[5] in such a way that

[5] Considering an equality constraint $g_i(\mathbf{x}) = 0$ is equivalent to consider two inequality constraints $g_i(\mathbf{x}) \leq 0$ and $-g_i(\mathbf{x}) \leq 0$, so considering problems having only inequality constraints, equality constraints are implicitly considered

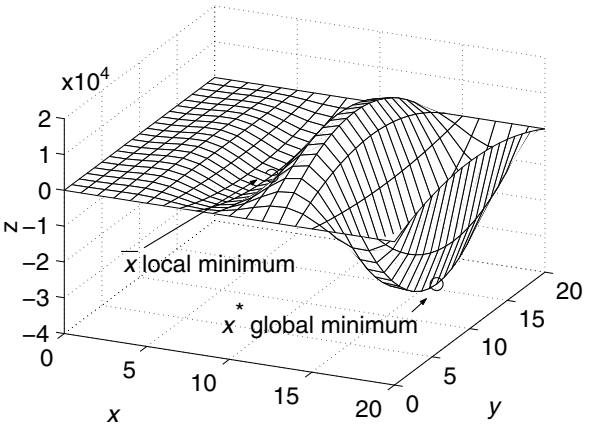


Fig. 2.4. Local and global minimum of the function: $z = 10(x^2 \sin(x)(-(y-10)^2 + 100))$, $0 < [x; y] < 20$

one single objective function $f(\mathbf{x})$ is minimised. In mathematical form a scalar optimisation can be written as

$$\begin{aligned} &\min_{\mathbf{x}\in\Re^{n_{dv}}} \; f(\mathbf{x}) \\ &\text{subject to:} \\ &g_i(\mathbf{x}) \leq 0, \quad i = 1, \ldots, n_c \\ &\mathbf{x}_{low} \leq \mathbf{x} \leq \mathbf{x}_{up} \quad (\mathbf{x} \in \mathcal{X}[\mathbf{x}_{low}, \mathbf{x}_{up}]) \end{aligned} \tag{2.5}$$

The subset $\mathcal{X} \subset \Re^{n_{dv}}$ is the design variable space of definition.

Let us notice that if an objective function $f(x)$ should be maximised (instead of minimised) the dual function $f'(x) = -f(x) \leq 0$ could be introduced and used.

Some important basic definitions, useful for the following explanation, are as follows.

Definition 2.1 (Global minimum). *A point* $\mathbf{x}^*$ *(Fig. 2.4) is a global minimum if* $f(\mathbf{x}^*) \leq f(\mathbf{x})\ \forall \mathbf{x} \in \mathcal{X}$.

Definition 2.2 (Local minimum). *A point* $\bar{\mathbf{x}}$ *(Fig. 2.4) is called local minimum if a* $\delta > 0$ *exists such that* $f(\bar{\mathbf{x}}) \leq f(\mathbf{x})\ \forall \mathbf{x} \in \mathcal{B}(\delta, \bar{\mathbf{x}}) \cap \mathcal{X}$, *where* $\mathcal{B}(\delta, \bar{\mathbf{x}}) = \{\mathbf{x} \in \mathcal{X} \mid \bar{\|}\mathbf{x} - \mathbf{x}\| < \delta\}$.

Definition 2.3 (Convexity).

i) A subset $\mathcal{X} \in \Re^{n_{dv}}$ *is convex if*

$$\alpha\mathbf{x}_1 + (1-\alpha)\mathbf{x}_2 \in \mathcal{X} \quad 0 \leq \alpha \leq 1 \quad \forall \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$$

ii) A real valued function $f(\mathbf{x})$ is convex on the convex subset $\mathcal{X}$ if

$$f(\alpha\mathbf{x}_1 + (1-\alpha)\mathbf{x}_2) \leq \alpha f(\mathbf{x}_1) + (1-\alpha)f(\mathbf{x}_2) \quad 0 \leq \alpha \leq 1 \quad \forall \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$$

The *optimality conditions* are the conditions that can help in finding the minimum of a function $f(\mathbf{x})$. The optimality conditions are important because they can be used to develop numerical methods (see [166]) and to check optimality of a given point (i.e. to check if that point is a minimum for the given function $f(\mathbf{x})$). Referring to the constrained problem (2.5) the optimality conditions can be given by the *Karush–Kuhn–Tucker* (KKT) *conditions*. (The optimality conditions can be expressed in several equivalent ways [166].)

A basic assumption for deriving the KKT conditions is that the minimum point is a regular point of the feasible set (see Sect. 2.9).

Definition 2.4 (Regular point). *A point* $\mathbf{x}$ *is called a* regular point *if the gradients of all the active*[6] *constraints at* $\mathbf{x}$ *are linearly independent.*

It is useful, before introducing the KKT conditions, to define the Lagrangian function, which for the problem (2.5) is defined as

$$L(\mathbf{x}, \boldsymbol{\eta}) = f(\mathbf{x}) + (\boldsymbol{\eta}^T \mathbf{g}(\mathbf{x})) \tag{2.6}$$

The KKT necessary conditions (first-order necessary conditions) are

Theorem 2.5 (KKT first-order necessary condition). *Let the objective function $f(\mathbf{x})$ and the constraint $g(\mathbf{x})$ functions of problem (2.5) be continuously differentiable at a vector $\mathbf{x}^* \in \mathcal{X}$. A necessary condition for $\mathbf{x}^*$ to be a local or a global minimum of $f(\mathbf{x})$ is*

$$\begin{aligned} &\nabla f(\mathbf{x}^*) + \sum_{i=1}^{n_c} \eta_i^* \nabla g_i(\mathbf{x}^*) = 0 \\ &\eta_i^* g_i(\mathbf{x}^*) = 0 \quad i = 1, \ldots, n_c \\ &\eta_i^* \geq 0 \quad i = 1, \ldots, n_c \end{aligned} \tag{2.7}$$

where η_i $(i = 1, \ldots, n_c)$ are the Lagrange multipliers (2.6).

The proof of the theorem can be found in [166]. The foregoing conditions are optimality necessary conditions for a constrained problem having regular functions.

If the objective function $f(\mathbf{x})$ and the constraint $g(\mathbf{x})$ functions are convex, then a local minimum is also a global minimum, and the KKT first-order conditions are necessary as well as sufficient [166].

The second-order necessary conditions follow.

[6] An inequality constraint $g_i(\mathbf{x}) \leq 0$ is active in $\mathbf{x}$ if $g_i(\mathbf{x}) = 0$

Theorem 2.6 (Second-order necessary condition). *Let* $\mathbf{x}^*$ *satisfy the first-order KKT conditions (2.7) for the problem (2.5). Let* $H(\mathbf{x}^*)$ *be the Hessian of the Lagrangian (2.6). Let* $\mathbf{d}$ *be a small non-zero feasible*[7] *change from* $\mathbf{x}^*$*. For* $j = 1, 2, \ldots, n_c$ *and for all* j *with* $g_j = 0$ *we have*

$$[\nabla g_j(\mathbf{x}^*)^T \mathbf{d}] = 0 \tag{2.8}$$

Defining $Q = [\mathbf{d}^T H(\mathbf{x}^*)\mathbf{d}]$*, if* $\mathbf{x}^*$ *is a local minimum point, it must be true that*

$$Q \geq 0 \tag{2.9}$$

for all $\mathbf{d}$ *satisfying Eq. (2.8).*

The proof of the theorem can be found in [166]. The inequality (2.9) is a necessary condition, a point that violates it cannot be a minimum point.

If in Eq. (2.9) only the inequality is valid then $\mathbf{x}^*$ is an isolated local minimum.

It must be stressed the all the gradient based optimisation methods [166] give only local minima unless some additional requirements are fulfilled such as convexity.

If the objective function $f(\mathbf{x})$ and the constraint $g(\mathbf{x})$ functions are convex, then the second-order conditions are necessary as well as sufficient [166].

2.10.2 Multi-objective Programming: Definition

The concepts introduced in Sect. 2.10.1 for scalar constrained optimisation will be extended here to vector (constrained) optimisation problems. A vector constrained optimisation problem can be formulated as a *multi-objective programming* (MOP) problem as follows. Find a vector of design variables (2.6) which satisfy a vector of constraints (2.7) and minimise a vector of objective functions (2.5). In mathematical form, given a design variable vector $\mathbf{x} \in \mathcal{X}$ which satisfies all the constraints $\mathbf{g(x)}$ and minimises the n_{of} components of the objective function vector $\mathbf{f(x)}$, the MOP problem can be written as[8]

$$\min_{\mathbf{x} \in \Re^{n_{dv}}} \mathbf{f}(\mathbf{x}) \quad \text{subject to}$$

$$\begin{aligned} & g_i(\mathbf{x}) \leq 0, \quad i = 1, \ldots, n_c \\ & \mathbf{x} \in \mathcal{X} \end{aligned} \tag{2.10}$$

where $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_i(\mathbf{x}), \ldots, f_k(\mathbf{x}))$ is the objective function vector, $\mathbf{x}$ the design variable vector, $\mathcal{X}$ the domain of definition of the variables (generally given as lower bound $\mathbf{x}_{low}$ and upper bound $\mathbf{x}_{up}$, $\mathbf{x}_{low} \leq \mathbf{x} \leq \mathbf{x}_{up}$),

[7] A small change $\mathbf{d}$ from $\mathbf{x}^*$ is called feasible if for any small $\mathbf{d}$, all the active constraints remain active

[8] Again, we will assume the vector function $\mathbf{f(x)}$ to be *minimized*. If $\mathbf{f(x)}$ were to be maximised, then one could retain the formulation presented in the following and consider $-\mathbf{f(x)}$

$\mathbf{g(x)}$ a vector of constraints. 'Minimise' means that we want to minimise all the objective functions *concurrently*. Obviously, we do not consider the trivial case when there is no conflict between objective functions (see Sect. 2.9.1). This means that the selected objective functions must be at least partially conflicting. In general, in MOP, it is not possible to find a single solution that is optimal for all the objective functions simultaneously.

The main target of MOP is to provide a rational approach to find optimal design solutions in presence of conflicting objective functions.

2.10.3 Pareto-optimal Solutions and Pareto-optimal Set

In a MOP problem (2.10) the concept of optimal solution (obvious for the scalar formulation) in not straightforward, it requires the special definition of *Pareto-optimal* solution[9].

Definition 2.7 (Pareto-optimal solution). *Given the multi-objective programming problem (2.10) with n_{dv} design variables and n_{of} objective functions, the Pareto-optimal (i.e. non-dominated, efficient or non-inferior) (vector) solution $\mathbf{x}_i$ satisfies the following conditions:*
$\nexists \mathbf{x}_j$:

$$\begin{cases} f_k(\mathbf{x}_j) \leq f_k(\mathbf{x}_i) \quad k = 1, 2, 3, \ldots, n_{of} \\ \exists l : \; f_l(\mathbf{x}_j) < f_l(\mathbf{x}_i) \end{cases} \tag{2.11}$$

If the above conditions hold, Pareto-optimal solutions will be denoted by $\mathbf{x}_i = \mathbf{x_i}^*$. Notice that the number of Pareto-optimal solutions approaches infinity. The whole set of Pareto-optimal solutions constitutes the *Pareto-optimal set*. Given a solution $\mathbf{x}_i^*$, if one tries to change it to improve one objective function, at least another objective function is worsened. For all non-Pareto-optimal solutions the value of at least one objective function f_l can be reduced without increasing the values of the other components. A typical characteristic of a MOP is the absence of a unique point that would optimise all the objective functions simultaneously.

Let us assume that a MOP problem with two objective functions and n_{dv} design variables has been formulated and solved, i.e. the Pareto-optimal set has been computed. Figure 2.5 refers to the two objective function space. The dashed area is the mapping of the design variable space $\mathcal{X}$ into the objective function space (see Sect. 2.8), the Pareto-optimal solutions lie on the curve *b*–*c*. The solution *a* is not obviously Pareto-optimal, all the solutions inside the area *abc* can reduce both the objective functions f_1 and f_2.

[9] The correct name Edgeworth–Pareto-optimal is not practically used in literature [238]. Francis Ysidro Edgeworth and Vilfredo Federico Damaso Pareto (see footnote 5) were the two economists who developed the theory of multicriteria optimisation. In 1881, Edgeworth [56] gave the definition of 'optimal solution' for a problem with two objective functions. Some years later, in 1906 Vilfredo Pareto [201] developed the definitions of Pareto-optimal solution for the *n*-dimensional case

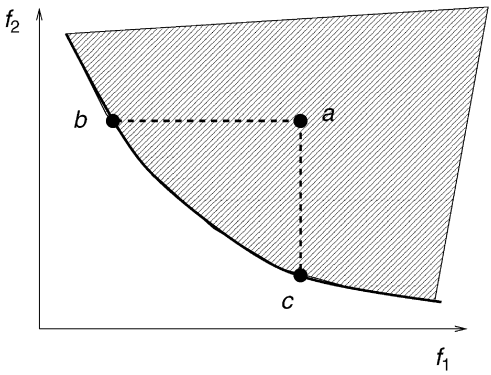


Fig. 2.5. Pareto-optimal set into the criterion space f_1, f_2

Any point in the Pareto-optimal set can be selected as an optimum solution. There is no generally accepted standard to judge (i.e. to rank) the solutions.

The designer has to choose a preferred solution among those (and only those) belonging to the Pareto-optimal set. It is advantageous to find a number of Pareto-optimal solutions. The designer acts as a decision maker (DM) and expresses preferences between two (or more) different Pareto-optimal solutions. The optimisation procedure generates information (Pareto-optimal solutions) for the designer who has to select a final solution. The designer will be interested in Pareto-optimal solutions only and the remaining solutions will be excluded. However, this cannot be the case if the problem has not been well formulated.

According to Definition 2.7 selecting Pareto-optimal solutions requires a compromise or a trade–off. In principle it is not always necessary to resort to a trade–off to improve the design process. Using Axiomatic Design [7] to formulate and solve a conceptual design problem (2.1.1) may produce better results than simply optimising an early tentative design solution.

Definition 2.7 introduces *global Pareto-optimality*. The concept of local Pareto-optimality is shown in Fig. 2.6. A solution $\mathbf{x}_i$ is locally Pareto-optimal if there exists $\delta > 0$ such that $\mathbf{x}_i$ is Pareto-optimal in the circle centred in $\mathbf{x}_i$ with radius δ. Any global Pareto-optimal solution is locally Pareto-optimal. The converse is valid for convex multi-objective problems.

The Pareto-optimal set always belongs to the boundary of the mapping of the design space into the space of objective functions ($\mathcal{C} \in \Re^{n_{of}}$). There is an infinity of Pareto-optimal solutions. It is easy to guarantee their existence [166], since all that is required is that the mapping of the design variable space into the objective function space ($\mathcal{C} \in \Re^{n_{of}}$) is lower bounded and closed. The Pareto-optimal set may be connected or have gaps (see Fig. 2.6 referring to a two-dimensional criterion space). We can guarantee that the Pareto-optimal set is connected when $\mathcal{C}$ is lower convex and closed [17, 58, 257].

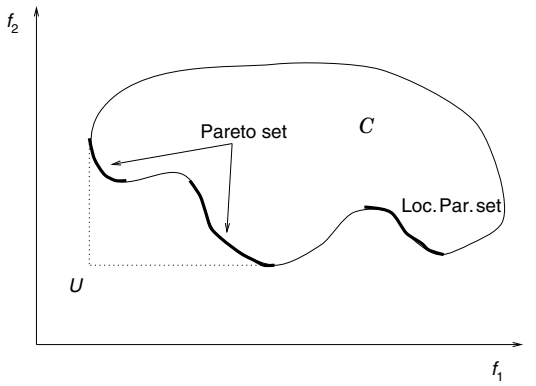


Fig. 2.6. Local and (global) Pareto-optimal sets in a two design variable space f_1, f_2

The necessary conditions for determining Pareto-optimal solutions are summarised in the following theorem (necessary conditions).

Theorem 2.8 (Pareto-optimal necessary conditions). *Let us consider the problem (2.10) if* $\mathbf{x}^* \in \mathcal{X}$ *is a regular local Pareto-optimal solution for the vector function* $\mathbf{f}(\mathbf{x})$ *then there exist multipliers* $\boldsymbol{\lambda} = (\lambda_1 \lambda_2 \cdots \lambda_{n_{of}})^T$ *and* $\boldsymbol{\eta} = (\eta_1 \eta_2 \cdots \eta_{n_c})^T$ *for which* $(\boldsymbol{\lambda}, \boldsymbol{\eta}) \neq \mathbf{(0, 0)}$ *such that*

$$\begin{aligned} &\sum_{k=1}^{n_{of}} \lambda_k \nabla f_k(\mathbf{x}^*) + \sum_{i=1}^{n_c} \eta_i \nabla g_i(\mathbf{x}^*) = 0 \\ &\eta_i g_i(\mathbf{x}^*) = 0 \quad i = 1, \ldots, n_c \\ &\boldsymbol{\eta} \geq \mathbf{0} \\ &\boldsymbol{\lambda} \geq \mathbf{0} \end{aligned} \tag{2.12}$$

The proof of the theorem can be found in [166]. The Lagrangian function in this case is

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\eta}) = \boldsymbol{\lambda}^T \mathbf{f}(\mathbf{x}) + \boldsymbol{\eta}^T \mathbf{g}(\mathbf{x}) \tag{2.13}$$

If we consider $\boldsymbol{\lambda}^T \mathbf{f}(\mathbf{x})$ where $\boldsymbol{\lambda} \geq \mathbf{0}$ instead of the single objective function $f(\mathbf{x})$ the Pareto-optimal necessary conditions are the same as the KKT necessary conditions for the general non-linear programming problems (2.12). This *scalarisation* process, as will be shown in the subsequent chapters, is generally suggested as a way to solve MOP problem.

The idea of minimising one objective function subject to constraints on the other objective functions has been formulated to produce a necessary and sufficient condition for Pareto-optimal solutions by Schmitendorf and Moriarty [223] .

Theorem 2.9 (Pareto-optimal Theorem). *A point* $\mathbf{y}^* \in \mathcal{C}$ *is Pareto-optimal if and only if for each* $j \in \{1, \ldots, n_{of}\}$

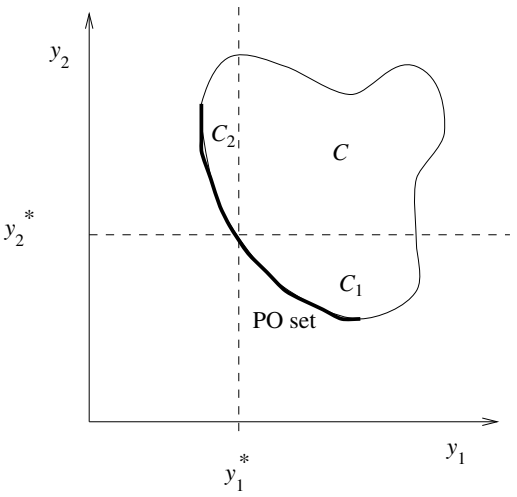


Fig. 2.7. Schmitendorf's theorem: two objective function space f_1, f_2

$$y_j^* \leq y_j \; \forall \mathbf{y} \in \mathcal{C}_j \tag{2.14}$$

where $\mathcal{C}_j = \{\mathbf{y} \in \mathcal{C} \; : \; y_i \leq y_i^* \; i = 1, \ldots, n_{of}; \; i \neq j\}$.

The proof of the theorem can be found in [166]. The theorem is illustrated in Fig. 2.7. The point with components y_1^* and y_2^* satisfies $y_1^* \leq y_1 \in \mathcal{C}_1$ and $y_2^* \leq y_2 \in \mathcal{C}_2$.

Also in the case of scalar optimisation (Sect. 2.10.1) second-order Pareto-optimality conditions can be expressed.

Theorem 2.10 (Second-order necessary condition). *Let the objective functions of the problem (2.10) be twice continuously differentiable at a design variable vector* $\mathbf{x}^* \in \mathcal{X}$. *In* $\mathbf{x}^*$, *let the active constraints be linearly independent. A necessary condition for* $\mathbf{x}^*$ *to be Pareto-optimal is that vectors exist such that* $0 \leq \boldsymbol{\lambda} \in \Re^{n_{of}}$ $(\boldsymbol{\lambda} \neq 0)$ *and* $\boldsymbol{\eta} \leq \in \Re^{n_c}$ *such that the first-order Pareto-optimal necessary condition in* $\mathbf{x}^*$ *is valid and, considering* $H(\mathbf{x}^*)$ *the Hessian of the Lagrange function (2.13), the following inequality is also valid*

$$\mathbf{d}^T H(\mathbf{x}^*)\mathbf{d} \geq 0$$

for all $\mathbf{d} \in \{\mathbf{0} \neq \mathbf{d} \in \Re^{n_{dv}} : \nabla f_i(\mathbf{x}^*)^T \leq 0$ *for all* $i = 1, \ldots, n_{of}$, $\nabla g_i(\mathbf{x}^*)^T \mathbf{d} = 0$ *for all* i *where* g_i *is an active constraint* $\}$.

The proof of the theorem can be found in [166].

2.10.4 Ideal and Nadir Design Solutions

The *ideal design variable vector* can be obtained by minimising each of the objective functions f_i individually, subject to the constraints. The ideal design

variable vector is represented by a point (also called 'utopia point' [166]) in the objective function space (see Fig. 1.4). The ideal design variable vector corresponds generally to a not-feasible design solution. It is only a reference design variable vector that constitutes the lower bounds of the Pareto-optimal set for each objective function.

An important reference square matrix is the one whose rows are respectively composed of objective function vectors which are obtained by minimising one single objective function at a time. The first row contains the objective function vector in which $f_1(\mathbf{x})$ is minimised. The second row contains the objective function vector in which $f_2(\mathbf{x})$ is minimised, and so on for the n_{of}.

$$\begin{pmatrix} f_{1\ \min}(\mathbf{x}_{f_{1\ \min}}) & f_2(\mathbf{x}_{f_{1\ \min}}) & \cdots & f_{n_{of}}(\mathbf{x}_{f_{1\ \min}}) \\ f_1(\mathbf{x}_{f_{2\ \min}}) & f_{2\ \min}(\mathbf{x}_{f_{2\ \min}}) & \cdots & f_{n_{of}}(\mathbf{x}_{f_{2\ \min}}) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(\mathbf{x}_{f_{n_{of}\ \min}}) & f_2(\mathbf{x}_{f_{n_{of}\ mim}}) & \cdots & f_{n_{of}\ \min}(\mathbf{x}_{f_{n_{of}\ min}}) \end{pmatrix} \tag{2.15}$$

Obviously the ideal objective vector is the main diagonal of the matrix (2.15). The maximum value of each column i of the matrix is a rough estimate of the upper bound of the objective function f_i over the Pareto-optimal set. The upper bounds of the Pareto-optimal set constitute the *nadir design variable vector*.

The row vectors in the matrix (2.15) are Pareto-optimal if they are unique.

2.10.5 Related Concepts

Weak Pareto-optimality

Weak Pareto-optimality: a vector is *weakly Pareto-optimal* if there does not exist any other vector for which all the components are better (Fig. 2.8).

The Pareto-optimal set is a subset of the weak Pareto-optimal set.

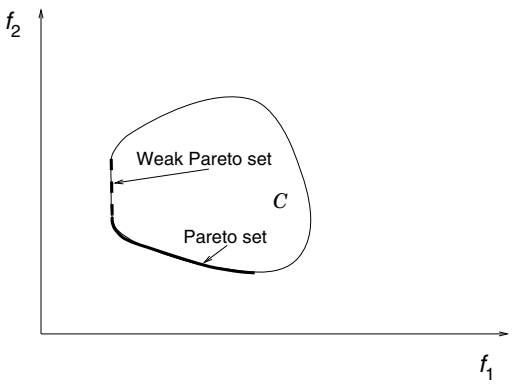


Fig. 2.8. Weak Pareto-optimal set in a two-objective-function space f_1, f_2

Proper Pareto-optimality

Let $\mathbf{f}(\mathbf{x}_1)$ and $\mathbf{f}(\mathbf{x}_2)$ be two objective function vectors evaluated, respectively, at two given design variable vectors $\mathbf{x}_1$ and $\mathbf{x}_2$. We denote as *trade-off* the ratio

$$\Lambda_{ij}(\mathbf{x}_1, \mathbf{x}_2) = \frac{f_i(\mathbf{x}_1) - f_i(\mathbf{x}_2)}{f_j(\mathbf{x}_1) - f_j(\mathbf{x}_2)} \tag{2.16}$$

If both $\mathbf{x}_1$ and $\mathbf{x}_2$ belong to the Pareto-optimal set, the *trade-off* is the ratio between the increment of one objective function $\mathbf{f}_i(\mathbf{x})$ and the decrease of other objective functions $\mathbf{f}_j(\mathbf{x})$, $j = 1, .., n_{of} \quad i \neq j$.

Proper Pareto-optimal solutions are defined with reference to trade-off. A Pareto-optimal solution with very high or very low trade-offs is not properly Pareto-optimal.

2.10.6 Basic Problems and Capabilities of Multi-objective Optimisation

Some of the basic problems of MOO (and thus of MOP) are briefly explained below.

- There are objective functions and there are constraints. The difference between them can be fuzzy and some of them will move from objective functions to constraints or vice versa. Some constraints are hard, some not; some will change or disappear while others may be introduced as the problem knowledge base expands.
- In many cases the design variable ranges are also fuzzy and flexible and there is a requirement for exploration outside of the default regions. The reason is that the real bounds and limits are not always known from the beginning.
- The solution of the design should contain both optimal solutions and suggestions of extending ranges and/or inclusion/removals of constraints.
- A set of results is required which the engineer can analyse. (That means that the engineer should be able to input those results to some other programs or to consult some database or persons for different aspects of given solutions).
- The designer should not be confused by the number of parameters and by the possibilities optimisation offers, cognitive overload must be avoided.

The problems of conceptual design relate to the fuzzy nature of initial design concepts. Computers should be able to help in exploration of those variants while suggesting some others as well.

An important capability that MOO offers is that of comparing different concept design solutions. Often engineers are in trouble if a well-defined comparison between two systems is requested. Actually they sometimes fall into the fatal mistake to consider – and compare – one non-optimal system with another which is optimal. The result of the comparison is thus influenced by

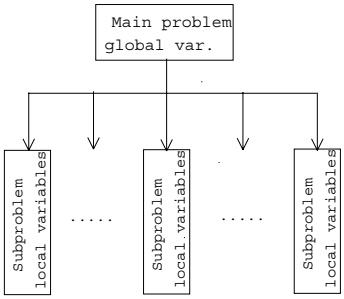


Fig. 2.9. Hierarchical strategy

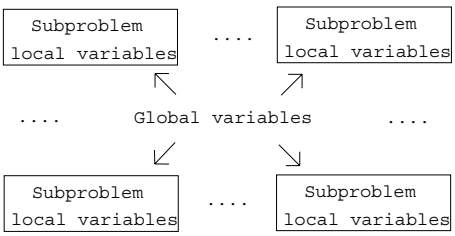


Fig. 2.10. Non-hierarchical strategy

the non-optimal choice of one system. The conclusion can be favourable to the optimal system, just because the first non-optimal system was considered. MOO allows to take into account *only* optimal systems, thus the comparison is correctly 'restricted' to optimal systems only. This allows to actually choose the best solution after a comprehensive optimal design process[10].

2.11 Decomposition of Design Problems

When the design problem is extremely complex the required computational power can be unacceptable. If this is the case the problem can be subdivided into smaller subproblems.

The *optimal-model–based decomposition method* [163, 164, 200] can be applied. The design variables have to be subdivided into local (relative to the subproblem only) and global (relative to the whole problem). The local variables are assigned in order to define a limited number of subproblems, related in a hierarchical or non-hierarchical manner (see Figs. 2.9, 2.10).

A hierarchical decomposition [200] (Fig. 2.9) usually leads to separate optimisations for each subproblem. The master problem is solved for the linking

[10] For example, if two concept designs of, respectively, two cars (one with front wheel drive transmission and the other with rear wheel drive transmission) should be compared, we should take into account *only* the Pareto-optimal design solutions representative of the two different concept designs

variables that are considered as parameters for the subproblems. Subproblems may be recursively partitioned to generate a multilevel hierarchy.

If a non-hierarchical decomposition (Fig. 2.10) subspace optimisation takes place in each subproblem, then bidirectional intervention between subproblems takes place, global sensitivity and model approximations provide to quantify influences of one subproblem to another.

These decompositions may benefit from an optimal decomposition that identifies weakly connected structures in the system model [163, 164].

3

Multi-objective Optimisation

In this chapter the main issues of multi-objective optimisation (MOO) will be introduced with particular reference to multi-objective programming (MOP). Particular attention will be devoted to those optimisation problems that can be fully defined mathematically as shown in Sect. 2.10. Many of the issues presented in the chapter can be extended to MOO.

3.1 Methods to Solve Multi-objective Programming (MOP) Problems

The solution of multi-objective programming (MOP) problems can be classified according to how the designer (i.e. the decision maker, DM) manages preference information, that is how the designer (searches for and) chooses the optimal solutions [226].

There are three general approaches to solve multi-objective programming (MOP) problems (Fig. 3.1). They are

- Pareto-optimal set generation methods
- Preference-based methods
- Interactive methods

In *Pareto-optimal set generation methods*, the decision maker chooses one of the alternative optimal solutions within the Pareto-optimal set after the Pareto-optimal set has been generated. In the *preference-based methods* the preferences of the decision maker are taken into consideration before the optimisation process. In the *interactive methods* the preferences are considered as the optimisation process goes on.

Of course, several variants to these three approaches exist. For sake of space in this chapter we will not describe every approach to the same extent. We will present in a detailed way the approach (Pareto-optimal set generation method) that has been employed for the engineering applications described in the Part II of the book.

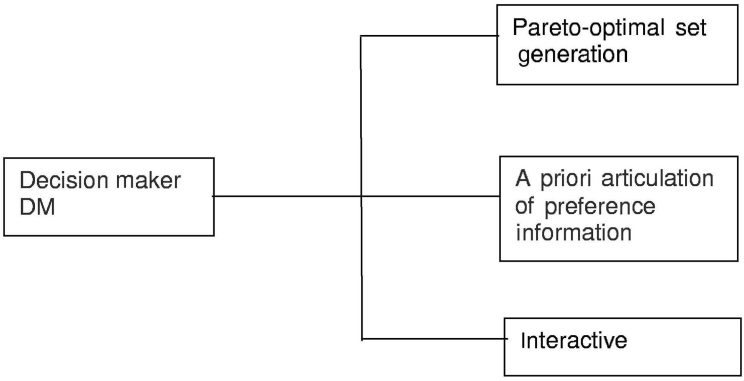


Fig. 3.1. Classification of approaches for solving MOP problems

3.2 Pareto-optimal Set Generation Methods

In many applications (e.g. all of the applications presented in Part II of this book) multi-objective programming (MOP) process consists of a number of steps. A generic MOP procedure is illustrated in the flowchart shown in Fig. 3.2.

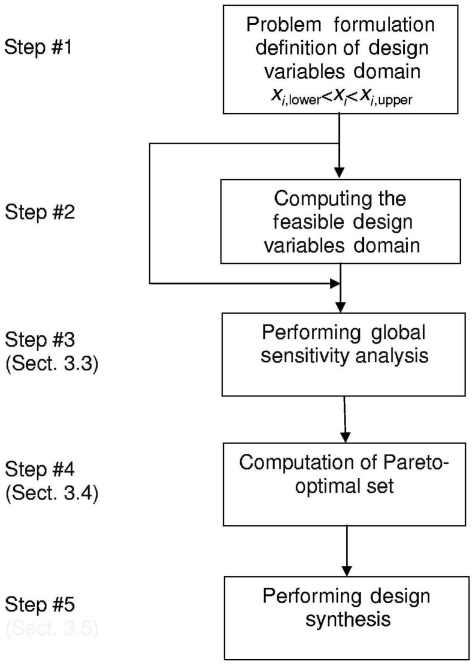


Fig. 3.2. Generic MOO/MOP procedure

In the first step, let us assume that the problem has been formulated according to the definitions of MOP given in Chap. 2. In particular, let us assume that the design variable domain has been defined after a proper analysis which depends on the particular problem under consideration[1].

The second step might require the computation of the feasible design variables domain (see Sect. 2.8). The feasible design variables domain comprises all the design variables vectors which give solutions meeting all the design constraints. In general, the designer has to take into account constraints both on design variables and on objective functions. Obviously, the feasible solutions set includes the subset constituted by the Pareto-optimal solutions. *Low discrepancy sequences* (see Sect. 3.4.2) can be employed for determining the feasible solutions set.

The third step will be treated in-depth in this chapter. Global sensitivity analysis (GSA, see Sect. 3.3) can be used to get an insight into the relationship between objective functions and design variables. Each design variable may vary within its feasible set, i.e. within ranges that are defined by upper and lower bound values meeting all the design constraints. These ranges are usually wide. Within one generic range the relationship between the design variable and a generic objective function is in general non-linear and can be either monotonic or non-monotonic. In any case there are proper techniques (described in this chapter) that are capable to return information on how much a design variable influences an objective function.

GSA can also be used to assess if there is a direct or an indirect relationship between objective functions. This is very useful in order to reduce, whenever possible, the number of objective functions that have to be taken into account in the subsequent MOP steps: If strongly correlated objective functions are found, the existence of a redundancy is discovered and the number of objective functions to be considered can be reduced with respect to the original number of objective functions introduced in the first step.

In the fourth step the Pareto-optimal set is computed by resorting to many different methods for solving a multi-objective programming problem (see Sect. 3.4).

A correlation analysis can be performed between the solutions belonging to the Pareto-optimal set. This can be done by resorting again to the GSA techniques already mentioned (see Sect. 3.3). The output of this correlation analysis gives the designer an insight to the optimisation problem under consideration.

In the fifth step, given the set of Pareto-optimal solutions, the decision maker (designer) has to select the final design solution (see Sect. 3.5).

[1]Let us emphasise that in practical cases it may be impossible to give a correct formulation of the problem before it is solved. In other words, an iterative procedure between modeling and optimisation is needed

3.3 Global Sensitivity Analysis

Global sensitivity analysis (GSA) allows to evaluate how much the variation of a generic design variable affects the variation of each objective function.

GSA differs from the well-known standard sensitivity analysis. Actually, standard sensitivity analysis refers to the variation of a generic objective function due to small variations of the parameters of the system (partial derivatives of the objective functions with respect to the system parameters). This gives the designer only a local and a very limited insight on the optimisation problem.

GSA describes the behaviour of the system when parameters are varied within broad ranges (e.g. within the whole feasible design variables domain). GSA provides measures on how much each design variable contributes to each objective function. This analysis can help to identify the key design variables in a particular optimisation problem. Of course also the less influential design variables are identified, that is, those design variables for which a large variation does not influence remarkably a generic objective function.

There are many kinds of GSAs.

A linear regression analysis is introduced in Sect. 3.3.1 and can be used to identify linear correlations between design variables and objective functions. This approach gives only a limited insight to the designer.

Variance-based methods (introduced in Sect. 3.3.2) are the most effective ones in giving, quantitatively, the information about how much a generic design variable influences an objective function and thus how much it is important to the designer. Unfortunately, variance-based methods cannot be exploited if the system model requires a lot of time for returning the results of the simulations.

If the monotonicity of a generic objective function is properly checked and confidently assessed, a standardised rank correlation analysis (introduced in Sect. 3.3.3) can be employed successfully.

By means of an artificial neural network (ANN, see Sect. 3.3.4) it is possible to perform a GSA. The interconnection weights of the trained Neural Network can be considered in order to get an insight into the influence of each design variable on the objective functions.

3.3.1 Global Sensitivity Analysis Based on Linear Regression Methods

GSA based on linear regression can be used to identify linear correlations between a generic design variable and a generic objective function (shortly we will write in the following 'between design variables and objective functions').

The method assumes that design variables x_i, $i = 1, \ldots, n_{dv}$ and objective functions f_j, $j = 1, \ldots, n_{of}$ are linearly correlated. By considering each objective function we have

$$f_j = \beta_{0j} + \sum_{i=1}^{n_{dv}} \beta_{ij} x_i \tag{3.1}$$

β_{ij} are the coefficients of the linear regression. The accuracy of this model can be measured by the coefficient of determination (R^2) [210].

After a proper normalisation of the design variables[2], the β_i can be used to rank the relative importance of each design variable x_i.

Pearson Correlation Coefficient

The linear relationship between two variables can be measured by using the Pearson correlation coefficient (PCC) [171].

The correlation coefficient is always between -1 and 1. If the sampled design variables and objective functions can be plotted on a straight line with positive slope the correlation coefficient P_{CC} assumes the value 1. If the slope is negative $P_{CC} = -1$. A lack of linear correlation corresponds to $P_{CC} = 0$.

Given a bivariate data set of size N,

$$\{(x_{11}; x_{21}); (x_{12}; x_{22}); \cdots; (x_{1N}; x_{2N})\}$$

the sample covariance $\mathrm{Cov}(x_1; x_2)$ is defined as

$$\mathrm{Cov}(x_1; x_2) = \frac{1}{N-1} \sum_{i=1}^{N} (x_{1i} - \bar{x}_1)(x_{2i} - \bar{x}_2)$$

where $\bar{x}$ is the mean value of x in the data set.

The correlation coefficient P_{CC} is defined by

$$P_{CC} = \frac{\mathrm{Cov}(x_1; x_2)}{\Sigma_{x_1} \Sigma_{x_2}}$$

where Σ_{x_1} is the sample standard deviation of $x_{11}, \ldots, x_{1N}$, i.e.

$$\Sigma_{x_1} = \sqrt{\frac{\sum_{i=1}^{N} (x_{1i} - \bar{x}_1)^2}{N-1}}$$

[2]The normalisation of a design variable can be made as follows. Let x_i, $i = 1, \ldots, n_{dv}$, $x_{imin} \leq x_i \leq x_{imax}$ be the set of design variables the scaling (or normalisation) of x_i is

$$x_{inorm} = \frac{x_i - x_{imin}}{x_{imax} - x_{imin}} \quad i = 1, \ldots, n_{of}$$

With this transformation the range of each new design variable $x_{i\,norm}$ is $[0, 1]$

3.3.2 Sobol Method

A fully quantitative variance-based method is the Sobol method [233]. It has conceptual similarities with another GSA method, the Fourier amplitude sensitivity test which is not treated in this book (for details on this topic see [132, 156]).

Let us assume that the relationship between design variables $\mathbf{x}=\{x_1, x_2, \ldots, x_{n_{dv}}\}^T$ and objective functions can be modeled by a set of functions

$$\mathbf{f}(\mathbf{x}) \tag{3.2}$$

Generally, the function $\mathbf{f}$ is not known explicitly, but it can be computed numerically.

The n_{dv} design variables are defined to vary in a n_{dv}-dimensional unit hypercube (this is always possible after a proper normalisation, see footnote 2)

$$\mathcal{O}^{n_{dv}} = \{\mathbf{x} \,:\, 0 \leq x_i \leq 1;\; i = 1, 2, \ldots, n_{dv}\} \tag{3.3}$$

Considering the GSA of a single objective function $f(\mathbf{x})$, the Sobol method is based on the partitioning of each objective function into summands of functions of the design variables x_i, of increasing dimensionality

$$f(\mathbf{x}) = f_0 + \sum_{i=1}^{n_{dv}} f_i(x_i) + \sum_{1 \leq i < j \leq n_{dv}} f_{ij}(x_i, x_j) + \ldots + f_{1,2,\ldots,n_{dv}}(x_1, x_2, \ldots, x_{n_{dv}}) \tag{3.4}$$

where f_0 is a constant and $f_i(x_i)$, $f_{ij}(x_i, x_j)$, $f_{1,2,\ldots,n_{dv}}(x_1, x_2, \ldots, x_{n_{dv}})$ are existing functions not known a priori.

The variance $D = \mathrm{Var}\,(f(\mathbf{x}))$ [219, 220] can be partitioned as well as the function $f(\mathbf{x})$ into partial variances (all the summands are orthogonal)

$$D = \sum_{i=1}^{n_{dv}} D_i + \sum_{1 \leq i < j \leq n_{dv}} D_{ij} + \ldots + D_{1,2,\ldots,n_{dv}} \tag{3.5}$$

where $D_{i_1,i_2,\ldots,i_t} = \int_0^1 \ldots \int_0^1 f^2_{i_1 \ldots i_t} \mathrm{d}x_{i_1} \ldots \mathrm{d}x_{i_t}$. $f_{i_1 \ldots i_t}$ denotes a generic term of the series expansion Eq. (3.4). The sensitivity index for a group of indices $\{i_1, i_2, \ldots, i_t\}$ with $1 \leq i_1 < \ldots < i_t \leq n_{dv}$ and $t = 1, 2, \ldots, n_{dv}$ is

$$S_{i_1,\ldots,i_t} = D_{i_1,\ldots,i_t}/D \tag{3.6}$$

The total sensitivity index (T_S) measures the total effect (including all the interaction effects) of design variable x_i and it is defined as the sum of all the sensitivity indices involving x_i:

$$T_S(i) = \sum S_{i_1,\ldots,i_t} \tag{3.7}$$

$T_S(i)$ can be also written as

$$T_S(i) = 1 - S_{ci} \tag{3.8}$$

where S_{ci} is the sum of all the $S_{i_1,\ldots,i_t}$ terms not involving the index i. Consequently, $T_S(i)$ equals the total effect of variable x_i, which includes the fraction accounted for any combination of x_i with the remaining variables.

The f_0 in Eq. (3.4) can be estimated by

$$\hat{f}_0 = 1/N \sum_{m=1}^{N} f(\mathbf{x}_m) \tag{3.9}$$

where N denotes the sample size. The total output variance D can be estimated by

$$\hat{D} = 1/N \sum_{m=1}^{N} f^2(\mathbf{x}_m) - \hat{f}_0^2 \tag{3.10}$$

The term $\hat{D}_i$ is estimated by

$$\hat{D}_i = 1/N \sum_{m=1}^{N} f(\mathbf{x}_m) f(\hat{\mathbf{x}}_m) - \hat{f}_0^2 \tag{3.11}$$

where $\mathbf{x}_m$ indicates that all the x_j $(j = 1, \ldots, n_{dv})$ are sampled a first time, while $\hat{\mathbf{x}}_m$ is estimated by re–sampling all the x_j except x_i.

One separate sample (of size N) is needed to estimate each of the $S_{i_1,\ldots,i_t}$. Being the number of terms in the expansion Eq. (3.5) $2^{n_{dv}} - 1$, $N \times 2^{n_{dv}}$ model evaluations have to be computed. Problems with a large number of design variables require a huge number of model evaluations. For this reason the total sensitivity indices $T_S(i)$ are used instead of the sensitivity indices S. The computation of $T_S(i)$ can be completed with a reduced number of model evaluations. More precisely, the number of model evaluations is $N \times (n_{dv} + 1)$, i.e. one sample for the estimate of f_0 plus one sample for each design variable.

The estimates are computed with large sample sizes (N) by using low discrepancy sequences (see Sect. 3.4.2).

Both $S_{i_1,\ldots,i_t}$ and $T_S(i)$ can be computed on ranks (see Sect. 3.3.3), gaining in robustness [219, 220].

3.3.3 Spearman Rank Correlation Coefficient

An efficient quantitative global sensitivity analysis (GSA) method is based on a rank regression analysis.

If the monotonicity of objective functions $\mathbf{f}(\mathbf{x})$ with respect to design variables $\mathbf{x}$ is properly checked and confidently assessed, the 'standardised rank regression analysis' based on the computation of the Spearman rank–order correlation coefficient r_s [210] can be employed successfully [144, 182]. To compute r_s the values of $f_i(\mathbf{x})$ and $\mathbf{x}$ are merely replaced with their corresponding ranks. The rank transformation can cope with non-linear

relationship between $f_i(\mathbf{x})$ and x_j and allows a robust estimation of the global sensitivity, but it underestimates higher order non-linearities (i.e. the effects of strong non-linearities are reported just as soft non-linearities, actually r_s reports only if there is a correlation). Another advantage of using r_s for GSA is that the coefficient of determination R^2 can be used to check the confidence of r_s, i.e. the validity of the estimated influence of the design variables on a generic objective function. A low value of R^2 indicates that r_s does not give a reliable estimation of the non-linear correlation (a non-monotonic relationship is occurring). In case a non-monotonic relationship does exist, the GSA can be performed anyway after a proper partition of the design variables space (see Sect. 1.2.2). An empirical method [95] is the partition of the design variables space into n_p sub–domains and the evaluation of the global sensitivity in each sub–domain.

The GSA can be performed not only between a generic objective function and a generic design variable, but also among a generic objective function and another generic objective function, in order to discover the relationships between objective functions.

An objective function is redundant if it does not affect the Pareto-optimal set. Redundancy check would be important to eliminate as many objective functions as possible for simplifying the MOO problem.

Let us now explain how r_s can be computed. Given two variables y_1 and y_2, the observed values of y_1 are ranked and substituted by their ranks. The same is done for y_2. For example, for a sample of five values per variable we have

y_1	y_2
20.4	9.2
19.7	8.9
21.8	11.4
20.1	9.4
20.7	10.3

So, substituting the variable values with the respective ranks

$\text{rank}(y_1)$	$\text{rank}(y_2)$
3	2
1	1
5	5
2	3
4	4

Let $\text{rank}(y_{1i})$ be the rank of y_{1i} and let $\text{rank}(y_{2i})$ be the rank of y_{2i}. Defining $\overline{\text{rank}}(\cdot) = \frac{1}{N}\sum_{i=1}^{N}\text{rank}(\cdot)$ and using these values, we compute the

Spearman correlation coefficient (r_s) by

$$r_s = \frac{\sum_{i=1}^{N}[\text{rank}(y_{1i}) - \overline{\text{rank}}(y_1)][\text{rank}(y_{2i}) - \overline{\text{rank}}(y_2)]}{\sqrt{\sum_{i=1}^{N}[\text{rank}(y_{1i}) - \overline{\text{rank}}(y_1)]^2 \sum_{i=1}^{N}[\text{rank}(y_{2i}) - \overline{\text{rank}}(y_2)]^2}} \tag{3.12}$$

The values of r_s can vary between –1 and 1. Values close to 1 (Fig. 3.3(a)) indicate strong direct correlations between y_1 and y_2, values close to –1 (Fig. 3.3(b)) indicate strong inverse correlations between y_1 and y_2, values close to 0 indicate the absence of correlations (Fig. 3.3(c)) or the presence of a non-monotonic correlation (Fig. 3.3(d)).

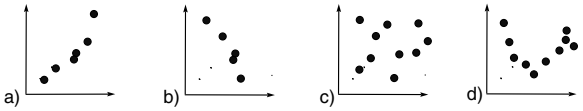


Fig. 3.3. Correlation or non-correlation between two objective functions

If we perform N simulations by means of a mathematical system model by using a uniformly distributed sample of design variable vectors $\mathbf{x}_1, \ldots, \mathbf{x}_N$ (see Sect. 3.4.2), a matrix $N \times n_{of}$ of objective function vectors can be assembled which reads

$$\begin{matrix} p=1 \\ p=2 \\ \cdots \\ p=N \end{matrix} \begin{pmatrix} f_1(\mathbf{x}_1) & f_2(\mathbf{x}_1) & \ldots & f_{n_{of}}(\mathbf{x}_1) \\ f_1(\mathbf{x}_2) & f_2(\mathbf{x}_2) & \ldots & f_{n_{of}}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(\mathbf{x}_N) & f_2(\mathbf{x}_N) & \ldots & f_{n_{of}}(\mathbf{x}_N) \end{pmatrix} \tag{3.13}$$

In order to analyse the existence of a correlation between two generic objective functions, we need to consider two different columns i j of the matrix (3.13) obtaining a bivariate set of objective functions corresponding to the same pth design variable vector $\mathbf{x}_p$.

This matrix can be analysed to find all of the existing correlations between the objective functions taken 2×2. Spearman's rank–order correlation coefficient can be employed. The following *matrix of correlation coefficients* can be constructed, in which the generic element $r_{s\,ij}$ refers to the correlation between the ith and the jth objective function. Obviously, $r_{s\,ij} = r_{s\,ji}$, i.e. the matrix is symmetrical

$$\begin{pmatrix} r_{s\,11} & r_{s\,12} & \cdots & r_{s\,1n_{of}} \\ r_{s\,21} & r_{s\,22} & \cdots & r_{s\,2n_{of}} \\ \vdots & \vdots & \ddots & \vdots \\ r_{s\,n_{of}1} & r_{s\,n_{of}2} & \cdots & r_{s\,n_{of}n_{of}} \end{pmatrix}$$

Three conditions can occurs

- direct correlation: $(0 < r_s \leq 1)$;
- inverse correlation: $(-1 \leq r_s < 0)$;
- no correlation or non-monotonic correlation: $(r_s = 0)$.

In presence of a strong direct correlation (r_s close to 1), it is possible to keep only one of the redundant objective functions reducing the computational expense in the subsequent optimisation process.

Redundancy check is important because it may significantly simplify the problem, if many redundant objectives can be eliminated.

3.3.4 Global Sensitivity Via Artificial Neural Network

By means of a procedure based on the use of an artificial neural network (ANN, see Chap. 4) it is possible to perform GSA [100, 202]. By analysing the interconnection weights of the trained ANN (w_{ij}) we can get an insight into the influence of each design variable on objective functions. A sensitivity analysis can be performed on the entire design variable space by computing the transition matrix $\mathbf{T}$ (size $n_{dv} \times n_{of}$) which gives the normalised contribution of the n_{dv} design variables on the values of the n_{of} performance indices.

$$\hat{t}_{ij} = \frac{t_{ij}}{\max_i |t_{ij}|} \quad i = 1, \ldots, n_{dv},\ j = 1, \ldots, n_{of} \tag{3.14}$$

where $\mathbf{T}_{n_{dv} \times n_{of}} = \prod_{l=1}^{n_l - 1} \mathbf{W}_{n_{l-1} \times n_l}$ and $\mathbf{W}_{n_{l-1} \times n_l}$ is the matrix of the interconnection weights between the layer l and the layer $l - 1$ (see Chap. 4) and n_l is the total number of layers in the network.

3.4 Pareto-optimal Set Computation

The computation of the whole or part of the Pareto-optimal set is the key step of MOO/MOP. As introduced in Chap. 1 to compute the Pareto-optimal (PO) set there are two class of methods, namely (1) 'broadly applicable methods' and (2) 'scalarisation methods' (solution of MOP problems via NLP[3]). The following methods and/or tools refer to the first class

- Exhaustive method,
- Low discrepancy sequences,
- Uniformly distributed sequences,
- Genetic algorithms.

[3]The scalarisation methods for generating Pareto-optimal sets convert the multi-objective programming problem into a single–criterium problem (scalarisation)

Some of the algorithms to solve the scalarisation methods are

- Unconstrained minimisation,
- Simplex method,
- Sequential unconstrained minimisation techniques,
- Method of feasible directions, sequential quadratic programming,
- Weighted sum,
- Constraints method.

All of these methods are often called *search* methods as they allow to search for the optimal solutions (Pareto-optimal set).

3.4.1 Exhaustive Method

By using the exhaustive method, the number of all possible combinations of design variables values is

$$n_{cdv} = n_v^{n_{dv}} \tag{3.15}$$

and the total time t_t for computing all of the objective functions as function of all possible combinations of design variables is

$$t_t = t_s n_{of} n_v^{n_{dv}} \tag{3.16}$$

The total time t_t for computing all of the objective functions is clearly a time too long to solve an engineering problem (see Chap. 1).

The critical factors in Eq. (3.16) are the number of design variables n_{dv}, the number of values the design variables may assume within their respective definition ranges n_v, and the simulation time for computing one single objective function t_s. Often, the number of objective functions n_{of} is less critical.

It is clear that complexity is strictly connected to the number of design variable of a system, more than to the number of objective functions (see Eq. (3.16)).

The exhaustive method is the simplest, but it is a more time-consuming search method due to the huge amount of computations required. Even for slightly complex optimisation problems the exhaustive method is generally unmanageable. The computation of the Pareto-optimal set can be performed according to the procedure described in Sect. 3.4.3.

3.4.2 Low Discrepancy Sequences

Low discrepancy sequences constitute a family of different series of points. A low discrepancy sequence can be used to define a set of design variable combinations to be exploited to sample objective functions.

The computation of the Pareto-optimal set by means of low discrepancy sequences is based on the following consideration. Instead of evaluating the objective functions and subsequently updating the design variables towards

the optimal values of the objective functions (e.g. by means of a gradient method, see [8]), several objective functions are evaluated a priori (and then they are at the designer's disposal to be analysed, the computation of the Pareto-optimal set can be performed according to the procedure described in Sect. 3.4.3). In principle, the design variable values can be chosen according to a random sequence. Theoretically, it is not possible to 'generate' in an n_{dv}-dimensional space a *random* sequence of points using a computer that is a completely deterministic machine. In fact a deterministic procedure can generate only a pseudo–random sequence. Low discrepancy sequences differ form pseudo-random sequences due to the fact that the points are more evenly distributed in the space. Considering the points generated in an n_{dv}-dimensional spaces using a pseudo–random generator we notice regions with no points at all and regions with concentrations (Fig. 3.4). That is clearly undesirable.

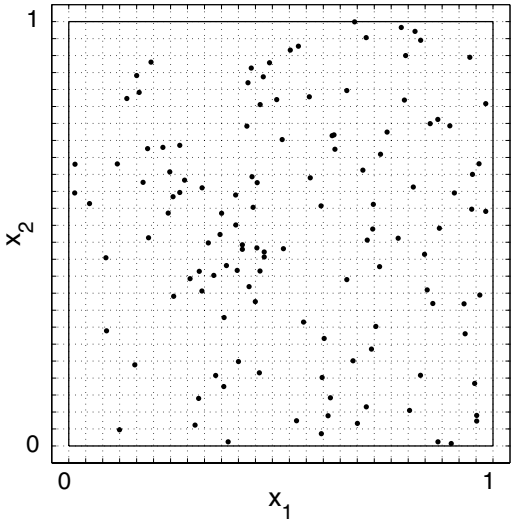


Fig. 3.4. Pseudo–random points in a bidimensional space

According to the definition of uniformity given by Weyl [154] if we consider a sequence of points[4] $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_N$ belonging to a unit hypercube $\mathcal{O}^{n_{dv}}$, the sequence $\mathbf{x}_i$ is called uniformly distributed in $\mathcal{O}^{n_{dv}}$ if

$$\lim_{N\to\infty} \frac{\#_N(\mathcal{G})}{N} = V(\mathcal{G}) \tag{3.17}$$

where $\mathcal{G}$ is an arbitrary domain in $\mathcal{O}^{n_{dv}}$, $\#_N(\mathcal{G})$ is the number of points $\mathbf{x}_i$ belonging to $\mathcal{G}$ and $V(\mathcal{G})$ is the volume of $\mathcal{G}$.

[4] A point $\mathbf{x}_i = \{x_{1i}, x_{2i}, \ldots, x_{n_{dv}i}\}^T$ represents a design variable vector

If a sequence is uniformly distributed, then

$$\lim_{N\to\infty} 1/N \sum_{i=1}^{N} f(\mathbf{x}_i) = \int_{[0,1]^{n_{dv}}} f(\mathbf{x}) d\mathbf{x} \tag{3.18}$$

for any real–valued continuous function f defined in $\mathcal{O}^{n_{dv}} = [0,1]^{n_{dv}}$.

Until recently, the only known design for uniformity on an n_{dv}-dimensional hypercube was a uniform grid. However, the uniform grid requires a huge number of points (see Sect. 3.4.1).

Several sampling techniques have been employed in order to define low discrepancy sequences having almost the same discrepancy of a uniform grid but employing a very limited number of points.

Low discrepancy sequences (and low discrepancy nets [183, 184]) have the best uniformity characteristics presently known [183, 246].

These sequences may be constructed with a number of points N acceptable for practical applications (i.e. allowing a fast computation of a Pareto-optimal set), even for spaces of large dimension ($n_{dv} > 30$).

Figure (3.5) shows a comparison between points generated by using a low discrepancy sequence and a sequence generated using a pseudo–random generator [152] in a bidimensional domain. The low discrepancy sequence shows evidently better uniformity characteristics. Moreover, the uniformity of the low discrepancy sequence is independent of the number of points, that is even if N is small, the points are distributed evenly in the space (this concept will be dealt with later) [154].

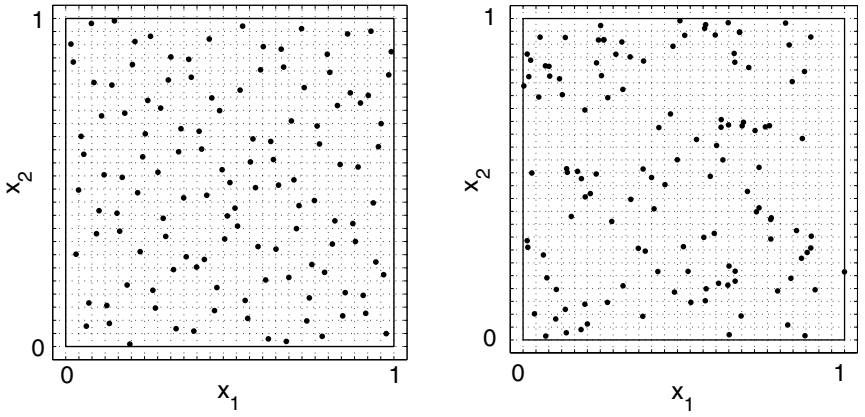


Fig. 3.5. Comparison of low discrepancy (Sobol) (a) and pseudo–random (Matlab 5.3©) generated points (b)

Discrepancy is a quantitative measure for the deviation of a sequence from a uniform distribution, that is the discrepancy measures how evenly a given set of points is distributed in a given hypercube (unit cube $\mathcal{O}^{n_{dv}}$ in dimension n_{dv}). The two most common definitions of discrepancy follow.

Definition 3.1 (Discrepancy D_N and D_N^*). *Given a unit hypercube $\mathcal{O}^{n_{dv}}$ of dimension n_{dv}, for every sub-interval $\mathcal{A}$ of $\mathcal{O}^{n_{dv}}$ of the form $[u_1, v_1) \times \ldots \times [u_{n_{dv}}, v_{n_{dv}})$ we divide the number of points $\mathbf{x}_k$ in $\mathcal{A}$ by N and we take the absolute difference of this quotient to the volume of $\mathcal{A}$; the maximum difference is the discrepancy D_N.*

In mathematical form

$$D_N(\mathbf{x}_1, \ldots, \mathbf{x}_N) := \sup_{\substack{0 \leq u_i < v_i \leq 1 \\ i = 1, \ldots, n_{dv}}} \left| \frac{\#\quad \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \cap \prod_{i=1}^{n_{dv}} [u_i, v_i)}{N} - \prod_{i=1}^{n_{dv}} (v_i - u_i) \right| \tag{3.19}$$

An alternative definition is given by

$$D_N^*(\mathbf{x}_1, \ldots, \mathbf{x}_N) := \sup_{\substack{0 \leq v_i \leq 1 \\ i = 1, \ldots, n_{dv}}} \left| \frac{\#\quad \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \cap \prod_{i=1}^{n_{dv}} [0, v_i)}{N} - \prod_{i=1}^{n_{dv}} (v_i) \right| \tag{3.20}$$

where $\prod_{i=1}^{n_{dv}} [u_i, v_i)$ and $\prod_{i=1}^{n_{dv}} [0, v_i)$ indicate hypercubes cartesian product whose dimensions are defined by u_i and v_i.

Since for every point set $\mathcal{Q} = \mathbf{x}_1, \ldots, \mathbf{x}_N$ we have (as shown in [246])

$$D_N^*(\mathcal{Q}) \leq D_N(\mathcal{Q}) \leq 2^{n_{dv}} D_N^*(\mathcal{Q}) \tag{3.21}$$

the two measures D_N and D_N^* are equivalent in this sense.

We would like to have point sets $\mathcal{Q} = \mathbf{x}_1, \ldots, \mathbf{x}_N$ with discrepancy $D_N(\mathcal{Q})$ as small as possible. For practical reasons we need a small discrepancy for every N which is large enough.

If the dimension n_{dv} equals 1

- for every N an optimal point set $\mathcal{Q}$ exists such that $x_i = \frac{2i-1}{2N} : i = 1, \ldots, N$ with discrepancy $D_N(\mathcal{Q}) = \frac{1}{N}$ and $D_N^* = \frac{1}{2N}$.
- for any sequence with $N \to \infty$ in $\mathcal{O}^1$ we have $D_N \geq c \frac{\ln N}{N}$ [246] with the absolute constant $c > 0$. In fact there exist several sequences with $D_N = O(\frac{\log N}{N})$ [246].

If the dimension n_{dv} is ≥ 2

- there exist constructions (e.g. Hammersley point set [246]) which give for any N a point set $\mathcal{Q}$ with $D_N(\mathcal{Q}) = O((n_{dv} - 1)\frac{\ln N}{N})$.

- several point sequences (e.g. Halton, Sobol, Faure sequences [246]) have $D_N(\mathcal{Q}) = O(n_{dv}\frac{\ln N}{N})$.
- except for the case $n_{dv} = 2$ we do not know if it exists a constant $B_{n_{dv}} > 0$ (depending only on n_{dv}) such that for every point set $\mathcal{Q}$ in $\mathcal{O}^{n_{dv}}$ the inequality $D_N(\mathcal{Q}) \geq B_{n_{dv}}(n_{dv} - 1)\frac{\ln N}{N}$ holds; if such $B_{n_{dv}}$ exits, then there also exists a constant $\overline{B_{n_{dv}}} > 0$ such that for any point sequence in $\mathcal{O}^{n_{dv}}$ we have $D_N(\mathcal{Q}) \geq \overline{B_{n_{dv}}}\, n_{dv}\frac{\ln N}{N}$ for $N \to \infty$ [246].

Definition 3.2 (Low–discrepancy sequence). *For a given dimension n_{dv} a point sequence $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ in $\mathcal{O}^{n_{dv}}$ is called a low discrepancy sequence when $D_N(\mathbf{x}_1, \ldots \mathbf{x}_N) = O(n_{dv}\frac{\ln N}{N})$.*

For every low discrepancy sequence $\mathbf{x}_1, \mathbf{x}_2, \ldots$ of dimension n_{dv} there exists a constant $c_{n_{dv}}$ such that [246]

$$D_N^*(\mathbf{x}_1, \ldots, \mathbf{x}_N) \leq c_{n_{dv}} n_{dv}\frac{\ln N}{N} + O\left((n_{dv} - 1)\frac{\ln N}{N}\right) \tag{3.22}$$

When considering a low discrepancy sequence one can see that the space $\mathcal{O}^{n_{dv}}$ is evenly filled with points, and the process of filling is the same in every region of the hypercube. By working with low discrepancy sequences one can see that too few points may not fill the hypercube; it is obvious that a low discrepancy sequence may not be useful if a minimum number of points is not generated.

Low discrepancy sequences are useful to explore the design variables space. However, being able to compute a function f at any given design variable vector does not imply that one *'understands'* the behaviour of the function [45]. We may not know if the function is continuous, bounded or where its minimum is.

Evaluating f at a well-chosen set of locations $\mathbf{x}_1, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_N$ by computing the responses $\mathbf{y}_1, \ldots, \mathbf{y}_i, \ldots, \mathbf{y}_N$, the function f can be visualised. By plotting the responses versus the input variables, we may identify strong dependencies between $\mathbf{x}$ and $\mathbf{y}$.

The same procedure can be adopted to find correlations between two generic objective functions and between design variables and objective functions. We can use low discrepancy sequences to find points with desirable values of $\mathbf{y}$ that can be used to identify the most promising sub–region of the design variables space. If we have a system model whose performances can be simulated quickly (see Chap. 4) we can perform millions of simulation and find the Pareto-optimal set directly by selecting the design variable vectors by means of low discrepancy sequences.

Generating Low-Discrepancy Sequences

In order to explore the responses of a function $f(\mathbf{x})$ into the domain $\mathbf{x} \in [0, 1]^{n_{dv}}$, it is natural to sample points on a regular grid. One chooses n_v

different values for each $\{x_1, \ldots, x_j, \ldots, x_{n_{dv}}\}$ on $[0,1]^{n_{dv}}$ and then analyses all the $n_v^{n_{dv}}$ combinations. This works well for small values of n_{dv} (2 or 3), but for large n_{dv} it becomes (as already mentioned before) completely impractical. Moreover, in situations where one of the responses y_i depends strongly on only one or two of the inputs x_j the grid design leads to waste samples, i.e. simulations that could have been time-consuming.

A more efficient approach may be given by *Latin hypercubes* [155]. The central idea of Latin hypercube sampling is to subdivide the domain [0,1] along each dimension of $\mathbf{x}$ into n_v sub–intervals ensuring that one sample lies in each sub–interval. This could be done by choosing some n_{dv}-independent uniform random permutations $\pi_1 \ldots \pi_{n_{dv}}$ of the integer numbers from 1 to N, and letting the sampling locations $(x_{ij};\ i = 1, \ldots, N;\ j = 1, \ldots, n_{dv}\ )$ to be

$$x_{ji} = \frac{\pi_{ji} - 0.5}{N} \tag{3.23}$$

where π_{ji} denotes the value of the permutation j in the ith position. In Fig. 3.6 a projection of $N = 25$ points from a centred Latin hypercube in $n_{dv} = 5$ design variables onto two dimensions is reported. Every design variable is sampled in each of 25 equally spaced one-dimensional sub–intervals of size 1/25. Latin hypercube samples look like random scatter in any bivariate plot. Some effort has been made to find good Latin hypercube samples [192].

One approach is to find Latin hypercube samples that have small correlations between design variables. This can be done by using the so-called *orthogonal arrays* (OAs). An OA [21] is an $N \times n_{dv}$ matrix of integers $1 \leq oa_{ji} \leq b$,

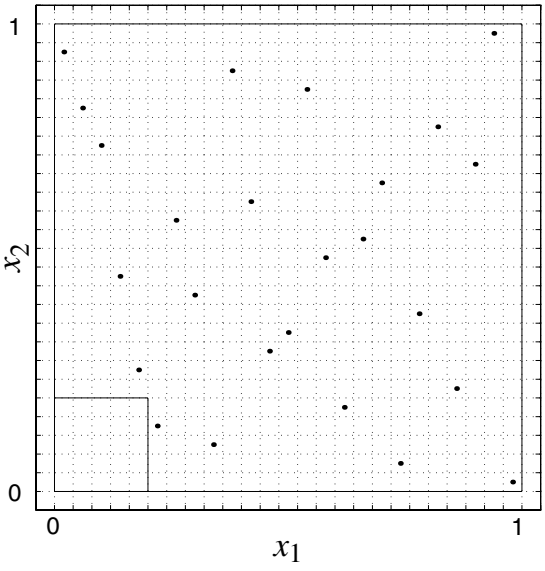


Fig. 3.6. Twenty-five points of a Latin hypercube sample

with b integer. The array has strength $t \leq n_{dv}$, if in every $N \times t$ sub–matrix of **OA**, each one of the b^t possible columns appear the same number of times ν. It must be $N = \nu b^t$.

The following matrix gives an example of an orthogonal array whose parameters are $(N, n_{dv}, b, t, \nu) = (25, 5, 5, 2, 1)$:

$$\mathbf{OA} = \begin{pmatrix} 1&1&1&1&1&2&2&2&2&2&3&3&3&3&3&4&4&4&4&4&5&5&5&5&5 \\ 1&2&3&4&5&1&2&3&4&5&1&2&3&4&5&1&2&3&4&5&1&2&3&4&5 \\ 1&2&3&4&5&2&3&4&5&1&3&4&5&1&2&4&5&1&2&3&5&1&2&3&4 \\ 1&5&4&3&2&2&1&5&4&3&3&2&1&5&4&4&3&2&1&5&5&4&3&2&1 \\ 2&1&5&4&3&4&3&2&1&5&1&5&4&3&2&3&2&1&5&4&5&4&3&2&1 \end{pmatrix}^T$$

The sampling location for an **OA** is,

$$x_{ji} = \frac{\pi_{ji}(oa_{ij}) - 0.5}{b} \tag{3.24}$$

where $\pi_j(oa_{ij})$ are some independent uniform permutations of the columns of **OA**. The central idea is to divide the domain $[0,1]^{n_{dv}}$ and consider the family of $b^{n_{dv}}$ sub–cubes obtained by splitting each axis into b intervals of equal size. Each row of **OA** can be used to identify one single sub–cube, e.g. the first sub–cube of the above **OA** is identified by the indices (1,1,1,1,2). To see the advantage of this technique, let us consider the sample distribution with respect to any of the t coordinate axes (i.e. project the samples into the subspace spanned by these axes). This subspace can be divided into b^t sub–cubes by splitting each axis into b intervals.

The main property of orthogonal array sampling is that each of these sub–cubes contains the same number of samples. Considering the above matrix **OA**, in each $N \times t$ (25×2) sub–matrix, by the definition of orthogonal arrays, each of the possible $b^t = 25$ rows occurs $\nu = 1$ times, so that there will be $\nu = 1$ sample in each sub–cube. The projection of 25 points of an orthogonal array sample on two dimensions with $b = 5$ and $t = 2$ would be identical to a 5×5 grid. In this case in every one-dimensional projection there will be exactly $\nu b^t / b = \nu b^{t-1} = 5$ samples in each interval of width $1/b = 1/5$. This is lower with respect to Latin hypercube sample that for the same number of samples $N = \nu b^t = 25$ has one point in each interval of width $1/(\nu b^t) = 1/25$.

There is a simple modification to orthogonal arrays that yields the same one-dimensional stratification properties given by Latin hypercube sampling, known as *orthogonal-array–based Latin hypercube sampling* [244]. The idea is to remap the νb^t symbols within each of the column into a single sequence $1, 2, \ldots, \nu b^t$, by mapping the νb^{t-1} identical copies of each symbol m with $1 \leq m \leq b$ into a random permutation of the symbols $\nu b^{t-1} m{+}1, \ldots, \nu b^{t-1}(m{+}1)$. This process is repeated for each column separately. Figure 3.7 shows a projection of 25 points from an orthogonal-array-based Latin hypercube sample over five variables onto two of the coordinate points. Each design

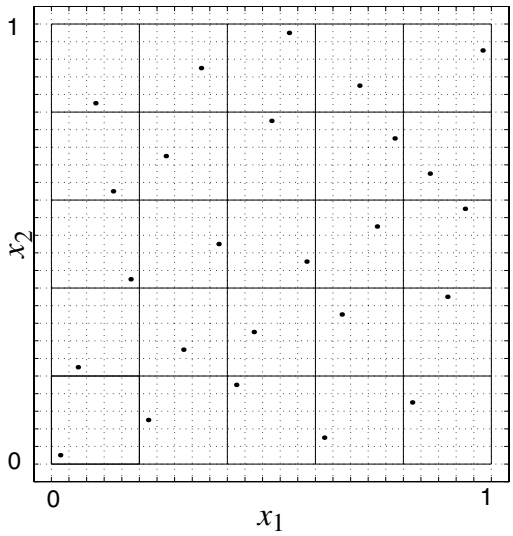


Fig. 3.7. Twenty-five points of an orthogonal-array-based Latin hypercube sample

variable is explored in each of the 25 equal monodimensional sub–interval of size $1/25$ and each pair of variable is explored in each of the 25 bidimensional squares of size $1/5 \times 1/5 = 1/25$.

Orthogonal-arrays-based Latin hypercubes samples were developed mainly to deal with design variable which could assume discrete values [192, 244]. Recently, researchers have developed techniques of sampling design variables in a continuous space [183, 246].

Here we briefly describe particular *low discrepancy sequences* known as (t, m, n_{dv})*-nets* and (t, n_{dv})*-sequences*. Let $n_{dv} \geq 1$ and $b \geq 2$ be integers. An elementary sub–cube in base b is defined by

$$\Xi = \prod_{j=1}^{n_{dv}} \left[\frac{c_j}{b^{k_j}}, \frac{c_j + 1}{b^{k_j}} \right] \tag{3.25}$$

for $k_j \geq 0$ and $0 \leq c_j < b^{k_j}$ integers, $\prod$ stands for cartesian product. Let $m \geq 0$ be an integer. A set of points $\mathbf{x}_i$, $i = 1, \ldots, b^m$ in $[0, 1)^{n_{dv}}$ is a $(0, m, n_{dv})$–net in base b if every elementary sub–cube Ξ in base b of volume $1/b^m$ includes one of the points. Let $t \leq m$ be a non-negative integer. A finite set of b^m points from $[0, 1)^{n_{dv}}$ is a (t, m, n_{dv})–net if every elementary sub–cube in base b of volume $1/b^{m-t}$ contains exactly b^t points of the sequence.

For $t \geq 0$, an infinite sequence of points $\mathbf{x}_i$ from $[0, 1)^{n_{dv}}$ is a (t, n_{dv})–sequence in base b if for all $k \geq 0$ and $m \geq t$ the finite sequence of points $\mathbf{x}_i$ with $i = kb^m + 1, \ldots, (k + 1)b^m$ is a (t, m, n_{dv})–net in base b. If we find that the first b^m points of the sequence are not sufficient for the solution of the problem under consideration (integration of a function, function minimisation

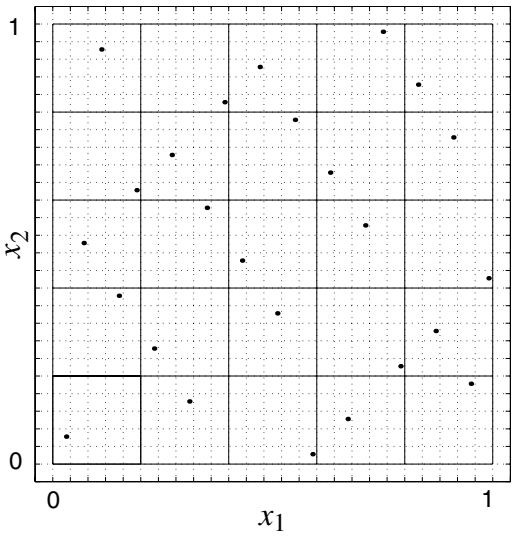


Fig. 3.8. Twenty-five points of a low discrepancy (0,2,5)–net in base 5. For each of the two variables that are plotted, there is one point in the reference square. Each variable is sampled once for all of the 25 one-dimensional sub-intervals

as will be showed in the following), we can find other b^m points that form a (t, m, n_{dv})–net that also fill in the places not occupied by the first set. If we continue to add point having b (t, m, n_{dv})-nets, the complete set of points form a $(t, m+1, n_{dv})$–net. The construction algorithm of (t, m, n_{dv})–net and (t, n_{dv})-sequences is given in [183]. The construction algorithm given by [232] gives (t, n_{dv})-sequences in base 2. For prime numbers of n_{dv} the construction given by [61] gives $(0, n_{dv})$-nets in base n_{dv} extended by [183] to prime powers[5] of n_{dv}. Using these constructions we can choose b to be the smallest prime power number greater or equal to n_{dv} and use the first n_{dv} variables of the corresponding $(0, b)$–sequence in base b in order to obtain a (t, n_{dv})–sequence with t minimum. Figure 3.8 shows the 25 points of a $(0, 2, 5)$–net in base 5 projected onto two of the five input coordinates. These are the 25 points of a $(0, 5)$–sequence in base 5. This design has the equidistribution properties similar to an orthogonal-array-based Latin hypercube sample as in Fig. 3.7. Moreover, every consecutive 25 points in the sequence $\mathbf{x}_{25k+1}, \ldots, \mathbf{x}_{25(k+1)}$ have these equidistribution properties. The first 125 points, shown in Fig. 3.9, have better equidistribution properties: in any pair of variables the points appear as a 5×5 grid of 5-point Latin hypercube samples and each individual input can be split into 125 cells having one point. The first 625 points are shown in Fig. 3.10.

Now, we discuss how we can use the good distribution property of low discrepancy sequences. (t, n_{dv})-sequences can be used [246] to estimate the

[5]For example, $n_{dv} = 3, 5, 7, 9, 11, 13, 17, 19, 23, 25, 27, \ldots$

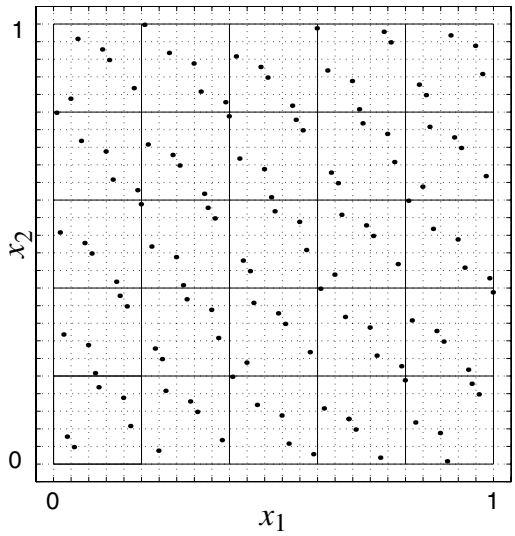


Fig. 3.9. One hundred and twenty-five points of a low discrepancy (0,3,5)–net in base 5. For each of the two variables that are plotted, the result is a 5×5 grid of 5-point Latin hypercube samples. The points in a three-dimensional projection can be split into 125 cubes having one point

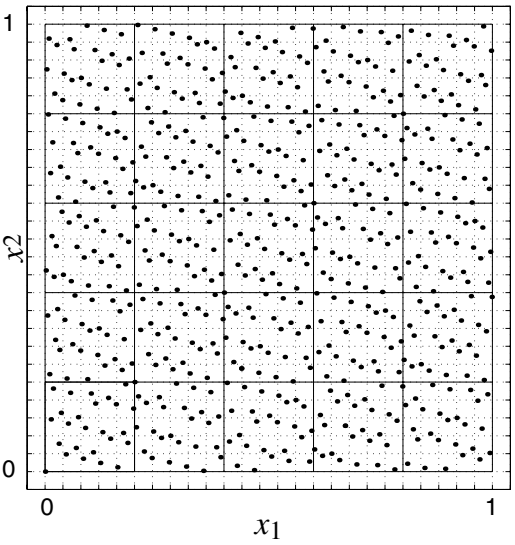


Fig. 3.10. Six hundred and twenty-five points of a low discrepancy (0,4,5)–net in base 5. For each of the two variables that are plotted, the square can be divided into 625 square of side 1/25 or into 625 rectangles of side $1/5 \times 1/125$ and each square or rectangle has one of the points. Each variable is sampled once for all of the 625 one-dimensional sub–intervals

minimum of a function $f(\mathbf{x})$,

$$\mu(f) = \inf_{\mathbf{x}\in[0,1]^{n_{dv}}} f(\mathbf{x}) \tag{3.26}$$

With (t, n_{dv})-sequences we can use [246]

$$\mu_N(f) = \min_{1\leq i\leq N} f(\mathbf{x}_i) \tag{3.27}$$

to achieve a good estimate. The idea is to distribute the samples as uniformly as possible, by choosing their locations deterministically. Let $\mathcal{Y}(N) = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ be a set of points in $[0,1]^{n_{dv}}$. Typically, the goal of this method is to minimise the *irregularity of distribution* of the sample with respect to some quantitative measure. One such measure is the *discrepancy* D_N^* of $\mathcal{Y}(N)$.

As will be shown, discrepancy is closely correlated to *dispersion* of points defined by

$$d_N(\mathcal{Y}(N)) = \max_{\mathbf{v}\in[0,1]^{n_{dv}}} \min_{1\leq i\leq N} \|\mathbf{v} - \mathbf{x}_i\| \tag{3.28}$$

where $\|\cdot\|$ denotes a distance between two points on the design variables space. The *modulus of continuity* is defined as

$$\omega(f, \tau) = \max_{\mathbf{v},\mathbf{z}\in[0,1]^{n_{dv}};\|\mathbf{v}-\mathbf{z}\|\leq\tau} |f(\mathbf{v}) - f(\mathbf{z})| \tag{3.29}$$

for all $\tau \geq 0$. Note that for a given function f, the modulus of continuity $\omega(f, \tau)$ is non-decreasing as τ grows. We thus have the following relation [183] between dispersion and the search error (see (3.26) and (3.27))

$$|\mu(f) - \mu_N(f; \mathcal{Y}(N))| \leq \omega(f, d_N) \tag{3.30}$$

This result ensures that a point set with small dispersion d_N gives better optimal solutions. Hereafter, if we assume as distance $\|V\|$ the maximum component of the vector $\mathbf{v}$, there are lower and upper bounds for the dispersion [246], given by

$$\frac{1}{2[N^{1/n_{dv}}]} \leq d_N \leq (D_N^*)^{1/n_{dv}} \tag{3.31}$$

Thus, having low discrepancy D_N^* means having low dispersion d_N, low modulus of continuity $\omega(f, d_N)$ and better optimal solutions.

3.4.3 Selection of the Pareto-optimal Set

The search procedure introduced in Sect. 3.4 leads to an extremely robust method for generating the Pareto-optimal set. We generate a finite number of feasible solutions (as many as 10^6 as shown in Chap. 1) and then we select the Pareto-optimal solutions according to the definition.

An algorithm which selects the Pareto-optimal solutions from a given set of feasible solutions can be written on the basis of the Contact Theorem [223].

Theorem 3.3. *Given a negative cone in* $\Re^{n_{of}}$

$$\mathcal{C}^- = \{\mathbf{f} \in \Re^{n_{of}} \ : \ \mathbf{f} \leq \mathbf{0}\} \tag{3.32}$$

($\mathcal{C}^-$ *is shown in Fig. 3.11) a vector* $\mathbf{f}^*$ *is a Pareto-optimal solution for the multi-objective optimisation problem if and only if*

$$(\mathcal{C}^- + \mathbf{f}^*) \cap \mathcal{C} = \{\mathbf{f}^*\} \tag{3.33}$$

The proof of the theorem is given in [166]. A graphical illustration of the theorem for a two-criterion problem is shown in Fig. 3.11. The method may be the only feasible approach to find PO sets for many engineering problems (non convex PO set).

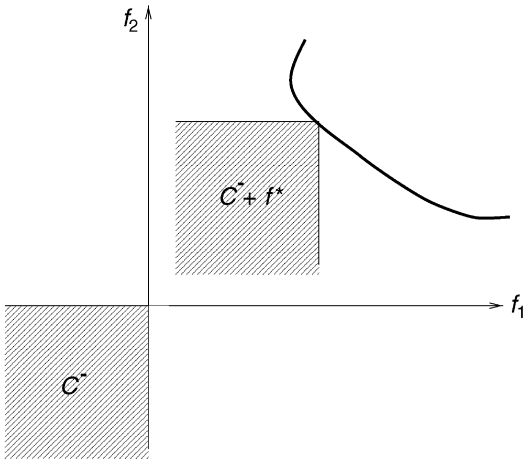


Fig. 3.11. Contact theorem: two-criterion problem

The major shortcoming of this approach is that there is no guarantee that the Pareto-optimal solutions for successive finite sets converge to those of the continuous problem. This fact is relevant from the mathematical point of view, but it is not important for the decision maker who is looking for any solution better than the actual one.

3.4.4 Genetic Algorithms

Genetic algorithms (GAs) are suitable to find the minimum of a function (or of a set of functions) by performing a semi–stochastic search [39, 40, 96, 97, 105, 117, 222].

An analogy may be drawn with the natural evolution process that took place on the Earth in the last 500 million years. GAs, similarly to their natural counterpart, use a design variable representation which refers to chromosomes.

GAs are based on an elitist reproduction strategy, where strongest members of the population (design solutions) are selected for reproduction and are given the opportunity to strengthen the chromosomal (i.e. genes, design variables) makeup of the next generation. Unlike many other search techniques, GAs consider multiple design solutions (a 'population') at each iteration. The design solutions in a population are called 'individuals'. The design variable vector referring to a particular design solution is typically encoded into binary or real strings ('chromosomes'). During each iteration, the individuals undergo three operations: selection, crossover and mutation.

The key issues of GAs are

- GAs work on function evaluation alone and do not require function derivatives;
- GAs proceed from several points in the design variable domain (population), consequently, the method has a better probability of locating global minimum;
- GAs allow design variable spaces consisting of a mix of continuous and discrete variables;
- GAs use probabilistic transition rules, not deterministic rules;
- GAs can be easily implemented on parallel computers.

In order to examine the multi-objective genetic algorithm, it is helpful to examine the scalar version first.

Structure of Genetic Algorithms

A genetic algorithm consists of the following steps:

- initial population generation,
- selection,
- crossover and mutation.

The initial population can be generated by a random sampling over the whole design variable domain. It is important that the initial population consists of a set of design solutions that is reasonably representative of the whole design solution domain. The methods presented in Sect. 3.4.2 can improve the sampling uniformity.

Selection is the operator which determines which individuals will be represented, and to what extent, in the next population. It is based on a 'fitness function' defined such that good design solutions have high fitness values compared to bad design solutions. If we consider a constrained problem, the fitness function usually incorporates one or more penalty terms corresponding to the constraints.

Central to this approach is the step by which a design variable vector is coded into a stringlike structure, and the most common approach is to represent each design variable by a binary number. The length of the string

determines the precision with which each design variable is represented. We code the design variable x_i as a binary integer of known length.

$$a_1\, a_2\, \ldots .\, a_{n_b}$$

The precision of this mapped coding may be calculated

$$\Pi = \frac{U_{max} - U_{min}}{2^{n_b} - 1}$$

where n_b is the length of the coding string, i.e. number of bits used. The single U_i parameter maps linearly between

$$\begin{aligned} 0000 \ldots 0 &= U_{min,i} \\ 1111 \ldots 1 &= U_{max,i} \end{aligned}$$

If an optimisation problem is defined by many design variables, a multi-parameter coding is needed. To construct a multi-parameter coding, we can simply concatenate (i.e. adjoining) as many single–parameter coding as we require.

$$a_{11}\, a_{21}\, \ldots .\, a_{n_b 1} a_{12}\, a_{22}\, \ldots .\, a_{n_b 2} \ldots a_{1 n_{dv}}\, a_{2 n_{dv}}\, \ldots .\, a_{n_b n_{dv}}$$

Binary design variables correspond to the smallest possible alphabet, which has some theoretical advantages. For example, with a very large alphabet it is more difficult to ensure that every possible gene is present in the initial population. However, studies (e.g. [119,162]) comparing binary to other representations tend to find superior performance by non-binary representation. The use of real variables in strings (with obvious modifications of the crossover and mutation operators) has been described by many authors (e.g. [119, 162]). Several methods of selecting candidate solutions that survive from one generation to the next have been reported in the literature.

A common selection strategy is the 'roulette wheel' selection, in which the probability of selecting a specific individual (design solution/combination of design variables) c_i is proportional to its fitness value f_i in a population (population size is N)

$$c_i = \frac{f_i}{1/N \sum_{j=1}^{N} f_j}$$

The roulette wheel generates a random number to determine which individual (design solution) will be selected; this is repeated as many times as there are design variable combinations in each population. This generally results in the most fitting individuals to be selected and the least fitting being discharged.

Crossover process allows exchange of genes (i.e. design characteristics) between individuals of the population. Crossover takes place after selection and occurs with a defined probability, the crossover rate (P_c). The simplest

binary form of this operation is the 'single-point crossover', a pair of chromosomes is selected, all the digits which occur after the cut point in one chromosome are replaced by the digits in the other one. This process generates new design solutions, which possess some characteristic of previous high fitness design solutions. The last operation to take place is mutation. It plays a secondary role in the operation of GAs. This involves simply altering the value of a few randomly selected bits in the population. This occurs with a fixed probability called mutation rate (P_m). Mutation is intended to enable GAs to explore new regions of the design feasible domain and escape from local optima. The processes of selection, crossover, and mutation are repeated until the termination criteria are met.

Characteristics of Genetic Algorithms

The implicit parallelism available in GAs is significant from a computational standpoint, and has been considered by Holland [117] and Goldberg [96]. The exploitative power of GAs exists in the fact that the presence of a 0 or 1 at some key locations along the string may be significant in itself. Such patterns of 0s and 1s occupying key locations are referred as 'schema'.

$$10 * *01 * * * * * * \ldots * 1$$

'*' indicates that a location along the string can be occupied either by a 1 or a 0.

The order of the schema, denoted as $O(\mathcal{H})$, is defined as the number of explicitly stated zeros and ones in that schema. The length $d(\mathcal{H})$ is taken as the distance between the first and last specific digit on the string. The number of useful schema processed by the genetic search in the initial stage is of $O({n_b}^3)$. Holland described this computational leveraging available in GAs as an implicit parallelism [117].

Denoting the number of schemata of type $\mathcal{H}$ at a given generation t as $m(\mathcal{H}, t)$, at generation $t + 1$ this number [105] would be

$$m(\mathcal{H}, t+1) = m(\mathcal{H}, t)\frac{f(\mathcal{H})}{f_{avg}}(1 - P_c\frac{d(\mathcal{H})}{n_b - 1} - P_m\, O(\mathcal{H}))$$

where

- $f(\mathcal{H})$ is fitness of the schemata $\mathcal{H}$,
- f_{avg} is average fitness of the population,
- P_c is probability of crossover,
- n_b is length of the coding string,
- P_m is probability of mutation,

It is clear from this expression that, low-order, shorter defining length schemata, which are also fit, increase rapidly during the search process.

The handling of constraint functions in a genetic-search-based approach is somewhat contrived. Penalty functions methods have been the most popular approach [43]. In the traditional GA implementation, the fitness is represented by a scalar measure that can be formulated as a proper mathematical combination of objective and constraint functions. The scalar fitness function can be formulated as (penalty function approach):

$$\min \overline{f} = \min(f + \Phi)$$

$\overline{f}$ is the modified fitness function that also contains the penalty term Φ. The penalty Φ is obtained as

$$\Phi = \begin{cases} \Gamma & \text{if } \Gamma \leq L \\ L + \alpha(\Gamma - L) & \text{if } \Gamma > L \end{cases}$$

L is the limiter value of the penalty

$$0.0 < \alpha < 1.0 \qquad \Gamma = r \left(\sum_{j=1}^{n_c} < g_j > \right)$$

where r is a penalty parameter and

$$< g_j >= \begin{cases} g_j \text{ if } g_j > 0 \\ 0 \text{ otherwise} \end{cases}$$

The penalty function approach involves a number of penalty parameters which must be set to obtain feasible solutions. This dependency[6] of the GA performance on penalty parameters has led researchers to construct sophisticated penalty function approaches.

An alternative to the penalty function approach is based on strategies that adapt useful features of the feasible designs into the infeasible population (expression–based strategies) [107]: the infeasible design could be replaced by an expressed chromosome that incorporates features of the feasible design. Consider two binary strings representing a feasible and infeasible design

$$(A) \quad \text{feasible} \quad 11100101$$

$$(B) \quad \text{infeasible} \quad 10000101$$

The combination is generally obtainable combining the best feasible design in the population (A) with each infeasible design (B). This can be obtained by simply substituting the 0 or 1 at a specific location in the string (B) with the corresponding value from string (A) with some prescribed probability P_e.

Expression–based strategies perform generally better than the penalty function approach, producing better results with less computational effort. In presence of equality constraints the strict equality is replaced by using a band around the equality constraints and the width of the band is gradually reduced.

[6] The inclusion of the penalty term can strongly 'distort' the objective function creating artificially locally optimal solutions

Multi-objective Programming and Genetic Algorithms

The first practical implementation of a genetic algorithm (GA) for multi-objective optimisation is due to Schäffer [222]. He created equally sized subpopulations one for each of the objective functions. In this scheme the selection was performed independently for each objective function; however, reproduction and crossover were performed across subpopulations. The real problem of this implementation is the tendency to concentrate solutions only in some limited areas of the Pareto set.

Based on the concept of Pareto-optimal set, Goldberg [96] suggested a Pareto GA that used non-dominated ranking procedure. Ranking methods are used to grade the population in terms of Pareto-optimality. One of the aims of multiple criteria optimisation is to generate and maintain a widespread range of Pareto-optimal solutions. In order to promote this diversity Goldberg also proposed a 'niche' technique (fitness sharing) to prevent a population from drifting to some sub–regions of the Pareto-optimal set. The method encourages individuals in the population to move away from crowded areas.

The method uses GA properties and non-dominated solution selection algorithm to give a set of Pareto-optimal solutions [28, 119, 190, 236, 251].

There is no particular mathematical condition about the objective functions. They can be either continuous or not, differentiable or not. GA are able to obtain a population of solutions and to conduct a parallel search from many non-dominated solutions. The basic concept in multi-objective optimisation using GA is that, the fitness function is selected according to the non-dominated property (i.e. Pareto-optimality). Non-dominated individuals always have a higher probability to be selected because they have the highest fitness values.

The Pareto GAs have two operators namely 'niche' and 'Pareto-set' filter besides the three basic operators, reproduction, crossover and mutation.

Non-dominated procedure works in the criterion space. The procedure (minimisation) can be easily implemented. All individuals are arranged in ascending order with respect to the first criterion f_1. If the value of f_1 is the same, we consider the ascending order respect the second criterion f_2, and so on. Dominated individuals are eliminated, the remaining ones are non-dominated points belonging to the rank 1. From the remaining population, non-dominated individuals are assigned rank 2. The ranking continues until the entire population is considered. Figure 3.12 illustrates ranking for a bidimensional problem.

The non-dominated individuals are assigned an artificial fitness associated with their level of Pareto-optimality. The fitness F_i each individual has in a rank is determined as

$$F_i = \frac{(N_r - i + 1)}{\sum_{i=1}^{N_r}(N_r - i + 1)N_{si}/N}$$

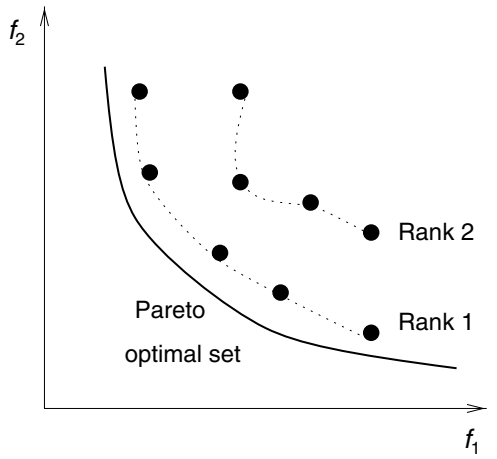


Fig. 3.12. Population rank into the criterion space f_1, f_2

where N_r is the highest rank of the population, N_{si} is the population size of rank i.

Sharing function approach [67, 97, 203, 236] is employed to maintain an even distribution of points into the objective functions space. A low fitness value is given to points with minimum distance relative to other points.

The distance between two individuals in the normalised objective functions space $\{f_1, f_2, \ldots, f_{n_{of}}\}^T$ is defined as

$$d_{ij} = \sum_{p=1}^{n_{of}} |f_p(\mathbf{x}_i) - f_p(\mathbf{x}_j)|$$

The sharing function is

$$Sh(d_{ij}) = \begin{cases} 1 - (\frac{d_{ij}}{\sigma_{share}})^{\alpha}, & \text{if } d_{ij} < \sigma_{share} \\ 0, & \text{otherwise} \end{cases} \tag{3.34}$$

where σ_{share} is the niche size and $\alpha = 2$–3. The niche size σ_{share} is relative to the portion of the Pareto-optimal set that the individual should stand for [67]. The parameter σ_{share} can be set to 0.01–0.02 for computation. The niche count m_i is computed across the N individuals in the population

$$m_i = \sum_{j=1}^{N} Sh(d_{ij})$$

The fitness function is adjusted (reduction based sharing) as

$$\overline{F_i} = \frac{F_i - Qm_i}{\text{rank}_\text{i}}$$

where

rank is the rank number
F_i is the nominal fitness value
Q is a weighting factor set by the user.

This algorithm rapidly converges and the solutions are distributed on the whole Pareto-optimal set.

Although there are many versions of evolutionary algorithms that are tailored to multi-objective optimisation, theoretical results about the convergence of GA are apparently not yet available. The result obtained [216] shows that a specific choice of the evolution rules is needed to assure the convergence with (probability one) to the actual Pareto-optimal set for a specific problem.

3.4.5 Unconstrained Minimisation

In the unconstrained minimisation problem we search for a minimum of the real–valued function $f(\mathbf{x})$

$$\begin{gathered} \min_{\mathbf{x}\in\Re^{n_{dv}}} \; f(\mathbf{x}) \\ \mathbf{x}_{low} \leq \mathbf{x} \leq \mathbf{x}_{up} \quad (\mathbf{x} \in \mathcal{X}) \end{gathered} \tag{3.35}$$

Several different methods are available to solve multidimensional unconstrained minimisation problems.

The main techniques that have been proposed for solving unconstrained optimisation problems are based on Newton's method. Newton's method requires [210] the computation of the gradient vector,

$$\nabla f(\mathbf{x}) = \left\{ \begin{array}{c} \frac{\partial f(\mathbf{x})}{\partial x_1} \\ \cdots \\ \cdots \\ \frac{\partial f(\mathbf{x})}{\partial x_{n_{dv}}} \end{array} \right\}$$

and the Hessian matrix,

$$\nabla^2 f(\mathbf{x}) = \left(\begin{array}{ccc} \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_{n_{dv}}} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_{n_{dv}} \partial x_1} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_{n_{dv}} \partial x_{n_{dv}}} \end{array} \right)$$

The basic Newton method minimises iteratively the quadratic function which represents the Taylor expansion of the vector function $f(\mathbf{x})$ in the neighbourhood of the point $\mathbf{x}_k$

$$Q_k(\mathbf{d}_k) = f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T \mathbf{d}_k + \frac{1}{2} \mathbf{d}_k^T \nabla^2 f(\mathbf{x}_k) \mathbf{d}_k$$

$$\mathbf{d}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$$

When the Hessian matrix $\nabla^2 f(\mathbf{x}_k)$ is positive definite, Q_k has a unique minimiser that can be obtained if

$$\frac{\partial Q}{\partial d_i} = 0 \quad (i = 1, \ldots, n_{dv})$$

which implies solving the linear system [210]

$$\nabla^2 f(\mathbf{x}_k)\mathbf{d}_k = -\nabla f(\mathbf{x}_k)$$

The next iterate is then $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k$. Convergence is guaranteed if the initial guess $\mathbf{x}_0$ is close to a local minimiser $\mathbf{x}^*$. Moreover, the rate of convergence is quadratic.

It is possible to enforce convergence when the initial guess $\mathbf{x}_0$ is not close to a minimiser $\mathbf{x}^*$ by using either a line–search or a trust–region approach. Line–search methods generate the iterate by setting $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$, where $\mathbf{d}_k$ is the search direction and α_k is selected so that $f(\mathbf{x}_{k+1}) < f(\mathbf{x}_k)$.

3.4.6 Simplex Method

A less efficient but frequently used method used to determine search directions is the simplex method due to Nelder and Mead [178]. The method requires only functions evaluations, not derivatives. A simplex is a figure consisting in n dimensions, of $n + 1$ points. In two dimensions the simplex is a triangle. Starting from a given initial simplex the method requires a series of steps. The point of the simplex where the function is largest is discarded and moved through the opposite face of the simplex to a lower point (reflection). When it is possible the method expands the simplex, to take larger steps (expansion). When it reaches a valley, the simplex can be contracted (contraction) in one or more directions. An appropriate sequence of such steps converges to a minimum of the function.

3.4.7 Sequential Unconstrained Minimisation Techniques

Penalty functions are used to transform inequality constrained optimisation problems into a sequence of unconstrained optimisation problems (sequential unconstrained minimisation techniques, SUMT). For this reason, the penalty function methods are sometimes referred to as indirect methods. The main reason for using penalty functions is that penalty functions are suitable for solving inequality and equality constrained optimisation problem (see (2.5)).

To use penalty functions, one transforms the constraints into *penalty functions* that are added to the cost function $f(\mathbf{x})$. There are two kinds of penalty functions: *exterior* and *interior*. Exterior penalty functions can be used for either equality or inequality constraints, while interior penalty functions can be used only for inequality constraints.

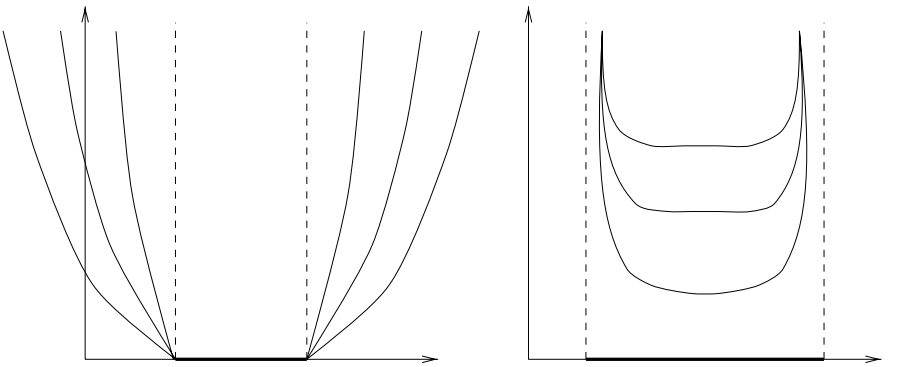


Fig. 3.13. Penalty functions: exterior (**a**), interior (**b**)

A typical set of penalty functions is shown in Fig. 3.13. The variety of penalty functions in the literature is almost endless. The remainder of this paragraph will concentrate on the introduction of some general interior and exterior penalty functions[7].

An exterior, inequality penalty function is the following:

$$\Phi(\mathbf{x}, r) = f(\mathbf{x}) + r \sum_{j=1}^{n_c} \langle g_j \rangle^z \tag{3.36}$$

where the bracket operator $\langle \cdot \rangle$ is $\langle g \rangle = \max[g, 0]$ and the exponent z is a non-negative constant and r is a positive small number that is sequentially increased. This formulation seems the most natural approach to penalty functions.

An interior inequality penalty function is

$$\Phi(\mathbf{x}, r) = f(\mathbf{x}) - r \sum_{j=1}^{n_c} \log(-g_j(\mathbf{x})) \tag{3.37}$$

over all $\mathbf{x}$ satisfying $g_j(\mathbf{x}) \leq 0$, r is a positive number that is sequentially decreased. The pseudo–objective function Φ is defined only in the feasible region. The starting point must be feasible.

The penalty function preferred [62] for combined equality and inequality constraints (see (2.5)) is

$$\Phi(\mathbf{x}, r) = f(\mathbf{x}) + \frac{1}{r^{1/2}} \sum_{i=1}^{n_{ec}} (g_i(\mathbf{x}))^2 - r \sum_{i=n_{ec}+1}^{n_c} \log(-g_i(\mathbf{x})) \tag{3.38}$$

[7] See [68, 206] for a more detailed discussion of the topic

over all $\mathbf{x}$ satisfying $g_i(\mathbf{x}) \leq 0$ $i = n_{ec}+1, \ldots, n_c$, r is a positive number that is sequentially decreased.

Obviously when all constraints are satisfied, $\Phi = f(\mathbf{x})$.

3.4.8 Method of Feasible Directions, Sequential Quadratic Programming

The sequential quadratic programming (SQP) method consists of three main steps:

- Updating of the Hessian matrix of the Lagrangian function,
- Quadratic programming problem solution,
- Line search and merit function calculation.

An overview of SQP methods can be found in [65, 81, 209]. Consider the constrained optimisation problem given by (2.5). A necessary condition to find the optimal value $\mathbf{x}^*$ is given by Eq. (2.7). This condition is the well-known Karush–Kuhn–Tucker condition (see Sect. 2.10.1). The second-order sufficient conditions are given by (2.8).

This numerical method uses a quadratic approximation of the Lagrangian function (Eq. (2.6)) and a linearisation of the constraints to find a solution of the optimisation problem.

We can formulate the following quadratic programming problem that is convex and satisfies the necessary and sufficient condition (see Sect. 2.10.1) and it is used by the SQP method to solve iteratively Eq. (2.5)

$$\begin{aligned} &\min_{\mathbf{d}_k \in \Re^{dnv}} Q(\mathbf{d}_k) = \frac{1}{2}\mathbf{d}_k^T H(\mathbf{x}_k)\mathbf{d}_k + \nabla f(\mathbf{x}_k)^T \mathbf{d}_k \\ &\nabla \mathbf{g}(\mathbf{x_k})\mathbf{d}_k \leq -\mathbf{g}(\mathbf{x}_k) \end{aligned} \tag{3.39}$$

where $Q(\mathbf{d}_k)$ is a quadratic approximation of the function $f(\mathbf{x}) - f(\mathbf{x}_k)$ in $\mathbf{x}_k$, $H(\mathbf{x})$ is a positive definite approximation of the Hessian of the Lagrangian function at iteration k and $\nabla \mathbf{g}$ is the gradient of the function $g(\mathbf{x})$ and it is a $n_c \times n_{dv}$ matrix. A good Hessian estimation is given by [209] and by the following algorithm

$$\begin{aligned} &H_{k+1} = H_k + \frac{\mathbf{q}_k \mathbf{q}_k^T}{\mathbf{q}_k^T \mathbf{s}_k} - \frac{H_k^T H_k}{\mathbf{s}^T H_k \mathbf{s}} \\ &\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k \\ &\mathbf{q}_k = \nabla f(\mathbf{x}_{k+1}) + \sum_{i=1}^{n_c} \eta_i \nabla g_i(\mathbf{x}_{k+1}) - \left(\nabla f(\mathbf{x}_k) + \sum_{i=1}^{n_c} \eta_i \nabla g_i(\mathbf{x}_k) \right) \end{aligned}$$

At each iteration of the SQP method solve the quadratic programming problem given by (3.39) and the $\mathbf{d}$ found is used to move in the design variable space

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha \mathbf{d}_k \tag{3.40}$$

By applying SQP method, a direction that will reduce the objective function without violating the constraints is searched. A direction which reduces the objective function is called a usable direction and is given by

$$\nabla f(\mathbf{x}_k) \cdot \mathbf{d}_k \leq 0 \tag{3.41}$$

A direction which does not cause active constraints violation upon move is a feasible direction

$$\nabla g_{active}(\mathbf{x}_k) \cdot \mathbf{d}_k \leq 0 \tag{3.42}$$

Directions satisfying both Eqs. (3.41) and (3.42) are usable and feasible. The parameter α in Eq. (3.40) is adapted to avoid constraint violation.

A non-linearly constrained problem can often be solved in fewer iterations than an unconstrained problem using SQP. One of the reasons for this is that, because of limits on the feasible area, the optimiser can make well–informed decisions regarding directions of search and step length.

3.4.9 Weighted Sum

One popular (but generally not recommended) scalarisation method to find the Pareto-optimal is the weighted sum of the objective functions. By minimising the weighted sum of the objective functions for different weight settings, many Pareto-optimal points can be generated. In mathematical form

$$\min_{\mathbf{x} \in \mathcal{F}} \lambda_1 f_1(\mathbf{x}) + \lambda_2 f_2(\mathbf{x}) + \ldots + \lambda_{n_{of}} f_{n_{of}}(\mathbf{x})$$

where

$$0 \leq \lambda_i \leq 1 \quad \sum_{i=1}^{n_{of}} \lambda_i = 1$$

It is obviously supposed that the objective functions are normalised. Notice that as objective functions are normalised, we set the weights as λ_i ($0 \leq \lambda_i \leq 1$). By varying the weights λ_i systematically various points on the Pareto-optimal set are generated. Note that the weighting coefficients are not directly related with importance of the objective functions but are only factors which when varied, locate solutions in $\mathcal{X}$. The (proper) Pareto-optimal set is obtained by altering the weighting coefficients.

For instance, considering a problem with two objective functions (Fig. 3.14), $\phi(\mathbf{x}) = \lambda_1 f_1(\mathbf{x}) + \lambda_2 f_2(\mathbf{x})$ is the expression of a straight line in the objective function space. The solution of the scalar problem $\min_{\mathbf{x} \in \mathcal{F}} \phi(\mathbf{x})$ is a non-dominated solution for the original multi-objective problem. We consider multiple problems varying λ_1 and λ_2 to try to construct the entire Pareto-optimal set.

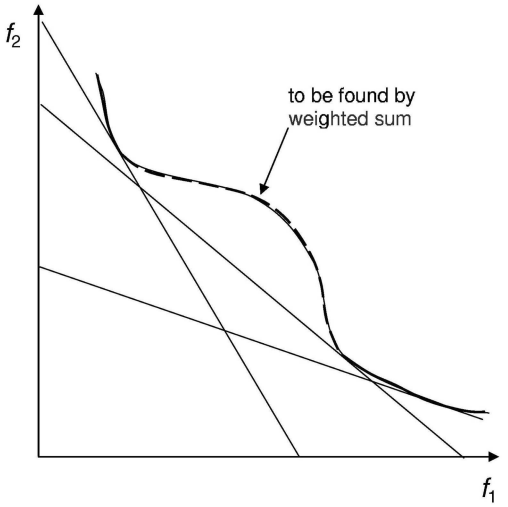


Fig. 3.14. Weighted sum of two objective functions. The Pareto-optimal set being not convex, the weighted method fails to find the whole Pareto-optimal set

This procedure has shortcomings, and it is usable only if the Pareto-optimal set is entirely convex (see Definition 2.3 and Sect. 3.4.10) (a property that may be difficult to check) [64, 237]. Only for linear programming problems the convexity requirement is satisfied, hence the weighted sum methods may be used to find the complete set of Pareto-optimal solutions.

3.4.10 Constraints Method

Constraints method is probably the most effective technique to compute the Pareto-optimal set. We consider only one objective function, the others are converted into constraints (see Chaps. 1, 2)

$$\min_{\mathbf{x}\in\mathcal{F}} f_1(\mathbf{x})$$

$$f_2(\mathbf{x}) \leq \epsilon_2 \quad f_3(\mathbf{x}) \leq \epsilon_3 \quad \ldots \quad f_{n_{of}}(\mathbf{x}) \leq \epsilon_{n_{of}}$$

The problem is reformulated as a standard NLP problem (2.5). The Pareto-optimal set is obtained by varying the constraints levels $\epsilon_i, \quad i = 2, \ldots, n_{of}$. This assures the existence of the solution for the constrained problem. If the problem is solved for all possible values of ϵ_i and the resulting solutions are unique, then this solutions constitute the entire Pareto-optimal set. If the solutions are not unique for some values of ϵ_i, then the Pareto points must be selected by direct comparison (i.e. applying the definition of Pareto-optimality, see Sect. 2.10.2).

Reconsidering the problem with two objective functions, we obtain

$$\min_{\mathbf{x}\in\mathcal{F}} f_1(\mathbf{x})$$

$$f_2(\mathbf{x}) \leq \epsilon$$

The minimum of $f_1(\mathbf{x})$ is the circle in Fig. 3.15. The admissible region is the grey region in figure. The minimum $f_1(x)$ is a non-dominated solution for the MOP problem. Varying the level of constraint ϵ we compute a different minimum for f_1 and a different non-dominated solution.

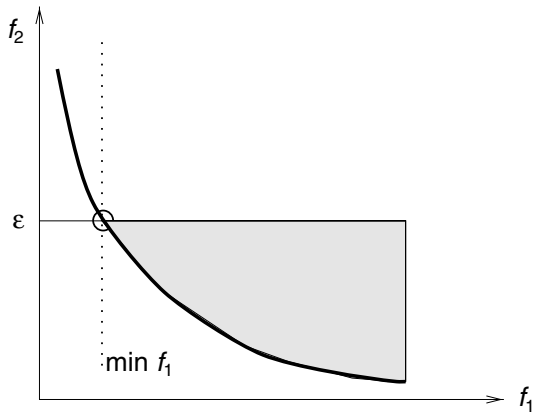


Fig. 3.15. Constraints method for a problem with two objective functions

Example: Comparison of Weighted Sum and Constraints Method

Considering a very simple problem [239], it is possible to show that the weighted sum method cannot be successfully applied when the Pareto-optimal set is not convex.

We have a cylindrical reservoir that is defined using two design variables: the radius r and the height h. We want to find the solution that has the maximum volume (minimum opposite of volume) and the minimum external surface:

$$\text{Volume} = \pi r^2 h = -f_1(r,h) \tag{3.43}$$

$$\text{Surface} = 2\pi rh + 2\pi r^2 = f_2(r,h) \tag{3.44}$$

Performing the scalarisation using the weighted method, we have

$$\phi(r,h,\lambda_1,\lambda_2) = \lambda_1 f_1(r,h) + \lambda_2 f_2(r,h) \quad \lambda_1,\lambda_2 \geq 0 \quad \lambda_1+\lambda_2 = 1 \tag{3.45}$$

Additionally, we check

$$\nabla(\phi(r^*, h^*)) = 0 \tag{3.46}$$

$$\text{eig}(H(r^*, h^*)) > 0 \tag{3.47}$$

We check for the first-order condition (3.46) and we are able to find stationary solutions (r^*, h^*). We check for the second order condition (3.47) analysing the eigenvalues of the Hessian matrix H at the stationary points (r^*, h^*). The Hessian matrix must be positive definite. Some eigenvalues are not positive, so the stationary points (r^*, h^*) are not minima. We are not able to find Pareto-optimal points, but this conclusion is wrong.

Resorting to the constraint method, we reformulate the problem maximising the volume and constraining the external surface.

$$\min_{r,h} f_1(r, h)$$
$$f_2(r, h) \leq \epsilon$$

We check for the first-order condition, we find the stationary points, we check for the second-order condition and we see that we are able to find minimum points.

What is the reason for this? We can plot the Pareto-optimal set in the design variables space, radius and height, and the Pareto-optimal set in the objective functions space: f_1 and f_2. The Pareto-optimal set is not convex (Fig. 3.16) and the weighted sum method fails.

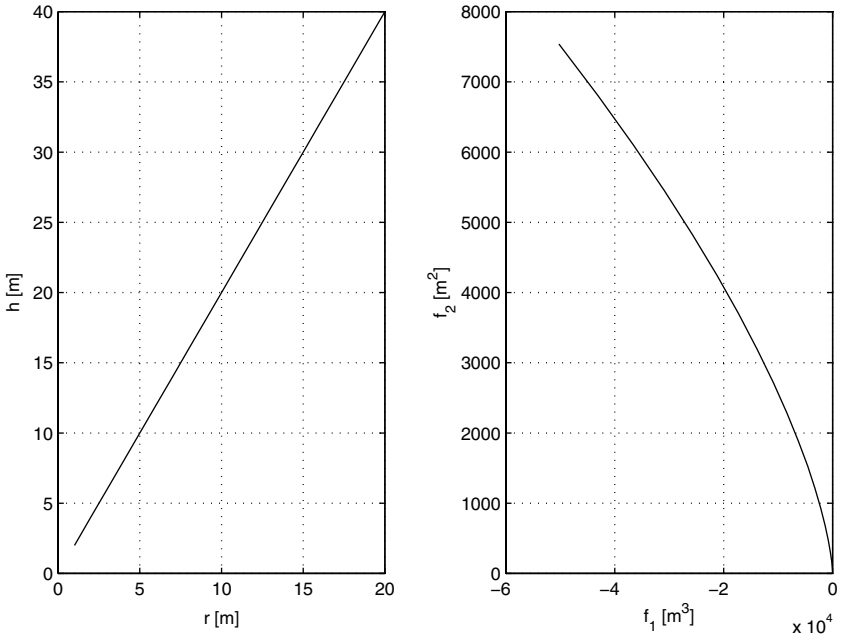


Fig. 3.16. Pareto-optimal set in the design variables and objective functions space

The expressions of the Pareto-optimal set are given by

$$h = 2r$$
$$V = \frac{S^{\frac{3}{2}}}{(2\pi)^{\frac{1}{2}} 3^{\frac{3}{2}}}$$

3.5 Design Synthesis – Choosing a Final Design Solution

Given the set of Pareto-optimal solutions, the decision maker (DM) has to select a preferred design solution within the Pareto-optimal set. Every Pareto-optimal solution is equally acceptable. The final selection requires information that is not contained in the objective functions. The DM preference structure may be complex.

3.5.1 Utility Function

A *utility function* can be used to represent the preferences of the designer among the Pareto-optimal solutions:

$$U(\mathbf{f}(\mathbf{x})) : \Re^{n_{of}} \rightarrow \Re \tag{3.48}$$

The function U maps the objective function vector to a scalar figure of merit. The best design solution is found by maximising U.

The following two theorems show the relationship between the utility function and the Pareto-optimal solutions. Proofs can be found in [166].

Theorem 3.4. *Let U be coordinate–wise decreasing*[8]*. Then if $\mathbf{x}^*$ is optimal for U it is Pareto-optimal.*

Theorem 3.5. *Let $\mathbf{x}^*$ be Pareto-optimal. Then there exists a coordinate–wise decreasing utility function U such that $\mathbf{x}^*$ is optimal.*

Generally, the function U is selected to be strongly decreasing [166]. Two solutions are said to be on the same *indifference curve* if the designer finds them equally good. The final solution of a multi-objective optimisation problem is the Pareto-optimal solution where the indifference curve is tangent to the Pareto-optimal set. The procedure is shown in Fig. 3.17 for a bidimensional problem.

This easy way to select a preferred design solution from the Pareto-optimal set cannot be used in practice. It is generally very difficult for a designer to give the mathematical expression U of his preference.

[8] A coordinate–wise decreasing utility function implies that all objective function are in minimisation form and that monotonicity holds for each of them (not always true) in actual applications

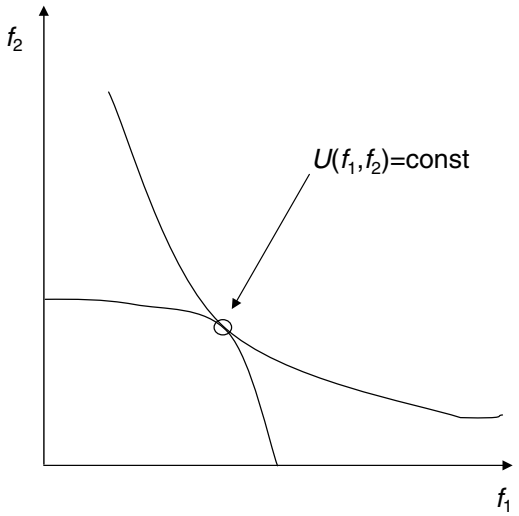


Fig. 3.17. Indifference curve into the objective function space f_1, f_2

The inability to encode the underlying utility function reliably is demonstrated in [179]. The paper shows that encoding methods that theoretically should produce identical utility functions generally fail.

One important thing to take into account in practice is that the aspirations of the DM may change during the solution process. It is difficult for the DM to know the real utility function before starting the optimisation process and without knowing the range of variation of the performances (i.e. objective functions) of the system under analysis.

3.5.2 Lexicographic Ordering

Lexicographic ordering requires the DM to order the objective functions according to their relative importance. Suppose that we have a hierarchy among the objective functions $f_i, \quad i = 1, \ldots, n_{of}$ in terms of their importance so that $f_1 \succ f_2 \succ \ldots \succ f_{n_{of}}$ ($\succ$ means more important than). According to this hierarchy, we can order the values of f_1 then the values of f_2 and so on. Such ordering is always possible, it provides a comparison between all elements, and it always leads to a unique optimal solution. The justification for using lexicographic ordering is its simplicity and the fact that DM usually make decision successively.

An example of lexicographic ordering is shown in Fig. 3.18 for a problem with two objective functions. The objective function f_1 is assumed to be the most important. After minimising f_1 there are two solutions left and after minimising the second objective function f_2 only y^0 is left.

Lexicographic ordering correspond to the weighting method when the weighting coefficients are very different in magnitude. The justification of

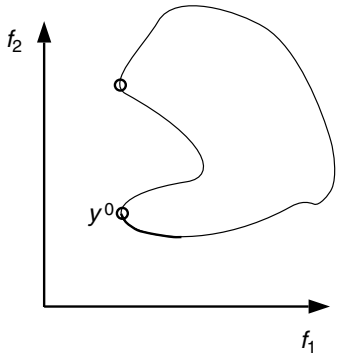


Fig. 3.18. Lexicographic ordering into the criterion space f_1, f_2

lexicographic ordering is in its simplicity and in fact that DMs make decisions step-by-step. DMs often have difficulties to order the objective functions by relative importance. This method can be used as part of the following solution method.

3.5.3 Goal Programming

The basic concept of goal programming is that the DM specifies optimistic target for the objective function to be attained (Fig. 3.19). Any deviation from this targets are minimised. These targets forms a *goal*. Goals are represented by vector $\tilde{\mathbf{y}}$. The goals are in the same form of constraints. After specifying

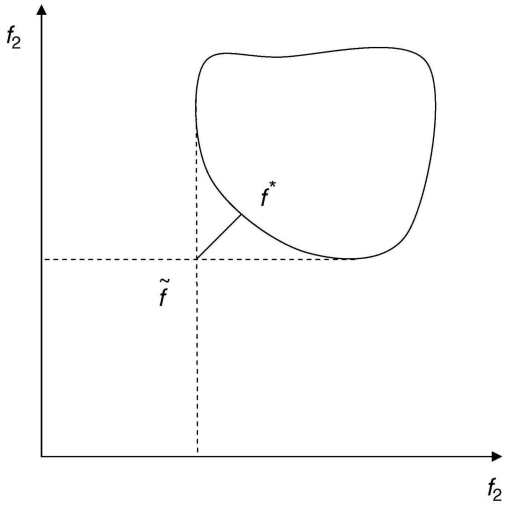


Fig. 3.19. Goal programming method for a problem with two objective functions

the goals, the following task is to minimise the deviations of the objective functions from these goals.

Two approaches are commonly used:

- *weighted* approach, where the weighted sum of the deviational variables is minimised;
- *lexicographic* approach, where the DM defines an order of importance of the objective function and the optimisation is performed by considering this scale.

In the weighting approach the DM must specify information about the weights. The greater the weighting coefficient is, more important is the objective function. The weights can be also defined as $w_i = 1/|\tilde{y}_i|$.

The problem can be written as

$$\begin{aligned} &\min_{\mathbf{x}\in\Re^{n_{dv}}} \sum_{i=1}^{n_{of}} w_i \, |f_i(\mathbf{x}) - \tilde{y}_i| \\ &\text{subject} \quad \text{to} \\ &\mathbf{x} \in \mathcal{F} \end{aligned} \tag{3.49}$$

Goal programming is a very widely used and popular solution method for practical multi-objective optimisation problems. One reason is that goal–setting is an understandable and easy way of making decisions. The specification of the weighting coefficients or the lexicographic ordering may be more difficult. The weights do not have so direct an effect on the solution obtained as in the a priori weighting method (Sect. 3.4.9).

It may be difficult to specify the weights because they have no direct physical meaning. Presenting the ranges of the Pareto-optimal set, or at least the ideal objective vector, to the DM may help in the selection of the goal vector. One must be careful with the selection of the aspiration levels so that the Pareto-optimality of solution must be guaranteed.

3.5.4 Preference Via Trained Artificial Neural Network

Artificial Neural Networks (ANNs) have been employed effectively [241] to capture the DM's preference structure within an interactive framework. The DM initially reveals his preferences by evaluating a sample of non-dominated (i.e. Pareto-optimal) design solutions. This information is used to train an ANN. At each iteration, the trained ANN is used to select a subset of 'most preferred solutions' from a large set of additional non-dominated solutions.

3.5.5 Min–Max Methods

If the minimum objective function value of goal programming equals zero, the solution obtained may not be Pareto-optimal. This happens when all the aspiration levels are feasible.

To overcome this shortcoming we introduce another form of goal programming, the min–max method.

The min–max method is found to be appropriate in dealing with multi-objective programming problems, also in the presence of non-convex criteria space.

The method is defined in terms of a function of relative deviations given by

$$z_i(\mathbf{x}) = \frac{|f_i(\mathbf{x}) - \tilde{f}_i|}{|\tilde{f}_i|} \tag{3.50}$$

where $z_i(\mathbf{x})$ is the deviation at point $\mathbf{x}$, $f_i(\mathbf{x})$ denotes the value of the ith objective function at $\mathbf{x}$, and $\tilde{f}_i$ is the optimum (minimum) of the ith function. The min–max method proceeds as follows. A point, which gives a satisfactory $f(\mathbf{x})$, is selected knowing all the separate optima of all the objective functions involved. The ith objective function for which z_i is maximum is found, then the function $z_i(\mathbf{x})$ is minimised. By setting

$$\min_{\mathbf{x}\in\mathcal{X}} \quad \max_{i\in\{1,\ldots,n_{of}\}} \quad z_i(\mathbf{x}) \tag{3.51}$$

generally a unique solution is found. Figure 3.20 shows how the method works. The weighted min–max method [191] differs from the previous in that it assigns different weights to the relative deviations of the different objective functions. Each different set of weights therefore identifies a different Pareto-optimal solution. By varying the weighting coefficients, all design of interest may be obtained from both convex and non-convex problems. Contrary to the simple weighting method, weights in this method reflect the true relative importance of the various criteria.

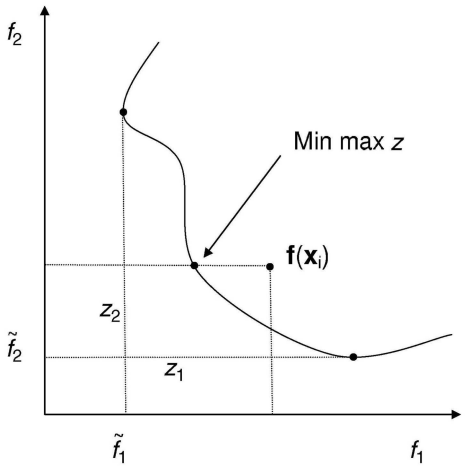


Fig. 3.20. Min–max method for a problem with two objective functions

3.5.6 Hierarchical Optimisation Method

The hierarchical optimisation method is applicable in the situations in which the criteria can be ordered in terms of importance. Let the numbering $1 \ldots n_{of}$ reflect this ordering in the sense that the first criterion is the most important and the n_{of}th is the least important. A detailed discipline of the topic is given in [200, 256].

Keeping the order we minimise each objective function separately.

Definition 3.6. *Find the minimum for the primary criterion, i.e. find* $\mathbf{x}^{(1)} = \{x_1^{(1)}, x_2^{(1)}, \ldots, x_{n_{dv}}^{(1)}\}^T$ *such that*

$$f_1(\mathbf{x}^{(1)}) = \min_{\mathbf{x} \in \mathcal{X}} f_1(\mathbf{x})$$

For $i = 2, \ldots, n_{of}$*, find the minimum of the* i*th objective function, i.e. find* $\mathbf{x}^{(i)} = \{x_1^{(i)}, x_2^{(i)}, \ldots, x_{n_{dv}}^{(i)}\}^T$ *such that*

$$f_i(\mathbf{x}^{(i)}) = \min_{\mathbf{x} \in \mathcal{X}} f_i(\mathbf{x})$$

with additional constraints

$$f_{j-1}(\mathbf{x}) \leq (1 + \frac{\epsilon_{j-1}}{100}) f_{j-1}(\mathbf{x}^{(j-1)})$$

for $j = 2, 3, \ldots, n_{of}$*, where* ϵ_{j-1} *are the assumed coefficients of the function increments given in per cent.*

By changing the value of ϵ_{j-1}, all the Pareto-optimal set can be found. This method is also very useful when the DM wants to pick a final optimal solution from the Pareto-optimal set.

3.5.7 Normal–Boundary Intersection Method

Normal–boundary intersection technique is successful in the identification of the shape of the Pareto-optimal set. A detailed description of the topic is given in [38].

Let us consider the generic MOP formulation (2.10). Let $\mathbf{x}_i^*$ be the global minimisers of $f_i(\mathbf{x})$, $i = 1, \ldots, n_{of}$ and $\boldsymbol{\varphi}_i^* = \mathbf{f}(\mathbf{x}_i^*)$, $i = 1, \ldots, n_{of}$. Let $\boldsymbol{\varphi}^*$ denote the vector of minima of each function (Utopia point, see Chap. 2). Let $\boldsymbol{\Phi}$ be the $n_{of} \times n_{of}$ matrix whose ith column is $\boldsymbol{\varphi}_i^* - \boldsymbol{\varphi}^*$. The set of points that are convex combinations of $\boldsymbol{\varphi}_i^* - \boldsymbol{\varphi}^*$ are referred as CHIM [38] (see Definition 2.3). Given a convex combination vector $\boldsymbol{\beta}$, $\boldsymbol{\Phi\beta}$ represents a point into the CHIM. The CHIM for a problem with two objective functions (Fig. 3.21) is the line joining $f(\mathbf{x}_1^*)$ and $f(\mathbf{x}_2^*)$, i.e.

$$\left\{ \boldsymbol{\Phi\beta} : \boldsymbol{\beta} \in \Re^{n_{of}}, \sum_{i=1}^{n_{of}} \beta_i = 1, \ \beta_i \geq 0 \right\}$$

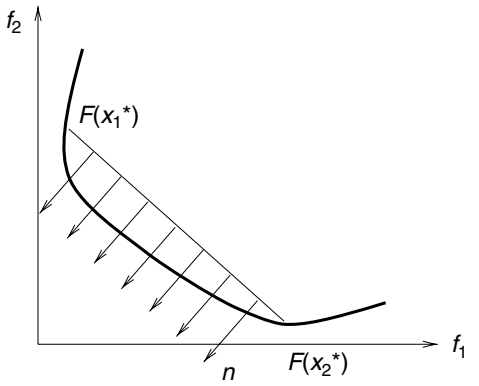


Fig. 3.21. CHIM for a problem with two objective functions

The idea behind normal–boundary intersection is to consider a uniform set of points on the CHIM, and find the intersection point between the Pareto-optimal set and a set of parallels straight lines emanating from the chosen set of points on the CHIM.

Let $\mathbf{n}$ denote the unit normal vector to the CHIM simplex pointing towards the origin. $\boldsymbol{\Phi\beta} + t\mathbf{n}$, $t \in \Re$, represents the set of points on the normal. The point of intersection between the normal and the boundary of the set of points $\mathbf{f}(\mathbf{x})$ closest to the origin is identical to the solution of the following problem:

$$\begin{aligned}
&\max_{x,t} t \\
&\text{s.t.} \\
&\boldsymbol{\Phi\beta} + t\mathbf{n} = \mathbf{f}(\mathbf{x}) - \boldsymbol{\varphi}^* \\
&g_i(\mathbf{x}) \leq 0, \quad i = 1, \ldots, n_c \\
&\mathbf{x} \in \mathcal{X}
\end{aligned}$$

$\boldsymbol{\beta}$ provides a parametrisation of the Pareto-optimal set.

The weakness of the approach is the fact that it may produce non-Pareto-optimal solutions for non-convex problems.

3.6 Interactive Methods

In interactive methods, the decision maker (DM) works together with the optimisation software. After every iteration some information is given to the DM and he (or she) provides some information to the program. Consistency of the response of the DM is the most important factor guaranteeing the success of the procedure.

These methods work according to the hypothesis that the DM is unable to indicate 'a priori' any preference information because of the complexity of

the problem. However, the DM is able to give some preference information as the search moves on. The DM then learns about the problem as he faces different possible problem solutions.

Advantages of these types of methods are as follows:

- there is no need for 'a priori' preference information,
- only local preference information is needed,
- it is a learning process where the DM gets a better understanding of the problem,
- as the DM takes an active part in the search it is more likely that he accepts the final solution.

The disadvantages are the following:

- the solutions are depending on how well the DM can articulate his preferences,
- a high effort is required from the DM during the whole search process,
- the solution is depending on the preferences of one DM,
- if the DM changes his preferences or if there is a change of DM, the process has to be restarted,
- a large number of objective functions may make the interactive method impractical.

These methods usually progress by changing weights in weighting formulations, by progressively reducing the search space or by changing the search direction based on input from the DM.

In interactive methods, the DM works together with an interactive computer program. A solution pattern is formed and repeated several times. After every iteration, some information is given to the DM and he is asked to answer some questions or provide some other type of information. The DM does not have to know any global preference structure.

After a number of iterations every interactive method should yield a solution that the DM can be satisfied with.

The basic steps in interactive algorithms can be expressed as

1. find an initial feasible solution,
2. interaction with the DM,
3. obtain a new solution. If the new solution (or one of them) or one of the previous solutions is acceptable to the DM, stop. Otherwise, go to step 2.

The main problem in designing an interactive method is to find the most effective method for the software to communicate with the DM. Several interactive methods are reported in the literature, refer to [166] for a comprehensive review.

3.6.1 Interactive Computation of the Pareto-optimal Solutions and Pareto-optimal Set Boundaries Through Pareto Sensitivity Analysis

As already indicated in [87, 88, 161, 229] and in Chap. 1, the optimisation method can be based on the approximation of the physical model responses $\mathbf{f}$. By using an approximation model we can obtain a set of approximated Pareto-optimal solutions. Every Pareto-optimal solution is equally acceptable but one simple final design solution should be obtained. We need a DM to select one solution out of the Pareto-optimal set. As already seen, a good strategy is to take a first starting solution that can be refined, by taking a priori designer's preferences in a mathematically consistent manner by considering some targets or goals to achieve on objective function set given by the vector $\tilde{\mathbf{y}}$. This target may be a feasible solution or not. At this point we can start to search the objective functions that minimise the absolute deviations from their targets.

The criterion, then, is to minimise the sum of the absolute values of the differences between target values and actually achieved values scaled by the interval of variation of the objective functions.

The design solution found with this method can be used as a starting point for a Pareto-sensitivity analysis.

By stopping the design process at this point we have some limitations:

- the solution found requires a priori selection of targets. In some cases we may have no information about the performances that can be achieved by our system and we may have no idea about the target parameters to put in the utility function;
- the solution found provides information for only one design solution;
- the solution found provides no information about trade-off analysis in the neighbourhood of the current solution.

An interactive choice of the final solution based on the analysis of the Pareto-optimal set can be useful to overcome these limitations. Given a variation in one objective function (f_j), Pareto sensitivity provides information on variation on the other objective functions $(\partial f_i/\partial f_j)$ in a given direction. Note that a given Pareto-optimal surface may have infinite tangent directions.

The Pareto sensitivity analysis presented here is based on the idea that we can consider a smooth function $\mathbf{m}(\mathbf{x}) : \Re^{\nu} \rightarrow \Re^{\chi}$ $(\nu > \chi)$. This function defines a surface

$$\mathbf{m}(\mathbf{x}) = 0 \tag{3.52}$$

It is possible to generate a normalised tangent vector $\mathbf{v_x}$ to the surface defined by the implicit function (3.52) by considering the relation

$$\nabla_{\mathbf{m}}(\mathbf{x})^{\mathbf{T}}\mathbf{v_x} = \mathbf{0} \tag{3.53}$$

where ∇ represents the Jacobian taken with respect to the variables $\mathbf{x}$. To retrieve a tangent vector to the Pareto-optimal surface we will use the Karush–Kuhn–Tucker (KKT) [166] condition as a necessary condition for optimality. The KKT condition is given for the optimisation problem (2.10) by

$$\begin{aligned}
&\boldsymbol{\nabla \mathbf{f}(\mathbf{x})\lambda + \nabla \mathbf{g}(\mathbf{x})\ \mu = 0}\\
&\mu_i g_i(\mathbf{x}) = 0 \quad i = 1, \ldots, n_c\\
&\mu \geq 0\\
&\boldsymbol{\lambda} \geq 0
\end{aligned} \tag{3.54}$$

where $\boldsymbol{\lambda}$ (n_{of} components) and $\boldsymbol{\mu}$ (n_c components) are vector of parameters. If we have a point $\mathbf{x}^*$ that is globally Pareto-optimal this condition is satisfied [166]. If a design solution $\mathbf{x}^*$ is globally Pareto-optimal, it implies that it is locally Pareto-optimal [166]. This assumption allows to perform the sensitivity analysis on the Pareto surface.

Let $\sum_{i=1}^{n_{of}} \lambda_i - 1 = 0$, if at a given Pareto-optimal point $\mathbf{x}^*$ the inequality constraints g_i are active, the values of μ_i are not equal to zero. We will consider only $n'_c \leq n_c$ active constraints.

Also we will consider the $n'_{of} \leq n_{of}$ objective function that have $\lambda_i \neq 0$. By applying the relation (3.53) to the KKT condition we have

$$\begin{aligned}
&\sum_{i=1}^{n'_{of}} \lambda_i \nabla^2 f_i(\mathbf{x}) \mathbf{v}_{\mathbf{x}} + \sum_{i=1}^{n'_c} \mu_i \nabla^2 g_i(\mathbf{x}) \mathbf{v}_{\mathbf{x}} + \nabla \mathbf{f}(\mathbf{x}) \mathbf{v}_{\boldsymbol{\lambda}} + \nabla \mathbf{g}(\mathbf{x}) \mathbf{v}_{\boldsymbol{\mu}} = 0\\
&\nabla \mathbf{g}(\mathbf{x})^T \mathbf{v}_{\mathbf{x}} = 0\\
&\sum_{i=1}^{n'_{of}} v_{\boldsymbol{\lambda} i} = 0
\end{aligned} \tag{3.55}$$

∇^2 denotes the Hessian of a function. This is a linear system of $n_{dv} + n'_c + 1$ equations on the variables $\mathbf{v} = \left(\mathbf{v}_{\mathbf{x}}^T \mathbf{v}_{\boldsymbol{\lambda}}^T \mathbf{v}_{\boldsymbol{\mu}}^T\right)^T$ ($n_{dv} + n'_c + n'_{of}$ unknown). There are $n_{dv} + n'_c + n'_{of} - n_{dv} - n'_c - 1 = n'_{of} - 1$ free parameters.

These free parameters can be used to add other constraints to the system (3.55). Constraint can be on $\mathbf{v}_{\mathbf{x}}$, $\mathbf{v}_{\boldsymbol{\lambda}}$ and $\mathbf{v}_{\boldsymbol{\mu}}$. We may impose to keep some components of $\mathbf{x}$, $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ unchanged. This condition is imposed by setting some components of $\mathbf{v}$ equal to zero. Other conditions can be on the direction of $\mathbf{v}_{\mathbf{x}}$ that must be close to a chosen objective function f_i that has the greatest improvement. This is obtained by solving the problem (3.55) by adding the condition $\min \nabla \mathbf{f_i}(\mathbf{x})^T \mathbf{v}_{\mathbf{x}}$. In this second case we have to solve a single objective linear minimisation problem [273].

The values of $\mathbf{v}_{\mathbf{x}}$ can be used to find the sensitivity of objective functions with respect to another objective function f_i

$$\partial f_j/\partial f_i = \nabla \mathbf{f_i}^T \mathbf{v_x}/\nabla \mathbf{f}_j^T \mathbf{v_x} \quad j = 1, \ldots, n'_{of} \tag{3.56}$$

A trade-off among objective functions exists if the computed derivative $\partial f_j/\partial f_i$ is negative. The magnitude of the trade-off is given by the magnitude of this derivative. Positive values of these derivatives indicate that there are no trade-offs between the objective functions.

The computation of $\partial f_j/\partial f_i$ can be used by the designer to evaluate a preferred direction to refine the actual design solution.

At this point the designer can use the information found to generate alternative solutions to the current solution. The designer can choose a direction to improve one objective function through a predictor–corrector framework. He has to consider that the Pareto surface is highly non-linear, non-smooth and has discontinuities so the prediction step must not be very large. $\Delta\epsilon$ being the improvement of the objective function f_i, we can define a prediction step as

$$h = -\Delta\epsilon/(\nabla \mathbf{f_i}^T \mathbf{v_x}) \tag{3.57}$$

We can generate a tangent prediction by using

$$\tilde{\mathbf{x}}_{i+1} = \mathbf{x}_i + h\mathbf{v_x} \tag{3.58}$$

After this step we assume that $\tilde{\mathbf{x}}_{i+1}$ is close to the Pareto-optimal surface. The problem

$$\begin{aligned} &\min_{\mathbf{x}} f_i(\mathbf{x}) \\ &f_j(\mathbf{x}) \leq f_j(\tilde{\mathbf{x}}) \quad j = 1, \ldots, n_{of} \quad j \neq i \\ &\mathbf{g}(\mathbf{x}) \leq 0 \end{aligned} \tag{3.59}$$

is then solved with a few iterations by using an SQP optimisation method. By solving this problem the designer obtains an alternative to the current solution chosen. The found solution is checked by comparison with the approximated Pareto-optimal set – obtained by using directly the definition of non-dominated solution (2.7) and low discrepancy sequences (see Sect. 3.4.2) – found before starting Pareto sensitivity analysis and interactive decision making. If this alternative solution is dominated with respect to the approximated Pareto-optimal set, the nearest in the approximated Pareto-optimal set is considered as new alternative solution, advising the designer that the solution found is not a solution found by solving the problem (3.59). Figure 3.22 summarises the proposed interactive decision strategy.

The Pareto-optimal set can be given in the space of the objective functions or in the space of design variables. Usually [30,166], many efforts are performed to compute the Pareto-optimal set in the objective function space. However, the accurate computation of the boundaries of the Pareto-optimal set in the design variable space is very important for the designer.

Pareto sensitivity analysis in conjunction with low discrepancy sequences search can be useful to solve these kinds of problems. Figure 3.23 shows how

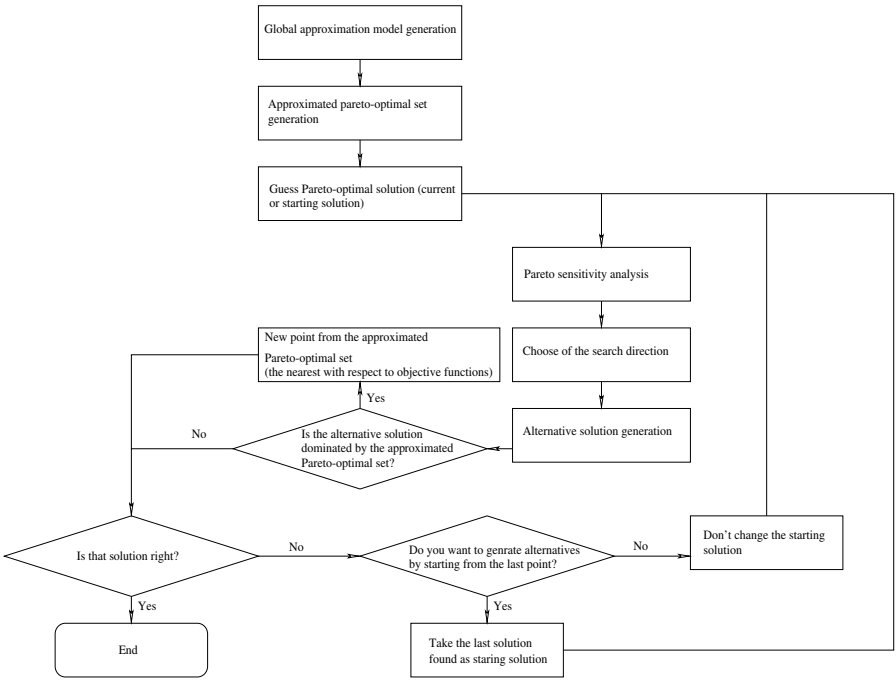


Fig. 3.22. Flow chart summarising Pareto sensitivity analysis and interactive decision making

to compute the boundaries of the Pareto-optimal set in the design variable space (multi-objective optimisation problem with two design variables). A Pareto-optimal solution denoted by $\Box$ is taken from a set of non-dominated solutions obtained with a Low Discrepancy Sequence search (Sect. 3.4.2) [184, 246]. This solution is taken as a starting solution to solve reformulated optimisation problems. The solutions denoted by $\circ$ are found by solving the following two optimisation problems by means of a modified SQP (Sect. 3.4.7) [206] method

$$\begin{aligned} &\min_{x_k,\boldsymbol{\lambda}} x_k \\ &x_i = x_i^{\Box} \quad i = 1,\ldots,k-1,k+1,\ldots,n_{dv} \\ &\mathbf{x} \in \mathcal{P} \end{aligned} \tag{3.60}$$

and

$$\begin{aligned} &\max_{x_k,\boldsymbol{\lambda}} x_k \\ &x_i = x_i^{\Box} \quad i = 1,\ldots,k-1,k+1,\ldots,n_{dv} \\ &\mathbf{x} \in \mathcal{P} \end{aligned} \tag{3.61}$$

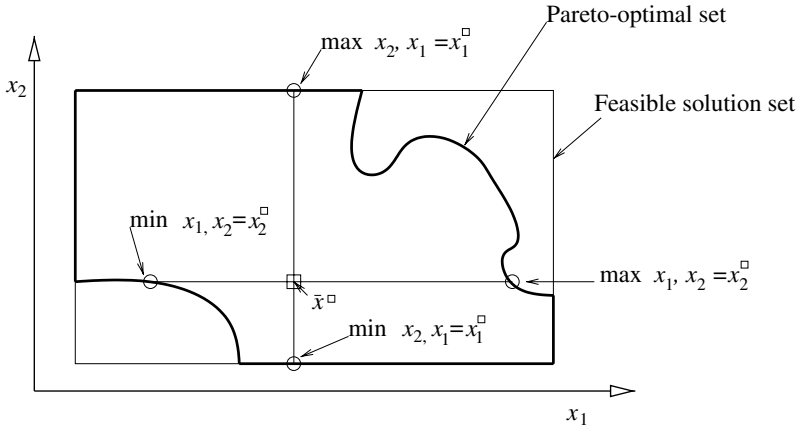


Fig. 3.23. Example of Pareto-optimal set boundary obtained in design variable space for a multi-objective optimisation problem with two design variables. □ is a Pareto-optimal solution taken from an initial approximated Pareto-optimal set. ○ are the solution found by applying the method presented in Sect. 3.6

At every step of the SQP algorithm (Sect. 3.4.7), the non-dominance of new points found is checked, comparing the new point found with the set of non-dominated points found by using low discrepancy sequences. The introduced method is particulary suitable to be integrated with global approximation techniques where the relationship between design variables and objective functions are approximated [5, 84, 85, 92, 104, 158], (see Sect. 4). The most widely used approximation methods are based on multi-layer perceptron neural networks (MLPNN) [88, 106, 161, 202] (see Sect. 4.6.1) and radial basis function neural networks (RBFNN) [159, 229] (see Sect. 4.6.2). These models are continuous and one-time differentiable so the method can be applied to these functions too.

3.7 Symbolical Derivation of PO Sets

3.7.1 Theorems of Monotonicity (Optimisation Problems with Constraints)

Before applying the constraints method to problems to be solved symbolically, monotonicity analysis has to be applied for determining constraint activity.

The following theorems [165, 200] (Theorems of Monotonicity) apply to the problem of finding the optimal compromise between objective functions.

If the objective functions and all inequality constraints are globally monotonic with respect to all design variables $x_i > 0$ then

Theorem 3.7. *If the variable x_i is explicitly represented in the objective function to be minimised, then there exists at least one active constraint with opposite monotonicity with respect to x_i.*

Theorem 3.8. *A variable x_i that is not explicitly represented in the objective function must either only be contained in constraints that are inactive, or else there must exist at least two active constraints having opposite monotonicities with respect to x_i.*

3.7.2 Theorems of Monotonicity (Optimisation Problems without Constraints)

If we consider two conflicting objective functions f_i which are analytical functions of two system model's design variables x_i, it is possible to apply directly the constraints method (see Sect. 3.4.10) to find the analytical expressions of both the optimal objective functions and the optimal design variables, i.e. one can find

- $f_1^* = f_1^*(f_2^*)$, i.e. the analytical expression which gives the optimal value of the objective function f_1 when f_2 is at its best (or vice versa, which is conceptually the same, $f_2^* = f_2^*(f_1^*)$)
- $x_1^* = x_1^*(x_2^*)$ (or vice versa $x_2^* = x_2^*(x_1^*)$), i.e. the analytical expression which gives the optimal values of the design variables x_1 and x_2 corresponding to f_1^* or f_2^*.

$f_1 = f_1(x_1, x_2)$ and $f_2 = f_2(x_1, x_2)$ being the objective functions and x_1 and x_2 the system model's design variables, the procedure to find the analytical expressions $f_1^* = f_1^*(f_2^*)$ and $f_2^* = f_2^*(f_1^*)$ is the following:

a) from the mathematical expression $f_1 = f_1(x_1, x_2)$ the expression $x_2 = x_2(f_1, x_1)$ is derived by fixing the value of f_1;
b) by substituting the expression derived at point a) in the expression $f_2 = f_2(x_1, x_2)$ the expression $f_2 = f_2(f_1, x_1)$ is obtained;
c) the minimum of f_2 is searched for by setting to zero the following derivative

$$\frac{d\, f_2(f_1, x_1)}{d\, x_1} = 0$$

and checking that

$$\frac{d^2\, f_2(f_1, x_1)}{d\, x_1^2} > 0$$

this corresponds to the search of the minimum of the objective function f_2, while the objective function f_1 is kept constant; from the expression of the first derivative, the expression $x_1 = x_1(f_1)$ can be obtained;

d) the expression $x_1 = x_1(f_1)$ is substituted into $f_2 = f_2(f_1, x_1)$ and by this way it is possible to get the expression $f_2^* = f_2^*(f_1^*)$ which defines the relationship between the two optimal objective functions;
e) the equation $f_2^* = f_2^*(f_1^*)$ is the image in the plane (f_1–f_2) of the equation $x_1^* = x_1^*(x_2^*)$ in the plane (x_1–x_2). $x_1^* = x_1^*(x_2^*)$ may be obtained by substitution.

This method is valid if there are no local Pareto-optimal solutions (see Sect. 2.10.2).

3.8 Illustrating the Pareto-optimal Set

Graphical illustrations help the decision maker to evaluate different PO solutions to make the choice of a final PO solution relatively easy.

If we have only two objective functions, the Pareto-optimal set can be plotted on a plane. If we have three objective functions a common way to illustrate the Pareto-optimal set is to draw a two-dimensional plot with fixed values assigned to the third objective function. With more than three objectives the graphical representation becomes more difficult to be interpreted.

Value paths or bar charts (see Fig. 3.24) have been extensively employed.

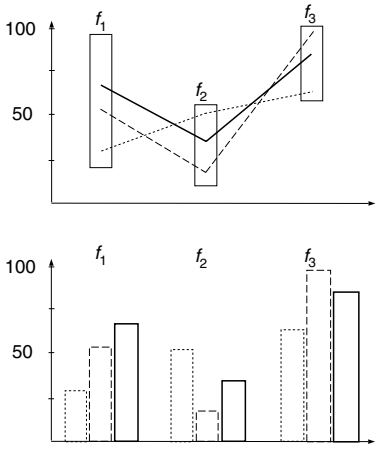


Fig. 3.24. Value paths (*top*) and bar chart (*bottom*). Design problem with three objective functions f_1, f_2, f_3. Three Pareto-optimal solutions are indicated. The bars in the value paths plot show the ranges of the objective functions in the Pareto-optimal set

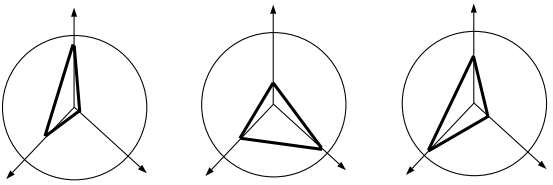


Fig. 3.25. Polar coordinate plot for a problem with three objective functions f_1, f_2, f_3. Three Pareto-optimal solutions are indicated

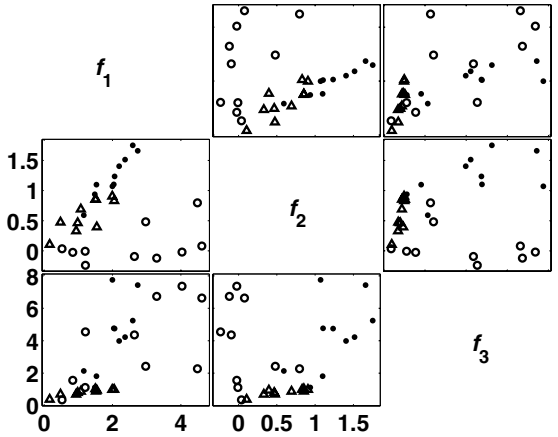


Fig. 3.26. Scatterplot matrix for a problem with three objective functions f_1, f_2, f_3. Three Pareto-optimal solutions subsets are shown (*circle, triangle, square*)

A polar coordinate system (see Fig. 3.25) can be used if the ideal and nadir objective functions are known. It is evident that a design solution is better if the area of the star is smaller (minimisation problem).

The scatterplot matrix is a square matrix ($n_{of} \times n_{of}$) of plots each representing one objective function pair (see Fig. 3.26).

4

Global Approximation

A major concern in all optimal design studies is the time required for the evaluation of the objective functions during the optimisation process. Although the computers continue to improve (steadily increasing computational power), it is still (and it will be always) important to develop proper computer programs (software) to save computation time for numerical analysis. As this time-saving can be dramatically important, significant emphasis is placed on investigating and developing techniques with the intent of minimising the computational effort required for the optimal design process.

Nowadays methods based on approximation concepts (see Sect. 1.2.5) take dominant position in the solution of complex design optimisation problems. The physical model is substituted by another purely mathematical model which is able to give computational results very quickly. This purely mathematical model is often the mathematical approximation of physical models (called *global approximation,* see Sect. 1.2.5) used in engineering optimisation when the computational effort is prohibitive.

Global approximation techniques provide several advantages, filtering of numerical noise associated with the computation, insight on the entire design space, detection of errors due to poor modeling (correction of poor-accuracy analyses).

Depending on the range of its applicability, the mathematical approximation of a physical model can be classified as local (valid in the vicinity of a design point) or global (valid in the whole design space). Local approximations are usually based on the Taylor series expansion at current point using a function value and its first derivative. Global approximations are normally based on the information taken in a series of points in the feasible design space.

The construction of a *global approximation* relies on the sampling of the design space at N_t locations to obtain response values for the objective functions.

The expression

$$\mathbf{y} = \mathbf{f}(\mathbf{x})$$

represents the true nature of a system model to be optimised. $\mathbf{f}$ is given in terms of design variables $\mathbf{x}$ and responses $\mathbf{y}$. The design space is defined by the upper and lower bounds on the vector of design variables $(x_1, x_2, \ldots, x_{n_{dv}})$.

From the N_t sampled data, approximation models are constructed to describe the response(s) as a function of the n_{dv} design variables.

Mathematically, we have

$$\hat{\mathbf{y}} = \mathbf{g}(\mathbf{x}) + \boldsymbol{\epsilon}$$

where $\hat{\mathbf{y}}$ are the observed response, $\mathbf{g}(\mathbf{x})$ is the approximating function and $\boldsymbol{\epsilon}$ represents error of approximation and/or random error. The approximating function $\mathbf{g}(\mathbf{x})$ can be a low-order polynomial, which can be either linear or quadratic.

The most common approximation approach is to apply the design of experiments (also called response surface method, see Sect. 4.3.1), i.e. employing regression analysis to create a polynomial approximation of the physical objective functions. This provides

– better understanding of the relationship between $\mathbf{x}$ and $\mathbf{y}$,
– facilitated integration of commercial computer codes,
– fast analysis tools for optimisation.

Recent studies have employed the design and analysis of computer experiments (DACE) [84] method for creating approximation models. The DACE modeling methods differ from polynomial approximation because they are able to capture multimodal trends whereas quadratic polynomials are unimodal by definition.

A problem associated with global approximation is that it is difficult to establish a priori the amount and the distribution of data (exact evaluations) that must be used in developing a suitable approximation. A process of trial and error is typically used. Research in order to provide a rational approach to this problem is under development [109].

4.1 Global Approximation Techniques

The procedure for generating a Global approximation model involves three main steps [230]:

– generation of training data,
– selection of the global approximation model,
– fitting of the model to the generated training data.

Table 4.1. Global approximation techniques

Training data generation	Global approximation model	Fitting
(Fractional) factorial	Polynomial	Least squares regression
Central composite	Splines	Maximum likelihood
Orthogonal array	Frequency domain	Back-propagation
Uniformly distributed sequences	Artificial neural networks	...
...	...	

Some options for each step are shown in Table 4.1. The options listed in Table 4.1 will be discussed briefly in the following sections. Uniformly distributed sequences and orthogonal arrays have been already discussed in detail in Sect. 3.4.2.

4.2 Training Data Generation

A computer experiment is represented by a matrix whose n rows denote n simulation runs and the columns denote the particular design variables combination which refers to generated training data. Properly designed computer experiments (i.e. proper generation of training data) are essential for effective GA model description.

4.2.1 (Fractional) Factorial Designs

Important sampling methods useful to generate global approximation models are based on factorial design [218].

Considering a *full factorial design* (i.e. grid, see Sects. 1.2.2, 3.4.2 and [218]) the number of variable vectors are ${n_v}^{n_{dv}}$ for n_{dv} design variables and n_v levels for each design variable (n_v is the number of the design variables within a feasible range, see Sects. 1.2.1 and 2.9). A full factorial design is one in which each computer experiment includes all the possible combinations of design variable type (n_{dv}) and levels (n_v). The reader can find comprehensive information on factorial/fractional sampling in [218].

Fractional factorial sampling [44] can be used when the number of design points (design variable vectors) for a full factorial sample is too large. The reduction of the number of points is done by eliminating points from full factorial design. This obviously can confound (i.e. alias) the effects of more design variables. Aliased effects cannot be estimated unless all of these effects are known independently (Sect. 3.4.2).

Central composite sampling is a two-level factorial design, augmented by n_0 centre points and augmented by a two 'star' points positioned symmetrically with respect to n_0. If we consider a face-centred composite design $2^{n_{dv}} + 2^{n_{dv}} + n_0$ points are sampled.

D-optimal design. A design is said to be D-optimal [42] if

$$\max \det(\mathbf{X}^T \mathbf{X}) \tag{4.1}$$

where $\mathbf{X}$ is a matrix with N rows (N is the number of computer experiments) and n_{dv} columns. We obtain a set of experiments that are nearly orthogonal (see Sect. 3.4.2). The disadvantage of this procedure is that the problem (4.1) is an optimisation problem computationally expensive.

4.2.2 Uniformly Distributed Sequences and Orthogonal Array

These kinds of samples, which have been already dealt with in Sect. 3.4.2, can be used to generate global approximation models.

Global approximation models are often defined by minimising the least squared error (see Sect. 4.4). This implies the estimation of an integral of the error

$$\int (\hat{y}(\mathbf{x}) - y(\mathbf{x}))^2 \mathrm{d}\mathbf{x}$$

by using

$$\frac{1}{N} \sum_{i=1}^{N} (\hat{y}_i(\mathbf{x_i}) - y_i(\mathbf{x_i}))^2$$

The uniformity properties of low discrepancy sequences (see Sect. 3.4.2) are well suited for this kind of estimation and for the setup of a global approximation model by using a limited number of simulation.

4.3 Selection of the Global Approximation Model and Fitting of the Model to the Generated Data

4.3.1 Polynomial Linear and Quadratic Interpolation

The most widely used approximation models are simple low-order polynomials. The first-order (linear) approximation is adequate when little curvature appears to exist. However, when function changes are large and significant curvature exists, the second order (quadratic) approximation is necessary to achieve an adequate accuracy.

Several n-dimensional linear and quadratic approximation methods are available in commercial mathematical software packages (Matlab, ...). The polynomial parameters are usually determined using a lest squares regression analysis to fit the response to existing data.

Given a set of N_t points $(x_{1i}, x_{2i}, \ldots, x_{n_{dv}i})^T$, $i = 1, \ldots, N_t$ in an n_{dv}-dimensional space we compute N_t exact evaluations

$$f(x_{11}, x_{12}, \ldots, x_{1n_{dv}})$$
$$f(x_{21}, x_{22}, \ldots, x_{2n_{dv}})$$
$$\ldots..$$
$$f(x_{N_t1}, x_{N_t1}, \ldots, x_{N_tn_{dv}})$$

The problem is to find the value of each objective function f at a certain point $\mathbf{x}$.

The major limitation of those software packages is that they require equally spaced parameters vectors $(x_{1i}, x_{2i}, \ldots, x_{n_{dv}i})^T$, $i = 1, \ldots, N_t$ in the n_{dv}-dimensional space. Equally spaced point grids (see Sect. 4.2.1) are too large for many applications. A uniform random data set or a uniformly distributed sequence (see Sect. 3.4.2) is fundamental for the definition of base points leading directly to the final result of good approximation (Sect. 3.4.2).

In particular cases (specified above) linear interpolation can be performed on the basis of a set of N_t randomly spaced points defined within an n_{dv}-dimensional space.

Basically, the interpolation is performed by using a weighted sum method on the simplex which contains the evaluation point. Special searches undertaken either to find the best simplex or if the simplex does not exist (extrapolation). Although the solution of the linear interpolation depends on the predefined base points, engineering applications should not suffer from this purely mathematical problem.

The procedure for finding the solution to the problem of the interpolation on n_{dv}-dimensional non-equally spaced grids is as follows:

1. Select n_s base points closest to the evaluation point. A very fast search can be done using hyper-spheres (centred on the evaluation point) with increasing radii.
2. Investigate the n_{dv}-dimensional grid to determine the $n_s + 1$ points which can be used to construct the (best) simplex which contains the evaluation point.
3. If we are able to construct the simplex we compute

$$f = \sum_{i=1}^{n_s+1} \frac{V_i}{V_t} f_i$$

where V_i is the hyper-volume of the simplex containing the evaluation point and obtained by elimination of one of the n_{s+1} points x_i, and V_t the total hyper-volume of the simplex defined by the $n_s + 1$ points.

The quadratic interpolation algorithm differs slightly from the linear one. More information about the objective function(s) are needed; the minimum number of points needed are $1 + n_{dv} + (n_{dv}(n_{dv} + 1))/2$ for the quadratic approach (they are $1 + n_{dv}$ for the linear case). The basic idea is to expand in Taylor's series the objective function(s) as

$$f = f^0 + (\nabla_{\mathbf{x}} f)(\mathbf{x} - \mathbf{x}_0) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^T (H(f))(\mathbf{x} - \mathbf{x}_0)$$

this is equivalent to find a quadric approximation function. We need $n_s \geq n_{dv}$ base points to compute $c_1, \ldots, c_{n_s}$:

$$\begin{cases} c_1(x_{11})^2 + \cdots + c_{n_{dv}}(x_{1n_{dv}})^2 + c_{n_{dv}+1}x_{11}x_{12} + \cdots + c_{n_s-n_{dv}-1}x_{11} + \\ \qquad \cdots + c_{n_s-1}x_{1n_{dv}} + c_{n_s} = f_1 \\ c_1(x_{21})^2 + \cdots + c_{n_{dv}}(x_{2n})^2 + c_{n_{dv}+1}x_{21}x_{12} + \cdots + c_{n_s-n_{dv}-1}x_{21} + \\ \qquad \cdots + c_{n_s-1}x_{2n_{dv}} + c_{n_s} = f_2 \\ \cdots\cdots\cdots \\ c_1(x_{n_s1})^2 + \cdots + c_{n_{dv}}(x_{n_sn_{dv}})^2 + c_{n_{dv}+1}x_{n_s1}x_{n_s2} + \\ \qquad \cdots + c_{n_s-n_{dv}-1}x_{n_s1} + \cdots + c_{n_s-1}x_{n_sn} + c_{n_s} = f_{n_s} \end{cases}$$

4.4 Least Squares Regression Polynomial Approximation

Let us consider a monodimensional problem We can consider a number N_t of exact evaluations $(\mathbf{x}_i, f(\mathbf{x}_i))$ $i = 1, \ldots, N_t$. Let $h_0(\mathbf{x}), \ldots, h_r(\mathbf{x})$ be some (r) guessed functions of $\mathbf{x}$ (basis functions) and $\mathcal{Y}$ the set of the linear combinations of $h_0(\mathbf{x}), \ldots, h_r(\mathbf{x})$

$$\mathcal{Y} = \left\{ \sum_{k=0}^{r} \beta_k h_k(\mathbf{x}) : \beta_k \in \Re \right\}$$

We are looking for an element in $\mathcal{Y}$

$$p_N(\mathbf{x}) = \beta_0 h_0(\mathbf{x}) + \beta_1 h_1(\mathbf{x}) + \ldots + \beta_r h_r(\mathbf{x})$$

that can approximate the data set.

The parameters β_k are determined through least squares regression which minimises the sum of the squares of the deviations of predicted values from the actual values [218].

We need to determine $r + 1$ coefficients β_k which minimise

$$E^2 = \sum_{i=1}^{N_t} (f(\mathbf{x}_i) - \sum_{k=0}^{r} \beta_k h_k(\mathbf{x}_i))^2$$

In matrix form

$$\begin{pmatrix} h_0(\mathbf{x}_1) & h_1(\mathbf{x}_1) & \cdots & h_r(\mathbf{x}_1) \\ \vdots & \vdots & \ddots & \vdots \\ h_0(\mathbf{x}_{N_t}) & h_1(\mathbf{x}_{N_t}) & \cdots & h_r(\mathbf{x}_{N_t}) \end{pmatrix} \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_r \end{pmatrix} - \begin{pmatrix} f(\mathbf{x}_1) \\ \vdots \\ f(\mathbf{x}_{N_t}) \end{pmatrix} = \begin{pmatrix} e_1 \\ \vdots \\ e_{N_t} \end{pmatrix} \quad (4.2)$$

i.e.

$$\mathbf{H}\boldsymbol{\beta} - \mathbf{f} = \mathbf{e}$$

$$E^2 = \mathbf{e}^T \cdot \mathbf{e} = (\mathbf{H}\boldsymbol{\beta} - \mathbf{f})^T(\mathbf{H}\boldsymbol{\beta} - \mathbf{f})$$

The minimum of E^2 occurs when the derivative of E^2 with respect to all $r+1$ parameters β_k is null. We can write this condition as a matrix equation

$$\begin{aligned} &\frac{\partial E^2}{\partial \beta_k} = 0 \quad k = 0, \ldots, r \\ &2(\mathbf{H}^T\mathbf{H}\boldsymbol{\beta} - \mathbf{H}^T\mathbf{f}) = 0 \end{aligned} \tag{4.3}$$

so

$$\mathbf{H}^T\mathbf{H}\boldsymbol{\beta} - \mathbf{H}^T\mathbf{f} = 0$$

or

$$\mathbf{A}\boldsymbol{\beta} = \boldsymbol{\gamma}$$

$$\sum_{k=0}^{r} \alpha_{jk}\beta_k = \gamma_j \quad j = 0, \ldots, r$$

where

$$\alpha_{jk} = \sum_{i=1}^{N_t} h_k(\mathbf{x}_i)h_j(\mathbf{x}_i) \quad \text{equivalent to} \quad \mathbf{A} = \mathbf{H}^T\mathbf{H}$$

$$\gamma_j = \sum_{i=1}^{N_t} f(\mathbf{x}_i)h_j(\mathbf{x}_i) \quad \text{equivalent to} \quad \boldsymbol{\gamma} = \mathbf{H}^T\mathbf{f}$$

The *normal equations*

$$\mathbf{A}\boldsymbol{\beta} = \boldsymbol{\gamma} \quad \text{equivalent} \quad \text{to} \quad \mathbf{H}^T\mathbf{H}\boldsymbol{\beta} = \mathbf{H}^T\mathbf{f}$$

can be solved to find the vector of parameters $\boldsymbol{\beta}$ by LU decomposition [210], which is one of the standard ways to solve a linear system.

An alternative technique involves QR decomposition of the design matrix $\mathbf{H}$. Often $p_N(\mathbf{x})$ is a quadratic polynomial. This type of polynomial model has been used frequently in engineering due to its computational simplicity. However, quadratic polynomial models may be of limited accuracy when the response data to be modeled have multiple local extrema.

Typical polynomial functions are simple and easy to use, but even a full quadratic polynomial requires at least $1 + N_t + N_t(N_t - 1)/2$ exact evaluations of the function to build approximations. This can be a large number if n_{dv} is big.

4.5 Kriging Interpolating Models

In this section only an introduction to Kriging modeling is recalled. A detailed documentation is found in [84, 218]. The Kriging model performs an interpolation (construction of surrogate [104]) not an approximation.

The deterministic response can be modeled as a realisation of a stochastic process $y(\mathbf{x})$

$$y(\mathbf{x}) = \beta_0 + Z(\mathbf{x}) \tag{4.4}$$

where β_0 is a parameter to be estimated. The random process $Z(\mathbf{x})$ is assumed to have zero mean. The covariance matrix of $Z(\mathbf{x})$ is

$$\mathrm{Cov}[Z(\mathbf{x}_i), Z(\mathbf{x}_j)] = \Sigma^2 \mathbf{R}[R(\mathbf{x}_i, \mathbf{x}_j)] \tag{4.5}$$

where $i, j = 1, \ldots, N_t$, Σ^2 is the process variance, $\mathbf{R}$ is the correlation matrix, and R the correlation function selected by the user.

For a smooth response a covariance function with some derivatives might be preferred. Kriging is extremely flexible due to the wide range of correlation functions $R(w, x)$ which may be chosen.

In [84, 218] an exponential correlation function is introduced

$$R(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\theta \sum_{l=1}^{n_{dv}} |x_{li} - x_{lj}|^2\right) \tag{4.6}$$

where θ is the unknown correlation parameter.

The expected value of the mean-squared error between the approximation model $\hat{y}(\mathbf{x})$ and the actual (physical) model $y(\mathbf{x})$ is given by

$$E = \mathrm{E}[(\hat{y}(\mathbf{x}) - y(\mathbf{x}))^2] \tag{4.7}$$

E is a function of θ that can be found using maximum likelihood estimation. Given the data $\mathbf{y}_s = \{y(\mathbf{x}_1), \ldots, y(\mathbf{x}_{N_t})\}^T$ and considering the linear predictor

$$\hat{y}(\mathbf{x}) = \mathbf{P}(\mathbf{x})\mathbf{y}_s \tag{4.8}$$

we have to minimise

$$E = \mathrm{E}[(\mathbf{P}(\mathbf{x})\mathbf{y}_s - \mathbf{y}(\mathbf{x}))^2] \tag{4.9}$$

In several comparative studies, Kriging has been seen to outperform splines and never performs worse than they do.

4.6 Artificial Neural Networks

In this section only some of the basic issues of artificial neural networks are dealt with. The interested reader may refer to [109, 113, 127] for an in-depth documentation on these topics.

Artificial neural networks (ANN) are a wide class of flexible non-linear regression approximation models.

ANNs were inspired by the data-processing characteristics that are typical of the brain. Similarities between biological and artificial systems are at least superficial. The only common feature is the attempt to simulate biological nervous systems by combining many simple computing elements (neurons) into a highly interconnected system.

ANNs rarely have more than a few hundred or a few thousand neurons, while the human brain has about one hundred billion neurons.

ANNs learn from experience. They are data-driven self-adaptive methods in that there are reduced a priori assumptions on the ANN structure.

Many ANNs models are similar or identical to well-known statistical techniques such as generalised linear models, polynomial regression, non-parametric regression [109].

4.6.1 Multi-layer Perceptron Neural Network

The multi-layer perceptron model (feedforward, back-propagating neural network) is the most widely used artificial neural network (ANN) architecture. This model finds application mainly as a function approximation tool; a trained ANN provides a mapping between some input $\mathbf{x} \in \Re^{n_{dv}}$ and output $\mathbf{y} \in \Re^{n_{of}}$. The multi-layer perceptron neural network is able to model a complex non-linear function with an arbitrary degree of accuracy. The multi-layer perceptron neural networks are general-purpose, flexible, non-linear models that can be used when the designer has little knowledge about the form of the relationship between the independent and dependent variables. A multi-layer perceptron neural network consists of a set of processing units (*neurons*) connected to each other over a large number of (weighted) lines (Fig. 4.1). Each neuron receives input from other neurons or external sources and computes an output signal. Neuron can be considered as multiple linear regression models with a non-linear (sigmoidal) transformation function. The complexity of the network can be varied by changing the number of hidden neurons and the number of hidden layers. For this reason multi-layer perceptron neural networks are especially valuable; the designer can easily vary the complexity of the model.

Usually, in order to approximate globally non-linear functions, multi-layer networks have been employed. Each *layer* consists of units which receive input from the layer directly below and send their output to units in layer directly above (without connections skipping one layer).

The first layer is the input layer in which external information is received. The last is an output layer. The input layer and the output layer are separated by one or more intermediate layers called hidden layers. The data processing can extend over multiple layers of neurons but the data flow from input to output is strictly feed-forward without feedback.

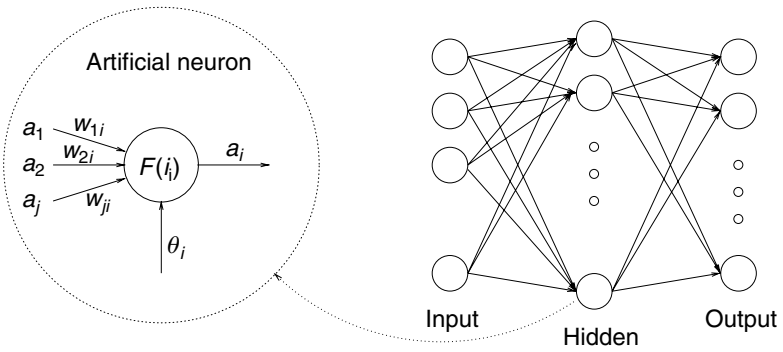


Fig. 4.1. A three-layer multi-layer perceptron neural network (zoom on an artificial neuron)

The total input i_i of the neuron i is given by the weighted sum of the output (a_j) of the units connected plus a bias (offset) term (ϑ)

$$i_i = \sum_j w_{ij}\, a_j + \vartheta_i \tag{4.10}$$

with w_{ij} the weight for the jth input.

The function which gives the effect of the total input i_i on the activation a_i of the neuron is a non-decreasing function of the total input

$$a_i = F(i_i) = F\left(\sum_j w_{ij} a_j + \theta_i\right) \tag{4.11}$$

Generally, a sigmoid function (Fig. 4.2) is introduced

$$F(i_i) = \frac{1}{1 + e^{-i_i\, s}} \tag{4.12}$$

s is the slope of the curve in the origin $i_i = 0$. Alternatively, a hyperbolic tangent function can be used (Fig. 4.2):

$$F(i_i) = \tanh(s i_i) \tag{4.13}$$

The main advantage of these functions is their ability to handle both large and small input signals. The slope (s) is representative of the available gain.

Back-propagation

A multi-layer perceptron neural network has to be configured in order to obtain the desired output by exploiting a set of input data. The network

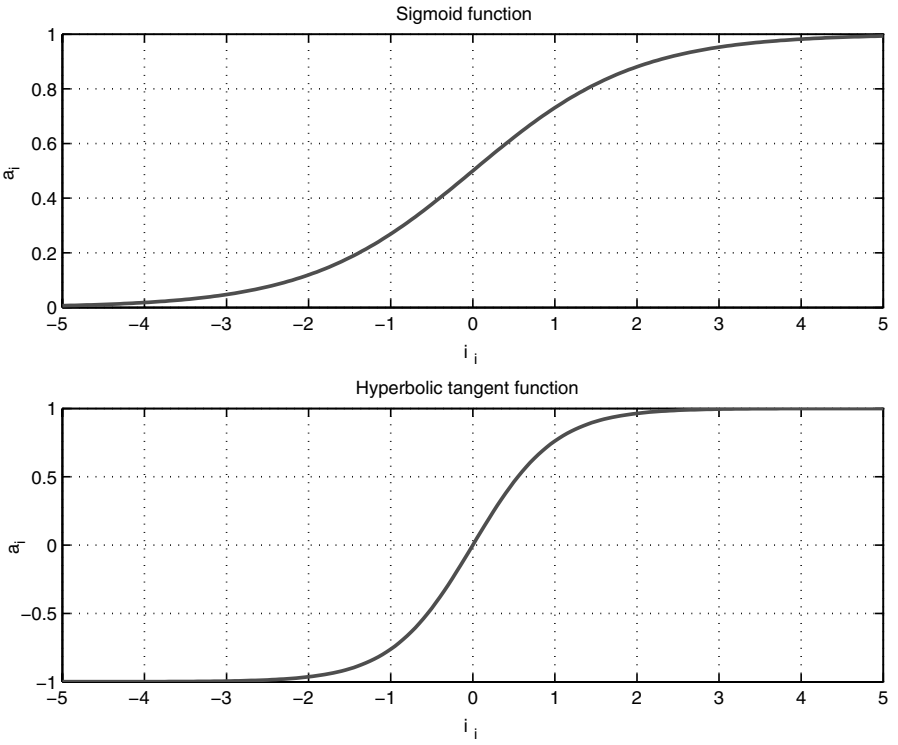


Fig. 4.2. Sigmoid and hyperbolic tangent activation function

is 'trained' by exploiting input data. The training is performed by changing the weights w_{ij} and the offsets θ_i according to some fitting rules. The method employed for multi-layer perceptron neural networks is often called *back-propagation learning rule* [127]. The central idea is that the errors of the neurons of the hidden layers are determined by back-propagating the errors from the output layer.

The sum of squared errors E^2 is given by

$$E^2 = \frac{1}{2}\sum_{i=1}^{N_o}(e_i)^2 \tag{4.14}$$

where e_i is the error at the ith neuron of the output layer and N_o is the number of output neurons. Let y_i be the output at the ith neuron and $\hat{y}_i$ the expected value of this neuron from the input data, the error e_i is determined as

$$e_i = \hat{y}_i - y_i \tag{4.15}$$

The weights and offsets are changed according to

$$\Delta w_{ij} = -\gamma \frac{\partial E^2}{\partial w_{ij}} \quad \Delta \theta_i = -\gamma \frac{\partial E^2}{\partial \theta_i} \tag{4.16}$$

where

$$\frac{\partial E^2}{\partial w_{ij}} = \frac{\partial E^2}{\partial i_i} \frac{\partial i_i}{\partial w_{ij}} \quad \frac{\partial E^2}{\partial \theta_i} = \frac{\partial E^2}{\partial i_i} \frac{\partial i_i}{\partial \theta_i} \tag{4.17}$$

$$\frac{\partial i_i}{\partial w_{ij}} = a_j \quad \frac{\partial i_i}{\partial \theta_i} = 1 \tag{4.18}$$

Defining

$$\delta_i = -\frac{\partial E^2}{\partial i_i} \tag{4.19}$$

The training rule is based on the steepest descent along the gradient of the error surface

$$\Delta w_{ij} = \gamma \, \delta_i \, a_j \quad \Delta \theta_i = \gamma \, \delta_i \tag{4.20}$$

The iterative procedure for evaluating the δ_i is called back-propagation algorithm. The partial derivatives (4.19) can be rewritten as

$$\delta_i = -\frac{\partial E^2}{\partial i_i} = -\frac{\partial E^2}{\partial a_i} \frac{\partial a_i}{\partial i_i} \tag{4.21}$$

Using (4.11)

$$\frac{\partial a_i}{\partial i_i} = F'(i_i) \tag{4.22}$$

$F'(i_i)$ is the derivative of the activation function of the ith neuron with respect to the input i_i. In order to evaluate the first term of the expression (4.21), we consider two different situations.

Assuming that the ith neuron is an output unit

$$\frac{\partial E^2}{\partial a_i} = -(\hat{y}_i - y_i) \tag{4.23}$$

Substituting in (4.21) and considering (4.22) we can write

$$\delta_i = (\hat{y}_i - y_i) F_i'(i_i) \tag{4.24}$$

Assuming that the ith neuron belongs to a hidden layer

$$\frac{\partial E^2}{\partial a_i} = \sum_{h=1}^{N_o} \frac{\partial E^2}{\partial i_h} \frac{\partial i_h}{\partial a_i} = \sum_{h=1}^{N_o} \frac{\partial E^2}{\partial i_h} \frac{\partial}{\partial a_i} \sum_{k=1}^{N_h} w_{hk} a_k = \sum_{h=1}^{N_o} \frac{\partial E^2}{\partial i_h} w_{hi} = -\sum_{h=1}^{N_o} \delta_h w_{hi} \tag{4.25}$$

where N_h is the number of neurons connected to the neuron h and substituting in (4.21) and considering (4.22) we obtain

$$\delta_i = F'(i_i) \sum_{h=1}^{N_o} \delta_h w_{hi} \tag{4.26}$$

The expressions (4.24) and (4.26) give an iterative procedure to evaluate δ for every unit into the network.

The back-propagation training method is quite slow, requiring the user to set various algorithmic parameters (i.e. γ) by trial and error. Fortunately, the training can be easily completed using the general purpose non-linear optimisation algorithms described in the previous chapters.

A trained multi-layer perceptron neural network can be used in lieu of the physical model of the system for optimisation purposes (global approximation).

Any well-trained ANN has the ability to approximate (and even to extrapolate) the output of the original model. The training process has to be stopped before the generalization error becomes too important (overtraining) [109].

A cross-validation procedure can be introduced in order to reduce this problem. The set of exact evaluations is split into two subsets, a training set (85–90%) and a validation set (10–15%). Only the bigger set (training set) is used for training. The training process is completed when the validation error (error computed on the validation set) stops decreasing.

Design of a Multi-layer Perceptron Neural Network

Design of a multi-layer perceptron neural network is an important yet difficult task. The number of hidden layers and the number of neurons in each hidden layer are determined by the output data of the model that the ANN is supposed to represent. Unfortunately, no general rule exists to guide the choice a priori.

Theoretical work in ANN has shown that a single hidden layer is sufficient to approximate any complex non-linear function to any desired degree of accuracy as long as enough hidden nodes are used [36].

It is important to notice that ANN representations are not unique, and there may be several different structures that can closely represent the actual output of the original model.

Although a network with one hidden layer is a universal approximator, there exist various applications in which more than one hidden layer can be useful. Sometimes a highly non-linear function can be approximated with fewer weights when multiple hidden layers are used [258].

Experimentation with a pilot sample is often used to find the appropriate numbers of hidden neurons. If the multi-layer perceptron network architecture is large enough, a multi-layer perceptron neural network can be a nearly universal approximator. Approximation results (see, e.g. [12, 36, 118]) show that virtually any real function of interest in $\Re^k$ can be appropriately approximated by one-hidden-layer sigmoidal neural networks.

Recently, techniques have been developed to determine optimal[1] network structures for a given problem. Network growing (adding hidden neurons starting with a small ANN) and pruning (deleting nodes starting with an oversized ANN) are the actions that are used for obtaining the optimal network.

4.6.2 Radial Basis Function Neural Network

An obvious disadvantage of multi-layer perceptron neural networks (MLPNN) is that they are genuine non-linear in the parameters (see Sect. 4.6.1). Fitting must be based on non-linear optimisation techniques and the parameters estimate (weights, offsets) may be trapped at a local minimum during the fitting procedure.

An efficient alternative to MLPNN is the radial basis function neural network (RBFNN) [27]. The original RBFNN method requires that there must be as many RBFNN centres as the training data points.

The construction of an RBFNN involves three layers with different roles [20, 26, 209]. The input layer is made up of source nodes that connect the network to its environment. The second layer, the only hidden layer in the network, applies a non-linear transformation from the input space to the hidden space (see Fig. 4.3); the hidden space in actual applications is often of high dimensionality. The output layer is linear.

In the case of a scalar output the model can be expressed by

$$\hat{f}(\mathbf{x}, \mathbf{w}) = \sum_{i=1}^{r} h_i(\mathbf{x}) w_i$$

$$h_i(\mathbf{x}) = \phi\left(\sum_{j=1}^{n_{dv}} (x_j - c_{ij})^2 / (\sigma_{ij})^2\right) \tag{4.27}$$

where ϕ is the *basis function*, every hidden unit transfer function $h_i(\mathbf{x}), i = 1, \ldots, r$ depends only on the scaled distance between the input $\mathbf{x}$ and a centre $\mathbf{c}_i$ scaled by a spread parameter $\boldsymbol{\sigma}_i$ [26] (the meaning of centre and spread will be explained in the following). These parameters ($\mathbf{c}_i$ and $\boldsymbol{\sigma}_i$) are to be tuned for every h_i,

The most widely used basis functions h_i (Fig. 4.4) are respectively the *Gaussian basis functions*:

$$h_i(\mathbf{x}) = \exp\left(-\sum_{j=1}^{n_{dv}} (x_j - c_{ij})^2 / (\sigma_{ji})^2\right) \tag{4.28}$$

and *thin plate spline function* [136]:

[1] An ANN structure is optimal if it minimises the number of hidden layers and the number of neurons without compromising the representation of the output of the original (physical) model

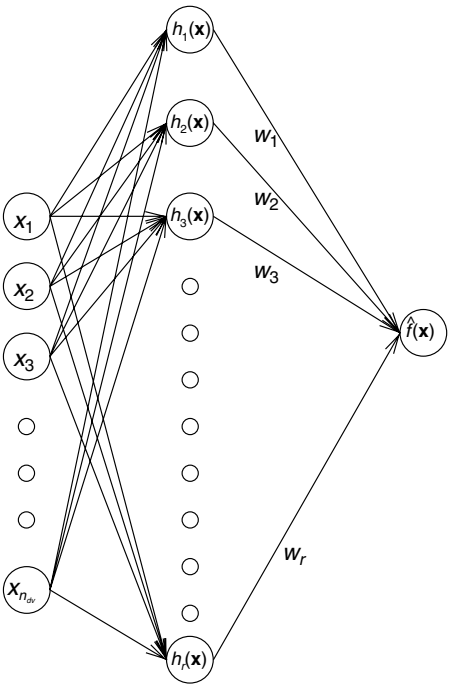


Fig. 4.3. The traditional radial basis function neural network for multiple inputs and one output

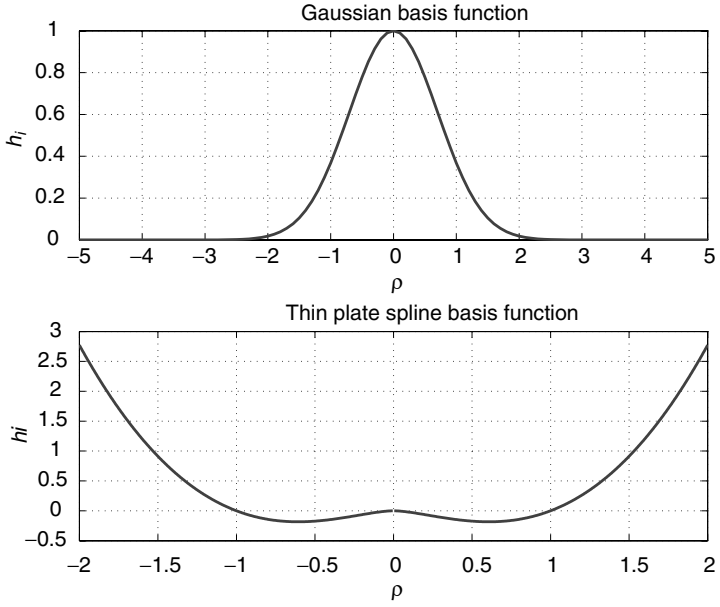


Fig. 4.4. The most frequently used basis functions for RBFNN.

$$\rho = \left(\sum_{j=1}^{n_{dv}} (x_j - c_{ij})^2 / (\sigma_{ji})^2 \right)^{1/2} \qquad h_i(\mathbf{x}) = \rho^2 \log(\rho) \tag{4.29}$$

As pointed out in [20], the choice of centres which determines the number of free parameters is of crucial importance. Too few centres in the network may not be capable of generating a good approximation to the target function; on the other hand, too many centres may fit misleading variations due to noise in the data. This is the bias variance dilemma typical of this type of problems [77].

In [172] unsupervised clustering of the training data were used to generate the centres, but no control was provided over the output data.

A better direct approach is selecting a subset of centres from a larger set which would overfit the data. This is the approach adopted in [26], where centres are added in an RBFNN one at a time until a sufficient approximation is obtained. In this method the training data are used as candidate centres. An approach is to reduce *complexity* of the global approximation model $\hat{f}$ (in order to avoid overfitting problems) by adding a *complexity penalty* term to the mean-squared error to be minimised:

$$\min \frac{1}{N_t} \left(\sum_{i=1}^{N_t} \left(y_i - \hat{f}(\mathbf{x}_i, \mathbf{w}) \right)^2 + \lambda_r \mathbf{w}^T \mathbf{w} \right) \tag{4.30}$$

where λ_r is known as regularisation parameter, r the number of hidden units and $\mathbf{w} = \{w_1, \ldots, w_r\}^T$ are the weights. This function is called *cost function* and the extra penalty term discourages large weights and prevents overfitting [83,248]. Equation (4.30) is quadratic in the weight vector and it has a unique minimum[2] at

$$\mathbf{w} = (\mathbf{H}^T \mathbf{H} + \lambda_r \mathbf{I}_r)^{-1} \mathbf{H}^T \mathbf{y} \tag{4.32}$$

where $\mathbf{I}_r$ is an identity $r \times r$ matrix and $\mathbf{H}$ is the design matrix defined in Eq. (4.2). We can obtain the sum of squared errors at the optimal weights

[2] By using the notation used in Eq. (4.2), the approximating function can be written in matrix form

$$\hat{\mathbf{f}}(\mathbf{x}_i, \mathbf{w}) = \mathbf{H}^T \mathbf{w}$$

so Eq. (4.30) becomes

$$\hat{E} = \frac{1}{N_t} [(\mathbf{y} - \mathbf{H}\mathbf{w})^T (\mathbf{y} - \mathbf{H}\mathbf{w}) + \lambda_r \mathbf{w}^T \mathbf{w}]$$

by setting

$$\frac{\partial \hat{E}}{\partial w_i} = 0$$

we have (see Eq. (4.3))

$$(\mathbf{H}^T \mathbf{H} + \lambda_r \mathbf{I}_r) \mathbf{w} = \mathbf{H}^T \mathbf{y} \tag{4.31}$$

$$\hat{E} = \frac{1}{N_t}\mathbf{y}^T\mathbf{P}^2\mathbf{y} \tag{4.33}$$

where

$$\mathbf{P} = \mathbf{I}_p - \mathbf{H}(\mathbf{H}^T\mathbf{H} + \lambda_r\mathbf{I}_r)^{-1}\mathbf{H}^T \tag{4.34}$$

$\mathbf{P}$ is called *projection matrix* [212]. The *effective number of free parameters*, playing a crucial role in determining model complexity, is given by [172]

$$\gamma = n_{dv} - \text{trace}(\mathbf{P}) = r - \lambda_r \,\text{trace}((\mathbf{H}^T\mathbf{H} + \lambda_r\mathbf{I}_r)^{-1}) \tag{4.35}$$

The parameter λ_r can be chosen according to different criteria. The most common is the minimisation of generalised cross-validation (GCV) error estimate [98]:

$$\epsilon^2_{GCV} = \frac{n^2_{dv}}{(n_{dv} - \gamma)^2}\frac{\mathbf{y}^T\mathbf{P}^2\mathbf{y}}{n_{dv}} \tag{4.36}$$

RBFNN tend to require large train data set, especially in multidimensional spaces. The RBFNN can be viewed as a non-linear regression model, the weights can be estimated by any non-linear least squares method, although this would yield a vastly over–paremetrised model if every observation was used as an RBF centre. For this reason the generated training data can be clustered.

The method used in the applications reported in Part II of this book [128, 129, 189], determines the centre positions and the sizes of the radial basis functions (RBFs) $h_i(\mathbf{x})$ by employing the hyper-rectangular subdivisions imposed by a regression tree [19, 72]. The first stage is to generate a decision tree (Fig. 4.5). The root node is the smallest hyper-rectangle that includes all of the generated training data. Its size (half-width) $\mathbf{s}_j$ and centre $\mathbf{c}_j$ in each dimension j are

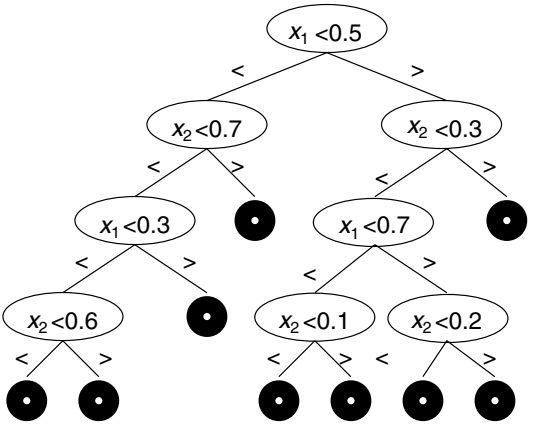


Fig. 4.5. An example of decision tree

$$\begin{aligned} \mathbf{s}_j &= \tfrac{1}{2}\left(\max_{i\in\mathcal{S}}(x_{ij}) - \min_{i\in\mathcal{S}}(x_{ij})\right) \\ \mathbf{c}_j &= \tfrac{1}{2}\left(\max_{i\in\mathcal{S}}(x_{ij}) + \min_{i\in\mathcal{S}}(x_{ij})\right) \end{aligned} \tag{4.37}$$

where $\mathcal{S} = \{1, 2, \ldots, N_t\}$ is the set of training indices. A bifurcation of the root node divides the training samples into left and right subsets, $\mathcal{S}_L$ and $\mathcal{S}_R$, with a boundary in one of the dimension j such that

$$\begin{aligned} \mathcal{S}_L &= \{i : x_{ij} \leq \zeta\} \\ \mathcal{S}_R &= \{i : x_{ij} > \zeta\} \end{aligned} \tag{4.38}$$

where $0 < \zeta < 1$, $\mathbf{x} \in [0,1]^{n_{dv}}$ ($\mathbf{x}_i$ is normalised, see Sect. 3.4.2). The mean output of both sides of the bifurcation is

$$\begin{aligned} \bar{y}_L &= \tfrac{1}{n_L}\textstyle\sum_{i\in\mathcal{S}_L} y_i \\ \bar{y}_R &= \tfrac{1}{n_R}\textstyle\sum_{i\in\mathcal{S}_R} y_i \end{aligned} \tag{4.39}$$

where n_L and n_R are the number of samples in each subset. The residual square error between global approximation model and training data is then

$$E(\zeta) = \frac{1}{N_t}\left(\sum_{i\in\mathcal{S}_L}(y_i - \bar{y}_L)^2 + \sum_{i\in\mathcal{S}_R}(y_i - \bar{y}_R)^2\right) \tag{4.40}$$

The bifurcation which minimises $E(\zeta)$ over all possible choices of ζ is used to create the children of the root node and is found by a simple discrete search over n_{dv} dimensions and N_t cases. The children of the root node is then split recursively in the same manner. The process terminates when all remaining bifurcations would create children containing fewer than N_{min} samples, where N_{min} is a parameter to be defined. The regression tree is used to generate potential RBFNN centres and spreads. Each hyper-rectangle of the regression

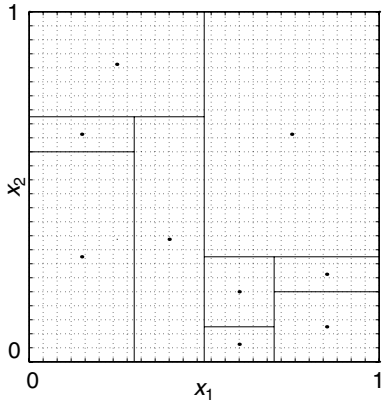


Fig. 4.6. The decomposed space in $[0,1]^2$ obtained by the decision tree reported in Fig. 4.5

tree defines a RBF centre $\mathbf{c}_j$ which is the hyper-rectangle centre itself (see Fig. 4.6). The spread is defined by the size $\mathbf{s}_j$ of the hyper-rectangle, scaled by a parameter α_{RBF} so that $\sigma_{ij} = \alpha_{RBF} s_{ij}$. This parameter α_{RBF} must be optimised during the network training procedure using the GCV prediction error. In the network there is no bias, since the root node of the regression tree acts as a bias.

Each collection of RBFNNs (one for each combination of trial values of α_{RBF}) forms a set of candidate RBFNNs from which a subset is selected to create a network. The network with the lowest prediction error (4.36) is the preferred global approximation model. RBFNN centres and spreads generated by the regression tree are selected and added to the network iteratively. At each step the candidate which most decreases the sum of squared errors and has not been selected before, is chosen to be added to the network. Initially, the predicted error decreases but as the network becomes larger its complexity increases and begins to overfit data. When the GCV error (4.36) starts

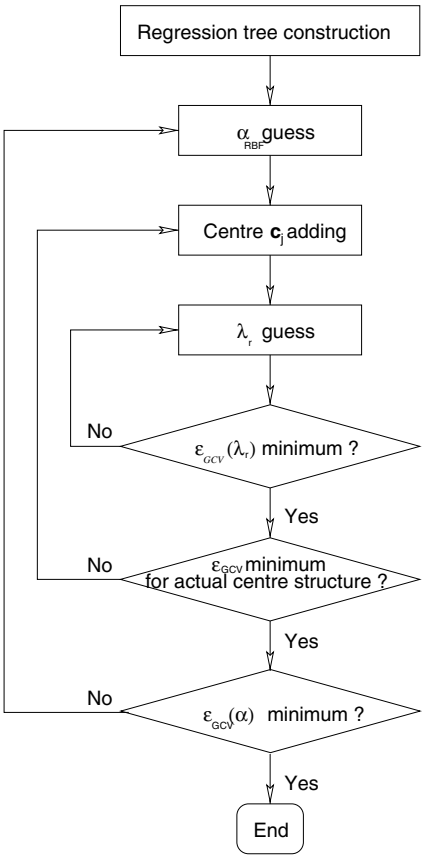


Fig. 4.7. Flow chart of RBFNN parameter definition using regression trees

to increase, the algorithm stops to add centres. As an additional safeguard against overfitting, the method uses the cost function (4.30) and the regularisation parameter λ_r is re-estimated between each iteration. Figure 4.7 shows the RBFNN parameters determination procedure. Efficient computation is based on Graham–Smith orthogonalisation [26] that allows fast and efficient computation of inverse matrix.

Part II

Applications

5

Optimal Ride Comfort and Active Safety of Road Vehicles

A basic study on the optimisation of the ride comfort and active safety of road vehicles is presented[1]. Both passively and actively suspended vehicles are considered. The optimisation of vehicle design variables is performed on the basis of multi-objective programming (MOP) (see Sect. 2.10). Simple analytical formulae for the estimation of ride comfort and active safety of vehicles are derived; these analytical formulae are used in conjunction with MOP to find symbolically (whenever possible) the optimal suspension design variables ensuring the best compromise among discomfort (i.e. the standard deviation of the vehicle body vertical acceleration), road holding (i.e. the standard deviation of the vertical force applied between tyre and road) and body–wheel working space (i.e. the standard deviation of the relative displacement between wheel and vehicle body).

In the literature a number of papers exist dealing with the problem of deriving simple analytical formulae for the estimation of the dynamic response of road vehicles subject to random excitations generated by road irregularity [25,120,137,140,176,247]. In each of the cited papers the derived formulae refer to a very simple power spectral density of the road irregularity (of the form $1/\omega^2$ [47]). Here a more complex power spectral density (of the form $1/(\omega^2+\omega_c^2)$) is used which estimates more accurately the amplitudes of the road irregularity at low excitation frequencies. The analyses made in literature [120, 167, 227] were generally based on purely numerical computations (in contrast with analytical ones), even when dealing with very simple vehicle system models. The results were correct, but authors do think that could be useful to present analytical solutions whenever possible.

With reference to wheel and body vibrations, a number of authors have dealt with the problem of deriving basic concepts useful for road vehicle suspension tuning [25, 41, 46, 78, 82, 90, 91, 137, 141, 167, 173, 227, 247]. Many important relationships have been highlighted among vehicle suspension design

[1] The complete and exhaustive documentation on the topics that are dealt with in this chapter can be found in [90, 91]

variables and suspension objective functions (i.e. performance indices). However, it seems that an ultimate general theory was not derived. This was mainly due to the fact that a theoretical definition of what is the best compromise among conflicting objective functions was not explicitly introduced and properly exploited. Another limitation was that authors resorted often to numerical simulations even when dealing with simple models. Referring to the tuning of gains of actively suspended vehicles, analytical formulae seem to have been derived only for the simple 1 d.o.f. system model (results reported in [16, 124]), establishing the well-known 'sky-hook' control strategy.

In this chapter a comprehensive general theory is exposed [90,91] by deriving analytical formulae defining the optimal relationship among vehicle suspension design variables and suspension objective functions. The derivation of these analytical formulae has been made possible by exploiting multi-objective programming (MOP) (Sect. 2.10).

In the first section, analytical formulae describing the motions of passively suspended road vehicles on rough road are derived. These formulae are used in the second section for deriving, by means of multi-objective programming, the optimal tuning of the vehicle suspension design variables.

In the third section the dynamic behaviour of actively suspended road vehicles is presented. Analytical formulae are derived for the description of the dynamical behaviour of actively suspended vehicles running on rough road.

These formulae are used in the fourth section where an attempt is made to define (in analytical form) the fundamental optimal relationships among vehicle suspension design variables and vehicle objective functions.

5.1 System Model of a Passively Suspended Vehicle

5.1.1 Equations of Motion and Response to Stochastic Excitation

The quarter car system model of a passively suspended road vehicle is shown in Fig. 5.1. The mass m_1 represents approximately the mass of the wheel plus part of the mass of the suspension arms, m_2 represents approximately one fourth of the body mass (m_2 could be computed more precisely taking into account the position of the centre of gravity along the wheelbase, as shown in [227,247]). k_2 represents the suspension stiffness, r_2 the suspension damping and k_1 the tyre radial stiffness. The excitation comes from the road irregularity ξ. The model is generally reputed sufficiently accurate for capturing the essential features related to discomfort, road holding and working space (see [227]). The linear equations of motions pertaining to the system model are

$$\begin{aligned} m_1\ddot{z}_1 - r_2\,(\dot{z}_2 - \dot{z}_1) - k_2\,(z_2 - z_1) + k_1\,(z_1 - \xi) = 0 \\ m_2\ddot{z}_2 + r_2\,(\dot{z}_2 - \dot{z}_1) + k_2\,(z_2 - z_1) = 0 \end{aligned} \tag{5.1}$$

The responses of the vehicle model are respectively the vertical vehicle body acceleration ($\ddot{z}_2$), the force applied between road and wheel (F_{z1}), the relative displacement between wheel and vehicle body ($z_2 - z_1$).

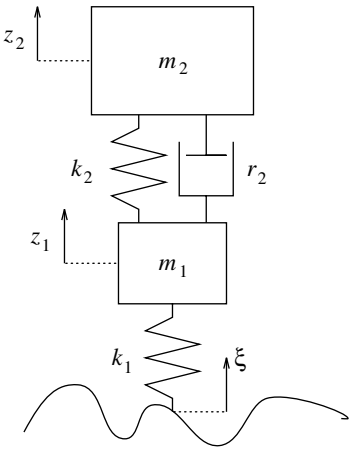


Fig. 5.1. Quarter car vehicle model

The discomfort is evaluated by computing the standard deviation of the vertical vehicle body acceleration ($\sigma_{\ddot{z}_2}$). The higher the standard deviation, the higher is the *discomfort*. This approach seems to be providing very good correlation with subjective ride comfort ratings [120, 231].

The standard deviation of the tyre radial force ($\sigma_{F_{z_1}}$) is related to *road holding*. The variation of tyre radial force can lead to a loss of contact with the ground and poor handling ability [120].

The standard deviation of the relative displacement between wheel and vehicle body ($\sigma_{z_2-z_1}$), i.e. the *working space*, is related to design and packaging constraints, as well as to wheel lateral vibrations. Road holding and working space are strictly related to *active safety*.

Discomfort ($\sigma_{\ddot{z}_2}$), road holding (σ_{F_z}) and working space ($\sigma_{z_2-z_1}$) are the *objective functions* to which reference will be made in the chapter.

The transfer function [88] between the displacement ξ and z_1 is given by

$$Z_1(j\omega) = \frac{k_1(k_2 + jr_2\omega - m_2\omega^2)}{D(j\omega)} \tag{5.2}$$

$$D(j\omega) = k_1k_2 + jk_1r_2\omega - (k_2m_1 + k_1m_2 + k_2m_2)\omega^2 + \\ -jr_2(m_1 + m_2)\omega^3 + m_1m_2\omega^4$$

The transfer function between the imposed displacement ξ and z_2 reads

$$Z_2(j\omega) = \frac{k_1(k_2 + jr_2\omega)}{D(j\omega)} \tag{5.3}$$

The transfer function between ξ and $\ddot{z}_2$ is

$$H_1(j\omega) = -\omega^2 Z_2(j\omega) \tag{5.4}$$

The transfer function between ξ and F_{z1} is

$$H_2(j\omega) = k_1\big(1 - Z_1(j\omega)\big) \tag{5.5}$$

The transfer function between ξ and $z_2 - z_1$ is

$$H_3(j\omega) = Z_2(j\omega) - Z_1(j\omega) \tag{5.6}$$

The displacement ξ (road irregularity) may be represented by a random variable defined by a stationary and ergodic stochastic process with zero mean value [47,123]. The power spectral density (PSD) of the process may be determined on the basis of experimental measurements and in the literature there are many different formulations for it (e.g. see [47] and [167]).

In the present chapter two PSDs have been considered

$$P_{SD\xi\,1}(\omega) = \frac{A_b\, v}{\omega^2} \tag{5.7}$$

$$P_{SD\xi\,2}(\omega) = \frac{A_v\, \omega_c}{{\omega_c}^2 + \omega^2} \tag{5.8}$$

where $\omega_c = a\, v$. The value of the coefficient a (rad/m) depends on the shape of the road irregularity spectrum.

In a log–log scaled plot (abscissa ω), the spectrum of Eq. (5.7) takes the shape of a line sloped at rate -2. The simple expression (5.7) approximates various roads with different degrees of accuracy. It generally overestimates the amplitudes of the irregularity at low frequency. In the following, Eq. (5.7) will be indicated as *one-slope* power spectral density (1S-PSD).

A better correlation with measured spectra can be obtained by resorting to more complex spectra as suggested in [47]. Equation (5.8) has been reported in [167, 250]. In a log–log scaled plot (abscissa ω) Eq. (5.8) takes the shape of a *two-slope curve*, thus reference will be made by the acronym 2S-PSD. Reference values for the parameters A_b and A_v are given in [90].

The PSD, P_{SDl}, of the output of an asymptotically stable system can be computed as (see, e.g. [176])

$$P_{SDl}(\omega) = |H_l(j\omega)|^2\, P_{SD\xi\,q}(\omega) \quad l = 1, \ldots, 3 \quad q = 1, 2 \tag{5.9}$$

for $l = 1$, P_{SDl} represents the PSD of the vertical acceleration of the vehicle body, for $l = 2$, P_{SDl} represents the PSD of the vertical force applied between tyre and road, and for $l = 3$, P_{SDl} represents the PSD of the relative displacement between wheel and vehicle body, the index $q = 1$ refers to the 1S-PSD and the index $q = 2$ refers to the 2S-PSD.

5.1.2 Derivation of Standard Deviations in Analytical Form

By definition (see [176]) the variance of a random variable described by a stationary and ergodic stochastic process is

$$\sigma_l^2 = \frac{1}{2\pi}\int_{-\infty}^{+\infty} P_{SDl}(\omega)d\omega \tag{5.10}$$

In [180] it is shown that an analytical solution exists for σ_l^2 if P_{SDl} can be written as

$$P_{SDl} = \frac{N_{k-1}(j\omega)\, N_{k-1}(-j\omega)}{D_k(j\omega)\, D_k(-j\omega)} \tag{5.11}$$

where D_k is a polynomial of degree k, and N_{k-1} is a polynomial of degree $k-1$ ($k \geq 1$). By inspection of Eqs. (5.4)–(5.9) one may understand that P_{SDl} can be written as in Eq. (5.11); in fact, for example, considering the vertical acceleration of the vehicle body ($\ddot{z}_2$), and being from Eq. (5.9)

$$P_{SD\xi\, q}(\omega) = |H_1(j\omega)|^2\, P_{SD\xi\, q}(j\omega) \quad q = 1, 2$$

and being

$$|H_1(j\omega)|^2 = H_1(j\omega)H_1(-j\omega)$$

and, for $q = 1$

$$P_{SD\xi\, 1}(\omega) = \frac{A_b\, v}{\omega^2} \quad \rightarrow \quad P_{SD\xi\, 1}(j\omega) = A_b\, v \frac{1}{j\omega}\, \frac{1}{-j\omega}$$

and, for $q = 2$

$$P_{SD\xi\, 2}(\omega) = \frac{A_v\, \omega_c}{{\omega_c}^2 + \omega^2} \quad \rightarrow \quad P_{SD\xi\, 2}(j\omega) = A_v\, \omega_c \frac{1}{\omega_c + j\omega}\, \frac{1}{\omega_c - j\omega}$$

the following expressions for $N_{k-1}(j\omega)/D_k(j\omega)$ can be obtained for $q = 1$ (1S-PSD), $k = 4$ (see Eqs. (5.3), (5.4))

$$\frac{N_2(j\omega)}{D_4(j\omega)} = (A_b v)^{\frac{1}{2}}(k_1 j)\omega(k_2 + r_2 j\omega)/[k_1k_2 + jk_1r_2\omega - (k_2m_1 + k_1m_2 + k_2m_2)\omega^2 + (-jr_2)(m_1 + m_2)\omega^3 + m_1m_2\omega^4] \tag{5.12}$$

for $q = 2$, (2S-PSD), $k = 5$ (see Eqs. (5.3) and (5.4))

$$\frac{N_3(j\omega)}{D_5(j\omega)} = (A_v\, \omega_c)^{\frac{1}{2}}(-k_1\omega^2(k_2 + r_2 j\omega))/[k_1k_2 + jk_1r_2\omega - (k_2m_1 + k_1m_2 + k_2m_2)\omega^2 + (-jr_2)(m_1 + m_2)\omega^3 + m_1m_2\omega^4]/(j\omega + \omega_c) \tag{5.13}$$

It is important to notice that P_{SDl} can be written as in Eq. (5.11) if $P_{SD\xi\, q}(j\omega)$ can be expressed as

$$P_{SD\xi\, q}(j\omega) = \frac{N_\xi(j\omega)\, N_\xi(-j\omega)}{D_\xi(j\omega)\, D_\xi(-j\omega)}$$

This occurrence has been firstly exploited in [247], with reference to the simple 1S-PSD ($q = 1$). The more refined 2S-PSD ($q = 2$) has been used in this chapter [90].

The analytical formulae presented in the following subsections have been derived by means of the analytical solutions of the integral (5.10) introduced in [180] and reported in Appendix 5.6.

Formulae Referring to the 1S-PSD (Eq. (5.7))

The analytical formulae giving the discomfort, road holding and working space are reported. They have been obtained by solving analytically Eq. (5.10) (1S-PSD).

– Variance of the vehicle body acceleration $\ddot{z}_2$ (square of $\sigma_{\ddot{z}_2}$)

$$\sigma^2_{\ddot{z}_2} = 1/2\, A_b v\, \overline{\sigma}^2_{\ddot{z}_2} \tag{5.14}$$

$$\overline{\sigma}^2_{\ddot{z}_2} = \frac{1}{m_2^2}\left(\frac{(m_2+m_1)k_2^2}{r_2} + k_1 r_2\right) \tag{5.15}$$

– Variance of the force acting between road and wheel F_z (square of σ_{F_z})

$$\sigma^2_{F_z} = 1/2\, A_b\; v\, \overline{\sigma}^2_{F_z} \tag{5.16}$$

$$\overline{\sigma}^2_{F_z} = (m_2+m_1)^2\,(P) \tag{5.17}$$

$$P = \left(\frac{(m_2+m_1)k_2^2}{m_2^2 r_2} - \frac{2k_1k_2m_1}{m_2 r_2(m_2+m_1)} + \frac{k_1^2 m_1}{r_2(m_2+m_1)^2} + \frac{k_1 r_2}{m_2^2}\right)$$

– Variance of the relative displacement between wheel and vehicle body $z_2 - z_1$ (square of $\sigma_{z_2-z_1}$)

$$\sigma^2_{z_2-z_1} = 1/2\, A_b\; v\, \overline{\sigma}^2_{z_2-z_1} \tag{5.18}$$

$$\overline{\sigma}^2_{z_2-z_1} = \frac{m_2+m_1}{r_2} \tag{5.19}$$

These formulae were already derived and presented in [25, 88, 137, 247].

Figure 5.2 shows respectively the discomfort ($\sigma_{\ddot{z}_2}$), road holding (σ_{Fz}) and working space ($\sigma_{z_2-z_1}$) as function of the vehicle speed (v) considering the reference vehicle.

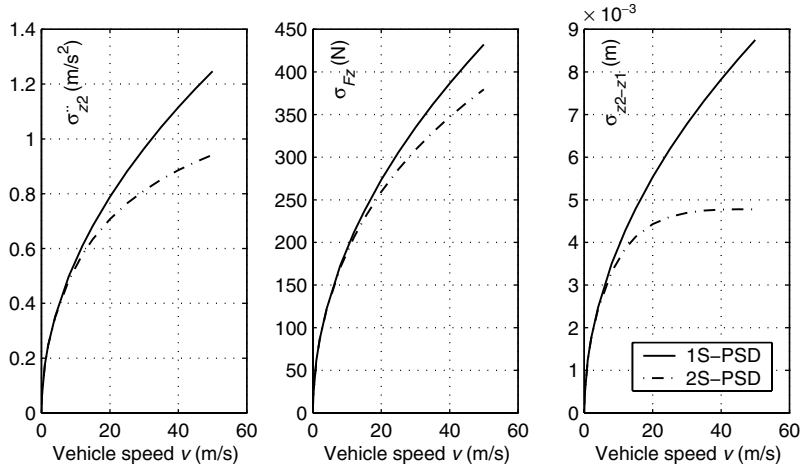


Fig. 5.2. Discomfort ($\sigma_{\ddot{z}_2}$), road holding (σ_{Fz}), working space ($\sigma_{z_2-z_1}$) as function of the vehicle speed. Data of the reference vehicle in Table 5.1, running condition data in Table 5.2

Table 5.1. Data of the reference road vehicle taken into consideration

Design variable	Reference value	Lower and upper bound[a]
m_{2r} (kg)	229	114–458
m_{1r} (kg)	31	15–62
k_{1r} (N/m)	120,000	60,000–240,000
k_{2r} (N/m)	20,000	10,000–40,000
r_{2r} (Ns/m)	1,200	600–2,400

[a]Lower and upper bounds refer to design variables sensitivity analysis

Table 5.2. Data of the road roughness taken into consideration

Parameter		Reference value
A_b	(m)	1.4E−5
$a = \omega_c/v$	(rad/m)	0.4
A_v	(m^2)	3.5E−5

Formulae Referring to the 2S-PSD (Eq. (5.8))

The analytical formulae giving the discomfort, road holding and working space are reported. They have been obtained by solving analytically Eq. (5.10) (2S-PSD).

– Variance of the vehicle body acceleration $\ddot{z}_2$ (square of $\sigma_{\ddot{z}_2}$)

$$\sigma^2_{\ddot{z}_2} = c_{rv} \frac{k_1^2 r_2^2 (k_2 + r_2\omega_c + m_2\omega_c^2) + k_1 k_2^2 ((m_1 + m_2)(k_2 + r_2\omega_c) + m_1 m_2 \omega_c^2)}{(m_2^2 r_2(D_S))} \tag{5.20}$$

where

$$D_S = k_1 k_2 + k_1 r_2 \omega_c + k_2(m_1 + m_2)\omega_c^2 + k_1 m_2 \omega_c^2 + (m_1 + m_2) r_2 \omega_c^3 + m_1 m_2 \omega_c^4$$

$$\omega_c = a\, v$$

$$c_{rv} = 1/2\, A_v\, a\, v$$

– Variance of the force acting between road and wheel F_z (square of σ_{F_z})

$$\sigma^2_{F_z} = c_{rv} \frac{A_{2S} + B_{2S} + C_{2S}}{k_1^2 m_2^2 r_2^2 (D_S)} \tag{5.21}$$

where

$$A_{2S} = k_1^4 (m_1 + m_2) r_2 (-2k_2 m_1 m_2 + m_1 r_2^2 + m_2 r_2^2)(k_2 + r_2\omega_c + m_2\omega_c^2)$$
$$B_{2S} = k_1^3 k_2^2 (m_1 + m_2)^2 r_2 (k_2 m_1 + k_2 m_2 + m_1 r_2 \omega_c + m_2 r_2 \omega_c + m_1 m_2 \omega_c^2)$$
$$C_{2S} = k_1^4 m_1 m_2^2 r_2 (k_1 k_2 + k_1 r_2 \omega_c + k_1 m_2 \omega_c^2 + k_2 m_2 \omega_c^2 + m_2 r_2 \omega_c^3)$$

– Variance of the relative displacement between wheel and vehicle body $z_2 - z_1$ (square of $\sigma_{z_2 - z_1}$)

$$\sigma^2_{z_2 - z_1} = c_{rv} \frac{k_1 (k_2 m_1 + k_2 m_2 + m_1 r_2 \omega_c + m_2 r_2 \omega_c + m_1 m_2 \omega_c^2)}{r_2 (D_S)} \tag{5.22}$$

The main difference between the formulae referring to the 1S-PSD ((5.14)–(5.19)) and those referring to 2S-PSD ((5.20)–(5.22)) is that in the first formulae the running condition parameters A_b and v are always not mixed with system model parameters $(m_1, m_2, k_1, k_2, r_2)$. The opposite occurs for 2S-PSD formulae in which running conditions parameters ω_c, c_{rv} are mixed with model parameters $(m_1, m_2, k_1, k_2, r_2)$.

This implies that for 1S-PSD excitation, the minima of $\sigma_{\ddot{z}_2}$, σ_{F_z}, $\sigma_{z_2 - z_1}$ (as function of the suspension parameters) do not depend on running conditions (A_b, v).

5.1.3 Parameter Sensitivity Analysis

The dynamic response of the road vehicle system model in Fig. 5.1 is analysed on the basis of Eqs. (5.14), (5.16), (5.18) and (5.20), (5.21), (5.22). A typical small-size sports car is taken into consideration (vehicle data in Table 5.1). All graphical results in the chapter refer to this vehicle, thus they do not have a general meaning and might be even qualitatively inaccurate for another (quite

different) vehicle. However, the formulae derived in the previous paragraph do have a general meaning and can be used for simulating the comfort, road holding and working space of every road vehicle that could be modeled as in Fig. 5.1.

The results of the parameter sensitivity analysis are shown in Figs. 5.3, 5.4 and 5.5.

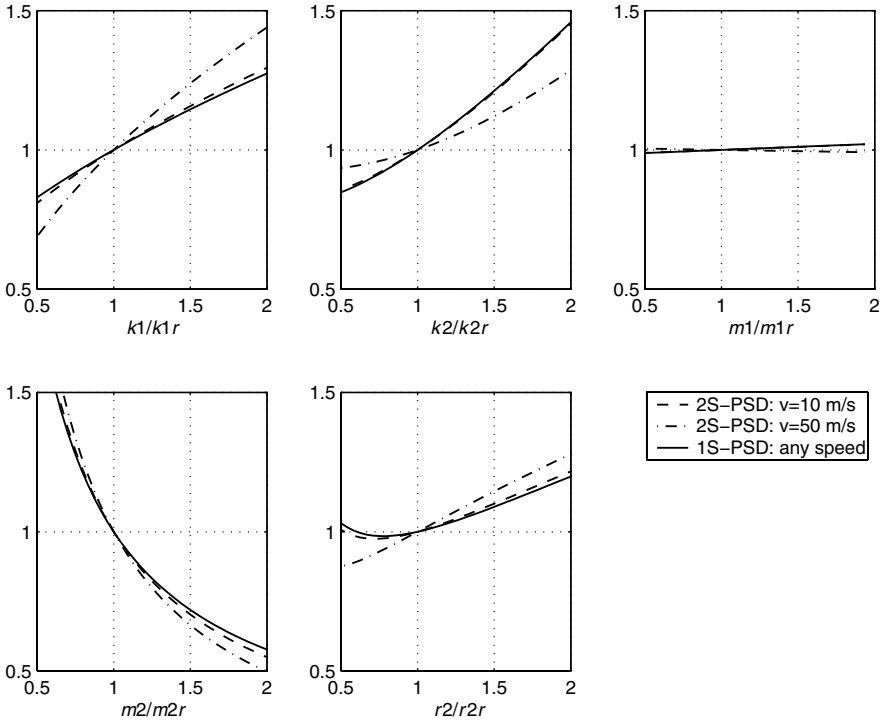


Fig. 5.3. $\sigma_{\ddot{z}_2}/\sigma_{\ddot{z}_2\,r}$: non-dimensional standard deviation of the vertical body acceleration as function of model parameters. Data of the reference vehicle in Table 5.1, running condition data in Table 5.2. Each diagram has been obtained by varying one single parameter, the other ones being constant and equal to those of the reference vehicle

The parameters are varied within wide ranges. The data are presented in non-dimensional form, i.e. the standard deviation of interest σ_j is divided by the corresponding one (σ_{jr}) computed by considering the parameters at their reference values reported in Table 5.1, i.e.

$$\sigma_{\ddot{z}_2\,r} = \sigma_{\ddot{z}_2}(m_{1r}, m_{2r}, k_{1r}, k_{2r}, r_{2r}) \tag{5.23}$$

$$\sigma_{Fz\,r} = \sigma_{Fz}(m_{1r}, m_{2r}, k_{1r}, k_{2r}, r_{2r}) \tag{5.24}$$

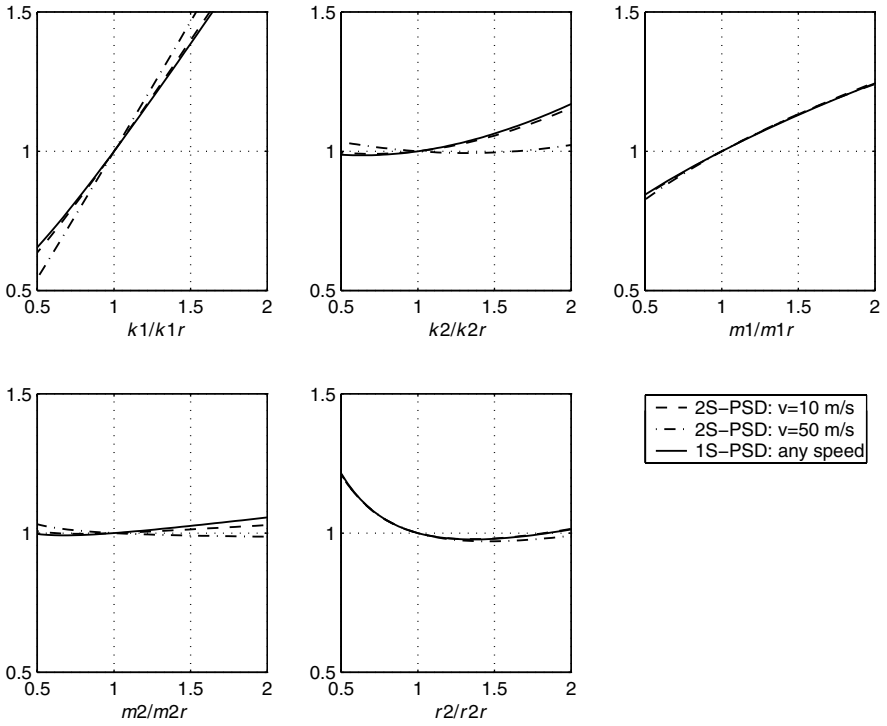


Fig. 5.4. $\sigma_{Fz}/\sigma_{Fz\,r}$: non-dimensional standard deviation of road holding as function of model parameters. Data of the reference vehicle in Table 5.1, running condition data in Table 5.2. Each diagram has been obtained by varying one single parameter, the other ones being constant and equal to those of the reference vehicle

$$\sigma_{z_2-z_1\,r} = \sigma_{z_2-z_1}(m_{1r}, m_{2r}, k_{1r}, k_{2r}, r_{2r}) \tag{5.25}$$

The non-dimensional standard deviations given by Eqs. (5.14), (5.16), (5.18) do not depend on vehicle speed. The opposite occurs for the non-dimensional standard deviations derived from Eqs. (5.20)–(5.22) referring to excitation given by 2S-PSD (Eq. (5.8)). For this reason these non-dimensional standard deviations are analysed at two different vehicle speeds, low speed (10 m/s) and high speed (50 m/s).

Standard Deviation of the Vertical Body Acceleration (Discomfort)

By inspection of Fig. 5.3 one may notice that

- the tyre radial stiffness k_1 influences significantly $\sigma_{\ddot{z}_2}$ (the influence is stronger at high speed considering the 2S-PSD),
- $\sigma_{\ddot{z}_2}$ increases with the suspension stiffness k_2,

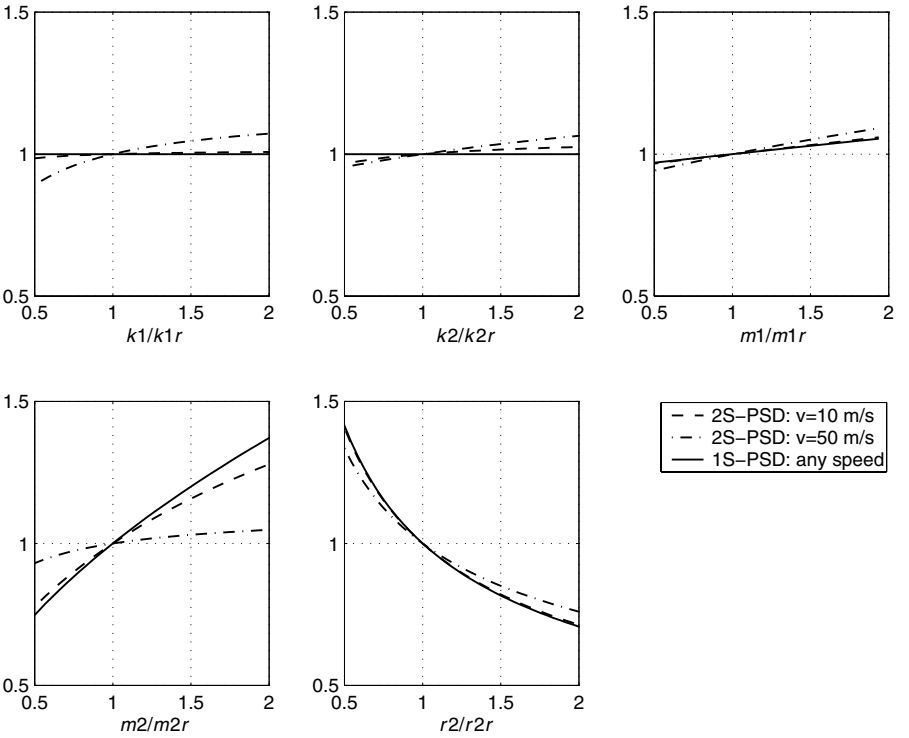


Fig. 5.5. $\sigma_{\ddot{z}_2-z_1}/\sigma_{\ddot{z}_2-z_1\,r}$: non-dimensional standard deviation of working space as function of model parameters. Data of the reference vehicle in Table 5.1, running condition data in Table 5.2. Each diagram has been obtained by varying one single parameter, the other ones being constant and equal to those of the reference vehicle

- $\sigma_{\ddot{z}_2}$ does not depend significantly on the wheel mass m_1,
- $\sigma_{\ddot{z}_2}$ depends strongly on the vehicle body mass m_2,
- the suspension damping r_2 has influence on the standard deviation $\sigma_{\ddot{z}_2}$.

Some of the above considerations can be derived by inspection of Eq. (5.14) or (5.20).

Standard Deviation of the Vertical Force (Road Holding)

Figure 5.4 shows that

- σ_{Fz} depends linearly on the tyre stiffness k_1,
- σ_{Fz} increases with the suspension stiffness k_2 (almost the opposite occurs at high speed considering the 2S-PSD),
- σ_{Fz} increases with the wheel mass m_1,
- σ_{Fz} does not depend significantly on the vehicle body mass m_2,

– the suspension damping r_2 has significant influence on the standard deviation σ_{Fz}.

Some of the above considerations can be derived by inspection of Eq. (5.16) or (5.21).

Standard Deviation of the Relative Displacement Wheel and Vehicle Body (Working Space)

By inspection of Fig. 5.5, one may notice that

– $\sigma_{z_2-z_1}$ is not influenced by k_1 and k_2 for the 1S-PSD excitation,
– the remarkable influence of m_2 on $\sigma_{z_2-z_1}$ is less important at high speed for the 2S-PSD excitation,
– $\sigma_{z_2-z_1}$ is strongly influenced by the suspension damping.

The above considerations can be derived by inspection of Eq. (5.18) or (5.22).

5.2 Passively Suspended Vehicle System Optimisation

5.2.1 Problem Formulation

Here the main issues of the solution of the MOP (Sect. 2.10) problem of a vehicle running on rough road are given. Let us consider a system model as the one depicted in Fig. 5.1. The relevant objective functions (i.e. performance indices) of the system are as shown in the previous sections, discomfort, road holding, working space. These objective functions are conflicting, i.e. improving one of them implies at least worsening another one. Usually designers are interested in finding the best compromise among these objective functions by varying the system's model design variables (e.g. suspension stiffness k_2 and damping r_2).

Finding the best compromise implies, in our case, the contemporary minimisation of the objective functions. So a vector composed of many objective functions has to be minimised.

5.2.2 Optimal Performances and Suspension Design Variables

The mathematical procedure described in 3.7.1 has been used to optimise the design variables of the suspension of a road vehicle described by the simple system model in Fig. 5.1.

The design variables to be optimised are the stiffness k_2 and the damping r_2 of the suspension and the tyre radial stiffness k_1, the objective functions are $\sigma_{\ddot{z}_2}$ (discomfort), σ_{F_z} (road holding) and $\sigma_{z_2-z_1}$ (working space).

By analysing the expressions given by Eqs. (5.14) and (5.16) one can notice that both $\sigma_{\ddot{z}_2}$ and σ_{F_z} increase monotonically with k_1 (if reasonable variations

for the suspension design variables are considered) and $\sigma_{z_2-z_1}$ depends only on r_2. For this couple of reasons, the problem can be simplified by considering only two design variables, k_2 and r_2. The first result is that in any case the tyre radial stiffness k_1 has to be kept at the lower bound of the admissible range.

The following paragraphs describe in detail the derivation of the Pareto-optimal set for different combinations of objective functions.

Suspension Design Variables for Optimal $\sigma_{\ddot{z}_2}$,σ_{F_z} (1S-PSD)

By applying the procedure based on the constraints method presented in Sect. 3.7.1 we find the suspension setting for optimal $\sigma_{\ddot{z}_2} - \sigma_{F_z}$.

We consider two design variables (k_1 is kept at the minimum)

- k_2 suspension stiffness,
- r_2 damping constant.

The objective functions are

- $\sigma_{\ddot{z}_2}$: discomfort (Eq. (5.15)),
- σ_{F_z}: road holding (Eq. (5.17)).

Solving Eq. (5.15) with respect to k_2, a function of r_2 and $\overline{\sigma}^2_{\ddot{z}_2}$ is obtained:

$$k_2 = \sqrt{\frac{\overline{\sigma}^2_{\ddot{z}_2} m_2^2 r_2}{(m_1+m_2)} - \frac{k_1 r_2^2}{(m_1+m_2)}} \tag{5.26}$$

By substituting k_2 in Eq. (5.17), $\overline{\sigma}^2_{F_z}$ can be expressed as

$$\overline{\sigma}^2_{F_z} = (m_1+m_2)^2\big(\overline{\sigma}^2_{\ddot{z}_2} + \frac{k_1^2 m_1}{r_2(m_1+m_2)^2} - \frac{2k_1 m_1}{m_2 r_2 (m_1+m_2)}(S_S)\big) \tag{5.27}$$

$$(S_S) = \sqrt{\frac{\overline{\sigma}^2_{\ddot{z}_2} m_2^2 r_2}{(m_1+m_2)} - \frac{k_1 r_2^2}{(m_1+m_2)}}$$

We now compute the minimum of Eq. (5.27) by setting to zero the first derivative with respect to r_2

$$\frac{d\,\overline{\sigma}_{F_z}}{d\,r_2} = 0 \quad \rightarrow \quad r_2 = \frac{k_1^2 m_2^2 \overline{\sigma}^2_{\ddot{z}_2}}{m_2^2(\overline{\sigma}^2_{\ddot{z}_2})^2(m_1+m_2) + k_1^3} \tag{5.28}$$

By substituting r_2 in Eq. (5.27), the relationship between optimal $\sigma_{\ddot{z}_2}$ and optimal σ_{F_z} can be obtained (see Fig. 5.6)

$$\begin{gathered} \overline{\sigma}^{*2}_{F_z} = m_2(m_1+m_2)\overline{\sigma}^{*2}_{\ddot{z}_2} + \frac{m_1 k_1^3}{m_2^2 \overline{\sigma}^{*2}_{\ddot{z}_2}} \\ 2\sqrt{\frac{m_1 k_1^3 (m_1+m_2)}{m_2}} < \overline{\sigma}^{*2}_{F_z} < \infty, \quad \sqrt{\frac{m_1 k_1^3}{(m_1+m_2)m_2^3}} > \overline{\sigma}^{*2}_{\ddot{z}_2} > 0 \end{gathered} \tag{5.29}$$

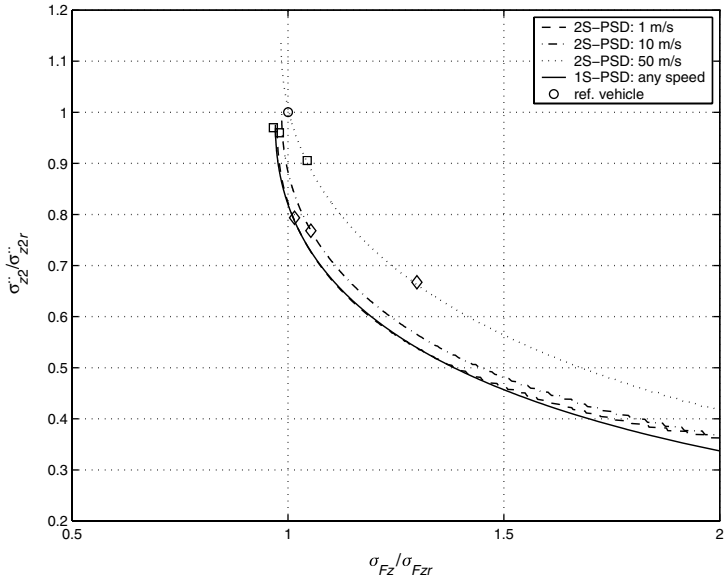


Fig. 5.6. Pareto-optimal sets for the $\sigma_{\ddot{z}_2}$–σ_{F_z} problem. Plot in non-dimensional form, vehicle data in Table 5.1, running condition data in Table 5.2. The points highlighted by using special symbols (*diamonds*, squares) refer to the points in Fig. 5.7

The corresponding expression in the optimal design variable space (Fig. 5.7) reads

$$k_2^* = \sqrt{\frac{m_2 k_1}{(m_1+m_2)}\left(\frac{m_2 k_1}{2(m_1+m_2)} - \frac{r_2^{*2}}{m_2} - \frac{\sqrt{k_1^2 m_2^2 - 4k_1 r_2^{*2}(m_1+m_2)}}{2(m_1+m_2)}\right)} \tag{5.30}$$

The last equation has been obtained by substituting Eqs. (5.15) and (5.17) into Eq. (5.29).

It is important to notice that Eq. (5.29), if taken without the constraints referring to $\overline{\sigma}^{*2}_{\ddot{z}_2}$ and $\overline{\sigma}^{*2}_{F_z}$, defines both Pareto-optimal solutions and non-Pareto-optimal solutions. This is due to the particular implementation of the constraints method we employed (see Sect. 3.7.1). Thus, in order to compute correctly the Pareto-optimal set, the inequalities have to be considered. The lower and upper bounds in Eq. (5.29) define the extrema ($f_{1(min)}$, $f_2(f_{1(min)})$ and $f_1(f_{2(min)})$, $f_{2(min)}$) corresponding to the two solutions that minimise one by one the two objective functions $f_1 = \overline{\sigma}_{F_z}{}^2$ and $f_2 = \overline{\sigma}_{\ddot{z}_2}{}^2$. The two extrema of the Pareto-optimal set can be obtained as follows. The minimum of $\overline{\sigma}_{\ddot{z}_2}{}^2$ is obtained by setting $\partial\,\overline{\sigma}_{\ddot{z}_2}{}^2/\partial\, r_2 = 0, \quad \partial\,\overline{\sigma}_{\ddot{z}_2}{}^2/\partial\, k_2 = 0$, so $f_{2(min)} = \overline{\sigma}^2_{\ddot{z}_2(min)} = 0$ (the corresponding $f_1(f_{2(min)}) = \overline{\sigma}^2_{F_z} = \infty$, see Eq. (5.29)). The minimum of $\overline{\sigma}_{F_z}{}^2$ is derived by setting $\partial\,\overline{\sigma}_{F_z}{}^2/\partial\, r_2 = 0\ \partial\,\overline{\sigma}_{F_z}{}^2/\partial\, k_2 =$

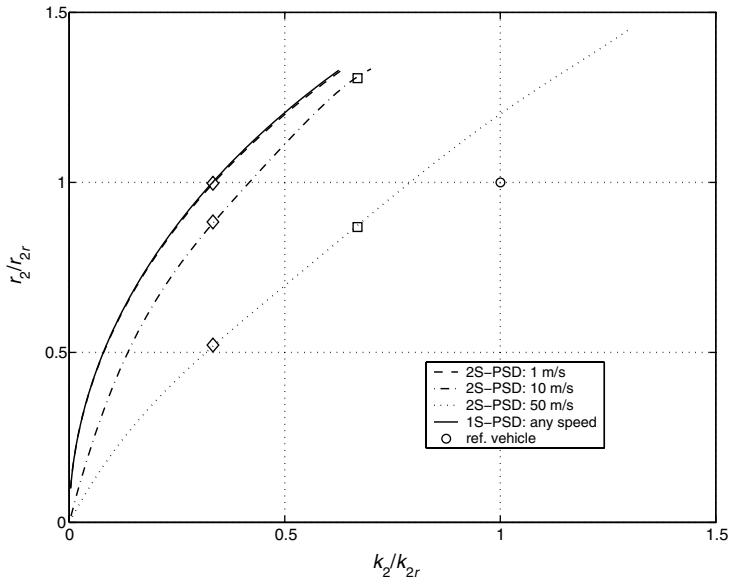


Fig. 5.7. Pareto-optimal sets (design variable space) for the $\sigma_{\ddot{z}_2}$–σ_{F_z} problem. Plot in non-dimensional form, vehicle data in Table 5.1, running condition data in Table 5.2. The points highlighted by using special symbols (*diamonds*, *squares*) refer to the points in Fig. 5.6

0 so $f_{1(min)} = \overline{\sigma}^2_{F_z(min)} = 2\sqrt{m_1 k_1^3 (m_1 + m_2)/m_2}$ and the corresponding $f_2(f_{1(min)}) = \overline{\sigma}^2_{\ddot{z}_2} = \sqrt{m_1 k_1^3/[(m_1 + m_2)m_2^3]}$.

Suspension Design Variables for Optimal $\sigma_{\ddot{z}_2}$,$\sigma_{z_2 - z_1}$ (1S-PSD)

The equation that defines the Pareto-optimal set for the discomfort – working space problem (Fig. 5.8) is

$$\overline{\sigma}^{*2}_{\ddot{z}_2} = \frac{k_1(m_1 + m_2)}{m_2^2 \overline{\sigma}^{*2}_{z_2 - z_1}} \tag{5.31}$$

Equation (5.31) has been obtained as shown in Sect. 3.7.1. Equation (5.31) contains the whole Pareto-optimal set being $f_{2(min)} = \overline{\sigma}^2_{\ddot{z}_2(min)} = 0$ (corresponding to $f_1(f_{2(min)}) = \overline{\sigma}^2_{z_2 - z_1} = \infty$) and $f_{1(min)} = \overline{\sigma}^2_{z_2 - z_1(min)} = 0$ (corresponding to $f_2(f_{1(min)}) = \overline{\sigma}^2_{\ddot{z}_2} = \infty$).

The equation of the Pareto-optimal set into the design variable space is

$$k_2^* = 0 \tag{5.32}$$

So, the Pareto-optimal set is the r_2 axis.

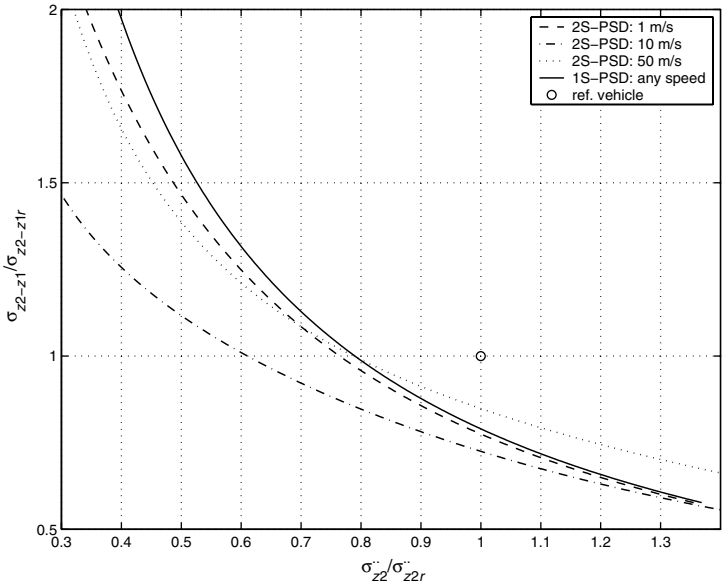


Fig. 5.8. Pareto-optimal sets for the $\sigma_{\ddot{z}_2}$–$\sigma_{z_2-z_1}$ problem. Plot in non-dimensional form, vehicle data in Table 5.1, running condition data in Table 5.2

Suspension Design Variables for Optimal $\boldsymbol{\sigma_{z_2-z_1}}$,$\boldsymbol{\sigma_{F_z}}$ (1S-PSD)

By applying again the procedure presented in Sect. 3.7.1 we derive the Pareto-optimal set for the road holding – working space problem (σ_{F_z},$\sigma_{z_2-z_1}$) (Fig. 5.9)

$$\begin{gathered}\overline{\sigma}^{*2}_{F_z} = \frac{m_1 k_1^2 \overline{\sigma}^{*2}_{z_2-z_1}}{m_1+m_2} - \frac{m_1^2 k_1^2 \overline{\sigma}^{*2}_{z_2-z_1}}{(m_1+m_2)^2} + \frac{k_1 (m_1+m_2)^3}{m_2^2 \overline{\sigma}^{*2}_{z_2-z_1}} \\ 2\sqrt{\frac{m_1 k_1^3 (m_1+m_2)}{m_2}} < \overline{\sigma}^{*2}_{F_z} < \infty, \quad \sqrt{\frac{(m_1+m_2)^5}{m_1 m_2^3 k_1}} > \overline{\sigma}_{z_2-z_1}{}^2 > 0\end{gathered} \tag{5.33}$$

The bounds of the inequalities can be obtained by computing the minimum of working space $f_{2(min)} = \overline{\sigma}^2_{z_2-z_1\,(min)} = 0$ (corresponding to $f_1(f_{2(min)}) = \overline{\sigma}^2_{F_z} = \infty$) and the minimum of the road holding $f_{1(min)} = \overline{\sigma}^2_{F_z\,(min)} = 2\sqrt{m_1 k_1^3 (m_1+m_2)/m_2}$, $f_2(f_{1(min)}) = \overline{\sigma}^2_{z_2-z_1} = \sqrt{(m_1+m_2)^5/(m_1 m_2^3 k_1)}$.

The Pareto-optimal set in the design variable space is

$$k_2^* = \frac{m_1 m_2 k_1}{(m_1+m_2)^2} \tag{5.34}$$

The Pareto-optimal set is the line parallel to r_2 axis (Fig. 5.10).

Suspension Design Variables for Optimal $\boldsymbol{\sigma_{\ddot{z}_2}}$,$\boldsymbol{\sigma_{F_z}}$,$\boldsymbol{\sigma_{z_2-z_1}}$ (1S-PSD)

The MOP problem can be reformulated by using the Constraints Method [59, 166, 238] as

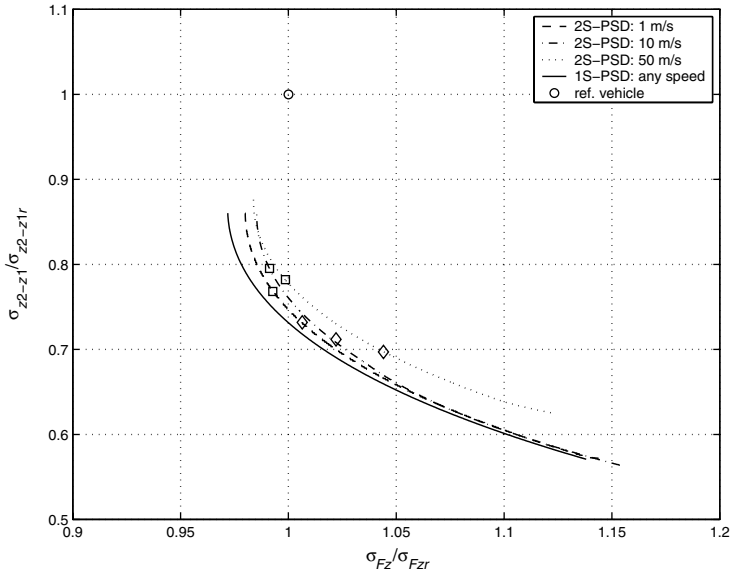


Fig. 5.9. Pareto-optimal sets for the $\sigma_{z_2-z_1}$–σ_{F_z} problem. Plot in non-dimensional form, vehicle data in Table 5.1, running condition data in Table 5.2. The points highlighted by using special symbols (*diamonds*, *squares*) refer to the points in Fig. 5.10

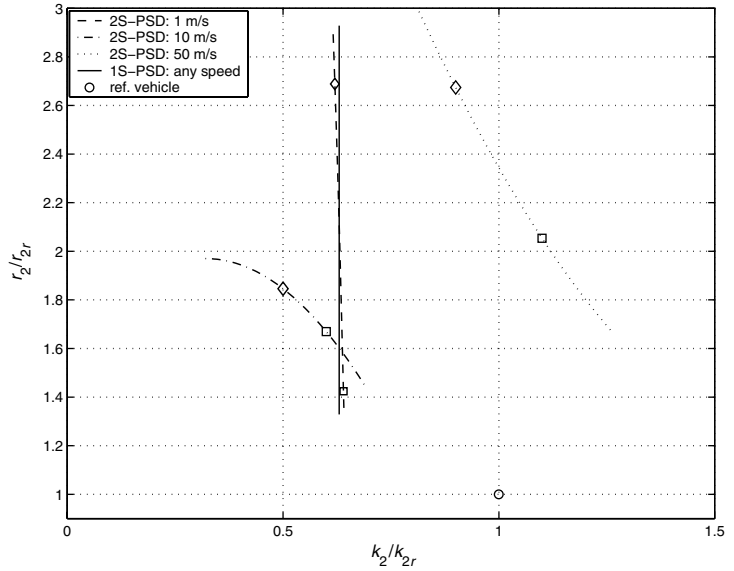


Fig. 5.10. Pareto-optimal sets (design variable space) for the $\sigma_{z_2-z_1}$–σ_{F_z} problem. Plot in non-dimensional form, vehicle data in Table 5.1, running condition data in Table 5.2. The points highlighted by using special symbols (*diamonds*, *squares*) refer to the points in Fig. 5.9

$$\begin{array}{l} \min \sigma^2_{z_2-z_1} = f_1(r_2) \quad \text{Eq.(5.18), (5.19)} \\ \sigma^2_{F_{z1}} = f_2(r_2, k_2) \leq \varepsilon_2 \quad \text{Eq.(5.16), (5.17)} \\ \sigma^2_{\ddot{z}_2} = f_3(r_2, k_2) \leq \varepsilon_3 \quad \text{Eq.(5.14), (5.15)} \end{array} \tag{5.35}$$

k_1 is not considered because it should be kept always at its minimum value. The variable k_2 is not explicitly represented in the objective function f_1. According to Theorem 3.8 (Sect. 3.7.1) there must exist at least two active constraints having opposite monotonicities with respect to k_2. So the two inequalities constraints in Eq. (5.35) are active (i.e. the equality holds) and the problem (5.35) is reduced to the solution of

$$\begin{array}{l} \sigma^2_{z_2-z_1} = f_1(r_2) \\ \sigma^2_{F_{z1}} = f_2(r_2, k_2) \\ \sigma^2_{\ddot{z}_2} = f_3(r_2, k_2) \end{array} \tag{5.36}$$

The analytical expression of the surface which contains the Pareto-optimal set in the $\sigma_{\ddot{z}_2}$, $\sigma_{F_{z1}}$, $\sigma_{z_2-z_1}$ space is derived from Eq. (5.36) by eliminating k_2, r_2.

$$\overline{\sigma}^{*2}_{F_z} = \overline{\sigma}^{*2}_{\ddot{z}_2}(m_2+m_1)^2 + \frac{m_1 k_1^2}{(m_2+m_1)}\overline{\sigma}^{*2}_{z_2-z_1} - \frac{2m_1 k_1}{m_2}(Q) \tag{5.37}$$

$$Q = \sqrt{\overline{\sigma}^{*2}_{\ddot{z}_2} m_2^2 \overline{\sigma}^{*2}_{z_2-z_1} - k_1(m_2+m_1)}$$

The corresponding Pareto-optimal design variables (k_2, r_2) can be readily calculated

$$r_2^* = \frac{m_2+m_1}{\overline{\sigma}^{*2}_{z_2-z_1}} \tag{5.38}$$

$$k_2^* = \frac{\sqrt{\overline{\sigma}^{*2}_{\ddot{z}_2} m_2^2 \overline{\sigma}^{*2}_{z_2-z_1} - k_1(m_2+m_1)}}{\overline{\sigma}^{*2}_{z_2-z_1}} \tag{5.39}$$

One may notice that r_2^*, k_2^* do not depend on $A_b \cdot v$, i.e. the optimal design variables settings do not depend on vehicle speed.

The Pareto-optimal set, defined by Eq. (5.37), is plotted in the $\sigma_{\ddot{z}_2}$, $\sigma_{F_{z1}}$, $\sigma_{z_2-z_1}$ space (objective functions space) in Fig. 5.11. Geometrically, in this case (three-dimensional space), the Pareto-optimal set is a surface. Figure 5.12 shows the projection of the Pareto-optimal set (surface) depicted in Fig. 5.11 onto the three planes $(\sigma_{F_{z1}}, \sigma_{z_2-z_1})$, $(\sigma_{\ddot{z}_2}, \sigma_{F_{z1}})$, $(\sigma_{\ddot{z}_2}, \sigma_{z_2-z_1})$. The Pareto-optimal sets in the design variable space (k_2, r_2) are shown in Fig. 5.13. k_1 is kept always at its minimum value.

The Pareto-optimal sets related to the problem formulated by considering only two objective functions (Sect. 5.2.2) are obviously on the border of the projection onto a bidimensional domain (plane) of the surface which represents the Pareto-optimal set related to the problem with three objective functions. Moreover, they constitute the boundaries of the Pareto-optimal surface.

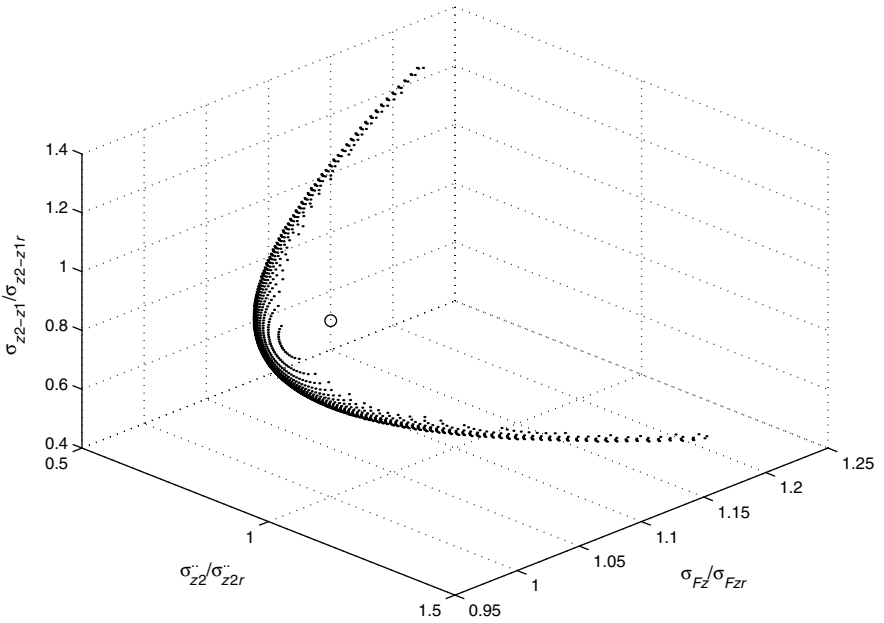


Fig. 5.11. Pareto-optimal set in the discomfort ($\sigma_{\ddot{z}_2}$) – road holding (σ_{F_z}) – working space ($\sigma_{z_2-z_1}$). Plot in non-dimensional form, vehicle data in Table 5.1, running condition data in Table 5.2. The 1S-PSD is considered. The reference vehicle is represented by the small circle

Suspension Design Variables for Optimal $\sigma_{\ddot{z}_2}$,σ_{F_z} (2S-PSD)

When the road irregularity is defined by the 2S-PSD, the analytical derivation of the Pareto-optimal set is rather impractical. So the Pareto-optimal set has been computed numerically, i.e. k_2 and r_2 have been varied and the response of the model in terms of $\sigma_{\ddot{z}_2}$ (Eq. (5.20)) and σ_{F_z} (Eq. (5.21)) has been computed. The Pareto-optimal solutions have been selected by applying directly the Pareto-optimality definition 2.7. The approximated Pareto-optimal set is plotted in Fig. 5.6 into the objective functions space and in Fig. 5.7 into the design variable space. Three different vehicle speeds (1, 10, 50 m/s) have been considered.

The relationship between k_2 and r_2 is different with respect to the case 1S-PSD as the vehicle speed increases (see Fig. 5.7).

Suspension Design Variables for Optimal $\sigma_{\ddot{z}_2}$, $\sigma_{z_2-z_1}$ (2S-PSD)

The same numerical procedure introduced in 3.7.1 has been used to compute the Pareto-optimal set referring to $\sigma_{\ddot{z}_2}$ and $\sigma_{z_2-z_1}$. The Pareto-optimal set is shown in Fig. 5.8. Three different vehicle speeds (1, 10, 50 m/s) have been considered.

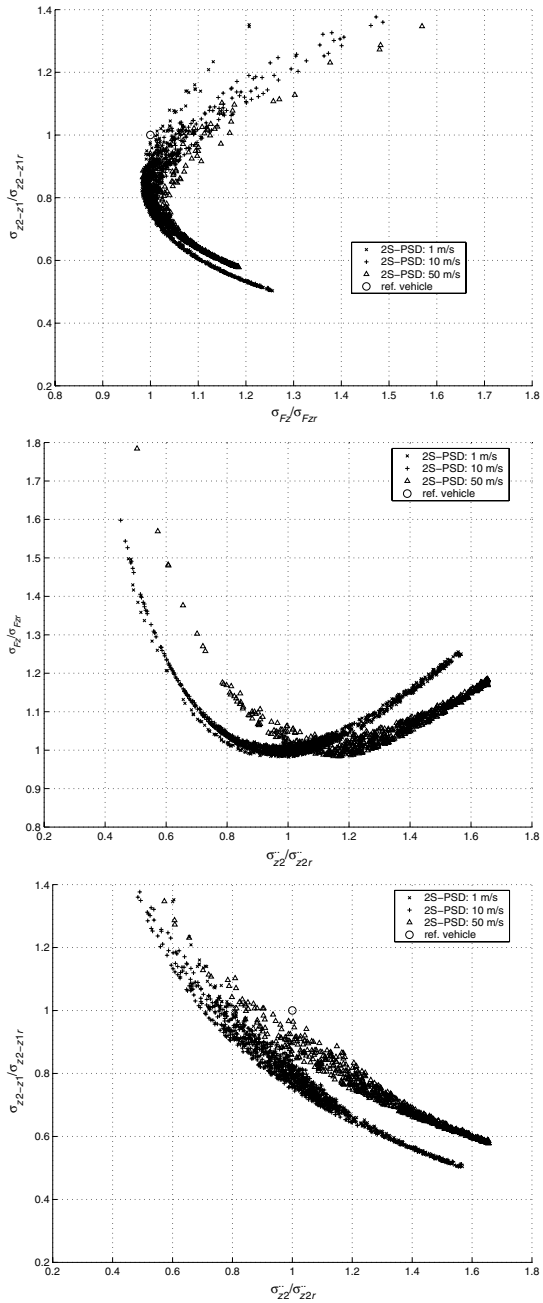


Fig. 5.12. Projection of the Pareto-optimal set shown in Fig. 5.11 in the discomfort ($\sigma_{\ddot{z}_2}$) road-holding (σ_{F_z}) working space ($\sigma_{z_2-z_1}$). Bidimensional projections. Plot in non-dimensional form, vehicle data in Table 5.1, running condition data in Table 5.2

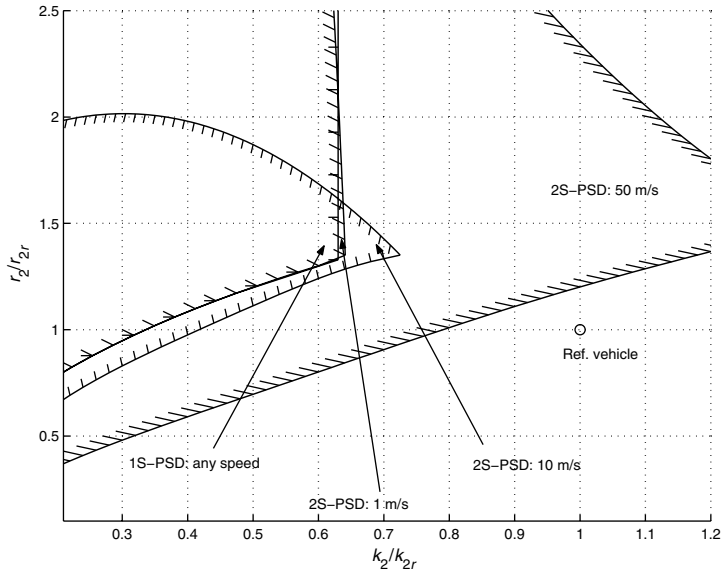


Fig. 5.13. Images of the Pareto-optimal set plotted in Fig. 5.12 onto the k_2–r_2 design variable space. Plot in non-dimensional form, vehicle data in Table 5.1, running condition data in Table 5.2. The optimal k_2, r_2 must lay within the indicated boundaries. The reference vehicle is not optimal within the speed range 1–50 m/s

The shape of the curve (Fig. 5.8) that represents the Pareto-optimal set $\sigma_{\ddot{z}_2}$–$\sigma_{z_2-z_1}$ is different from the curve obtained by considering the 1S-PSD, in this case the effect of the modified road irregularity spectrum is less important with respect to the $\sigma_{\ddot{z}_2}$,σ_{F_z} case.

Suspension Design Variables for Optimal $\sigma_{z_2-z_1}$–σ_{F_z} (2S-PSD)

The Pareto-optimal set considering $\sigma_{z_2-z_1}$ and σ_{F_z} is shown in Fig. 5.9 (design variable space) and Fig. 5.10 (objective functions space). Three different vehicle speeds (1, 10, 50 m/s) have been considered.

The relationship between k_2 and r_2 is different with respect to the case 1S-PSD (Fig. 5.10). This is due to the fact that $\sigma_{z_2-z_1}$ does not depend on k_2 when a road described by the 1S-PSD irregularity is considered.

Suspension Design Variables for Optimal $\sigma_{\ddot{z}_2}$–σ_{F_z}–$\sigma_{z_2-z_1}$ (2S-PSD)

A quasi–Monte Carlo method (see Sect. 3.4.2) has been used to approximate the Pareto-optimal set considering three objective functions, namely $\sigma_{\ddot{z}_2}$, σ_{F_z}, and $\sigma_{z_2-z_1}$. The projections of the Pareto-optimal surface onto the objective function space are shown in Fig. 5.12. The corresponding design variables are limited by the lines plotted in Fig. 5.13. Again three different vehicle speeds (1, 10, 50 m/s) have been considered.

5.3 System Model of an Actively Suspended Road Vehicle

5.3.1 Equations of Motion and Response to Stochastic Excitation

The adopted system models to analyse the dynamic behaviour of an actively suspended road vehicle are shown in Fig. 5.14. The mass m_1 represents the mass of the wheel plus part of the mass of the suspension arms, m_2 represents one fourth of the body mass [167, 247]. The excitation comes from the road irregularity ξ. The suspension characteristic is linear (k_1, k_2, G_1, G_2 are the most important parameters). The model (2 d.o.f.) is reputed sufficiently accurate for capturing the essential features related to discomfort, road holding and working space (see [4, 167, 227]). Tyre damping (usually very small) has been neglected. Tyre damping can couple the sprung m_2 and unsprung mass m_1 motions allowing reduction of the vehicle body vertical acceleration at frequencies near the wheel natural frequency [272].

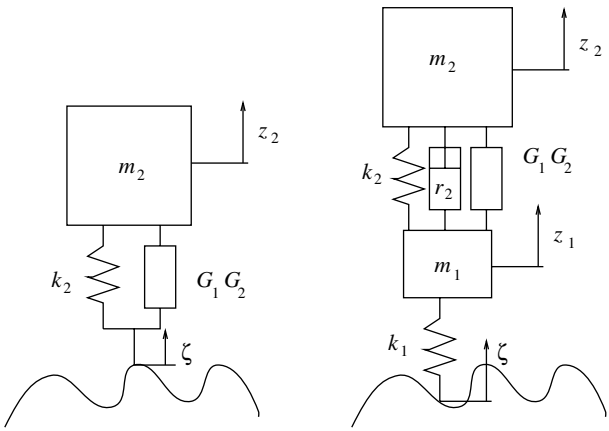


Fig. 5.14. Simplified actively suspended vehicle models, 1 d.o.f. (*left*) and 2 d.o.f. (*right*)

The equation of motion of the 1 d.o.f. model is

$$m_2\ddot{z}_2 + k_2(z_2 - \xi) + F_{act\,1} = 0 \tag{5.40}$$

The actuator is imagined to track faithfully a force demand signal determined by the controller. The actuator force is assumed to be function of both the relative displacement road – vehicle body and the absolute velocity of the vehicle body mass m_2 (sky-hook)

$$F_{act\,1} = G_1(z_2 - \xi) + G_2\dot{z_2} \tag{5.41}$$

where G_1 and G_2 are the controller gains.

The equations of motion of the 2 d.o.f. model are

$$\begin{array}{c} m_1\ddot{z}_1 - r_2\ (\dot{z}_2 - \dot{z}_1) - k_2\ (z_2 - z_1) + k_1\ (z_1 - \xi) - F_{act\,2} = 0 \\ m_2\ddot{z}_2 + \ r_2\ (\dot{z}_2 - \dot{z}_1) + k_2\ (z_2 - z_1) + F_{act\,2} = 0 \end{array} \tag{5.42}$$

The actuator force is assumed to be function of the relative displacement between wheel and vehicle body and of the absolute velocity of the vehicle body mass m_2 (sky-hook)

$$F_{act\,2} = G_1(z_2 - z_1) + G_2\dot{z_2} \tag{5.43}$$

where G_1 and G_2 are the controller gains.

The considered responses of the 2 d.o.f. vehicle model are respectively the vertical vehicle body acceleration ($\ddot{z}_2$), the force applied between road and wheel (F_z), the relative displacement between wheel and vehicle body ($z_2 - z_1$). By using the 1 d.o.f. model, only the vertical vehicle body acceleration ($\ddot{z}_2$), and the relative displacement between vehicle body and ground ($z_2 - \xi$) can be analysed.

Discomfort, road holding and working space are the *objective functions* (see Sect. 5.1.1) to which reference will be made in this section.

System Stability

The asymptotic stability of the two-system models has been tested according to Hurwitz criterion (necessary and sufficient condition).

The stability test for the 1 d.o.f. model yields $G_1 > 0$ and $G_2 > 0$, with $k_2 = 0$[2] . The same conditions ($G_1 > 0$ and $G_2 > 0$) hold for the 2 d.o.f. model, being $k_2 = 0$ and $r_2 = 0$[3].

5.3.2 Derivation of Standard Deviations in Analytical Form

The method to compute analytically the variance of a random variable described by a stationary and ergodic stochastic process is described in Sect. 5.1.2.

The analytical expressions of σ_j that will be presented in the following sub–sections have been derived by means of the analytical solution of the integral (Eq. (5.10)) reported in Appendix 5.6 and derived from [180].

[2] The effect of k_2 is actually the same as of G_1

[3] k_2 is zero for the reason reported in footnote 2; r_2 is set to zero for sake of simplicity. For non-vanishing r_2 the analytical expressions of $\ddot{z}_2$, σ_{F_z}, $\sigma_{z_2 - z_1}$ appear to be too involved to be simplified to such an extent useful for scientific and/or technical purposes

1 d.o.f. Model: Formulae Referring to the 1S-PSD (Eq. (5.7))

The analytical formulae giving discomfort and working space are reported. They have been obtained by solving analytically Eq. (5.10) (1S-PSD), setting $k_2 = 0$.

– Variance of the vehicle body acceleration $\ddot{z}_2$ (square of $\sigma_{\ddot{z}_2}$, discomfort)

$$\sigma^2_{\ddot{z}_2} = A_b v\, \overline{\sigma}^2_{\ddot{z}_2} \tag{5.44}$$

$$\overline{\sigma}^2_{\ddot{z}_2} = \frac{1}{2}\frac{G_1^2}{G_2 m_2} \tag{5.45}$$

– Variance of the relative displacement between road and vehicle body $z_2 - \xi$ (square of $\sigma_{z_2-\xi}$, working space)

$$\sigma^2_{z_2-\xi} = A_b\ v\, \overline{\sigma}^2_{z_2-\xi} \tag{5.46}$$

$$\overline{\sigma}^2_{z_2-\xi} = \frac{1}{2}\frac{G_2^2 + G_1 m_2}{G_2 G_1} \tag{5.47}$$

1 d.o.f. Model: Formulae Referring to the 2S-PSD (Eq. (5.8))

The analytical formulae giving discomfort and working space are reported. They have been obtained by solving analytically Eq. (5.10) (2S-PSD), setting $k_2 = 0$.

– Variance of the vehicle body acceleration $\ddot{z}_2$ (square of $\sigma_{\ddot{z}_2}$, discomfort)

$$\sigma^2_{\ddot{z}_2} = c_o \overline{\sigma}^2_{\ddot{z}_2} \tag{5.48}$$

$$\overline{\sigma}^2_{\ddot{z}_2} = \frac{1}{2}\frac{G_1^2(G_1 + \omega_c G_2)}{G_2 m_2(G_1 + \omega_c(G_2 + \omega_c m_2))} \tag{5.49}$$

$$\omega_c = a\, v$$

$$c_o = A_v\, a\, v$$

The coefficient a (rad/m) depends on the shape of the road irregularity spectrum.

– Variance of the relative displacement between road and vehicle body $z_2 - \xi$ (square of $\sigma_{z_2-\xi}$, working space)

$$\sigma^2_{z_2-\xi} = c_o \overline{\sigma}^2_{z_2-\xi} \tag{5.50}$$

$$\overline{\sigma}^2_{z_2-\xi} = \frac{1}{2}\frac{G_2^2 + G_1 m_2 + \omega_c G_2 m_2}{G_2(G_1 + \omega_c(G_2 + \omega_c m_2))} \tag{5.51}$$

2 d.o.f. Model: Formulae Referring to the 1S-PSD (Eq. 5.7)

The analytical formulae giving the discomfort, road holding and working space are reported. They have been obtained by solving analytically Eq. (5.10) (1S-PSD), setting $k_2 = 0$ and $r_2 = 0$. The corresponding expressions referring to a passively suspended road vehicle are reported in Sect. 5.1.2.

– Variance of the vehicle body acceleration $\ddot{z}_2$ (square of $\sigma_{\ddot{z}_2}$, discomfort)

$$\sigma^2_{\ddot{z}_2} = \frac{1}{2} A_b v \, \overline{\sigma}^2_{\ddot{z}_2} \tag{5.52}$$

$$\overline{\sigma}^2_{\ddot{z}_2} = \frac{G_1 k_1}{G_2 m_2} \tag{5.53}$$

– Variance of the force acting between road and wheel F_z (square of σ_{F_z}, road holding)

$$\sigma^2_{F_z} = \frac{1}{2} A_b \; v \, \overline{\sigma}^2_{F_z} \tag{5.54}$$

$$\begin{aligned} \overline{\sigma}^2_{F_z} = k_1(&-k_1 m_2^2 G_1 + k_1^2 m_2^2 + G_1^2 m_1^2 + 2G_1^2 m_1 m_2 + G_1^2 m_2^2 \\ &-2G_1 k_1 m_1 m_2 + G_2^2 k_1 m_1)/(G_1 m_2 G_2) \end{aligned} \tag{5.55}$$

– Variance of the relative displacement between wheel and vehicle body $z_2 - z_1$ (square of $\sigma_{z_2 - z_1}$, working space)

$$\sigma^2_{z_2 - z_1} = \frac{1}{2} A_b \; v \, \overline{\sigma}^2_{z_2 - z_1} \tag{5.56}$$

$$\overline{\sigma}^2_{z_2 - z_1} = \frac{k_1 m_2^2 + m_2 G_2^2 + G_2^2 m_1}{G_1 G_2 m_2} \tag{5.57}$$

2 d.o.f. Model: Formulae Referring to the 2S-PSD (Eq. (5.8))

The analytical formulae giving the discomfort, road holding and working space are reported. They have been obtained by solving analytically Eq. (5.11), (2S-PSD), with $k_2 = 0$ and $r_2 = 0$.

– Variance of the vehicle body acceleration $\ddot{z}_2$ (square of $\sigma_{\ddot{z}_2}$, discomfort)

$$\begin{aligned} \sigma^2_{\ddot{z}_2} = c_{rv}(&G_1 k_1^2 (G_1 + G_2 \omega_c + m_2 \omega_c^2))/(G_2 m_2 (m_1 m_2 \omega_c^4 \\ &+ G_2 m_1 \omega_c^3 + (G_1 m_1 + G_1 m_2 + k_1 m_2)\omega_c^2 \\ &+ G_2 k_1 \omega_c + G_1 k_1)) \end{aligned} \tag{5.58}$$

$$c_{rv} = \frac{1}{2} A_v \, a \, v$$

– Variance of the force acting between road and wheel F_z (square of σ_{F_z}, road holding)

$$\begin{aligned}\sigma^2_{F_z} = (c_{rv})(k_1^2(G_1^2(m_1+m_2)^2(G_1+G_2\omega_c+m_2\omega_c^2)\\ +(C_A)(C_B)+m_2^2(C_C)))/(G_1G_2m_2(G_1k_1+G_2k_1\omega_c\\ +G_1m_1\omega_c^2+G_1m_2\omega_c^2+k_1m_2\omega_c^2+G_2m_1\omega_c^3\\ +m_1m_2\omega_c^4))\end{aligned} \tag{5.59}$$

where

$$C_A = G_2^2m_1 - 2G_1m_1m_2 - 2G_1m_2^2$$

$$C_B = G_1k_1 + G_2k_1\omega_c + G_1m_2\omega_c^2 + k_1m_2\omega_c^2$$

$$\begin{aligned}C_C = G_1^2k_1 + G_1k_1^2 + G_1G_2k_1\omega_c + G_2k_1^2\omega_c + G_1^2m_1\omega_c^2 + G_1^2m_2\omega_c^2\\ +2G_1k_1m_2\omega_c^2 + k_1^2m_2\omega_c^2 + G_1G_2m_1\omega_c^3\end{aligned}$$

– Variance of the relative displacement between wheel and vehicle body $z_2 - z_1$ (square of $\sigma_{z_2-z_1}$, working space)

$$\begin{aligned}\sigma^2_{z_2-z_1} = c_{rv}(k_1((G_2^2m_1m_2+k_1m_2^3)\omega_c^2+(G_2^3m_1+G_2k_1m_2^2)\omega_c\\ +G_1G_2^2m_1+G_1G_2^2m_2+G_1k_1m_2^2))/(G_1G_2m_2(m_1m_2\omega_c^4\\ +G_2m_1\omega_c^3+(G_1m_1+G_1m_2+k_1m_2)\omega_c^2+G_2k_1\omega_c\\ +G_1k_1))\end{aligned} \tag{5.60}$$

5.3.3 Comparison of 1S-PSD and 2S-PSD Formulae

The main difference between the formulae referring to the 1S-PSD (Eqs. (5.44), (5.46), (5.52), (5.54) and (5.56)) and those referring to 2S-PSD (Eqs. (5.48), (5.50), (5.58), (5.59) and (5.60)) is that in the first formulae the running condition parameters A_b and v are always not mixed with system model design variables. The opposite occurs for 2S-PSD formulae in which running conditions parameters ω_c, c_{rv} are mixed with model design variables. This implies that for 1S-PSD excitation, the minima of $\sigma_{\ddot{z}_2}$, $\sigma_{z_2-\xi}$ as function of the suspension parameters do not depend on running conditions (A_b, v) (road data in Table 5.2). The opposite occurs for the 2S-PSD, for this reason the following analysis reported will be performed at three different vehicle speeds.

5.3.4 Validation Problems and Usefulness of the Presented Theory

Actual active or semi-active vehicle suspension systems are really complex. They are defined by many parameters (e.g. 100 or more). Thus, it was not possible to attempt a validation of Eqs. (5.44)–(5.60) as it was not possible

to establish a direct relationship between the actual system and the simple 2 d.o.f. model. Anyway one has to remember that, referring to a passively suspended vehicle, a quarter car model is often able to describe, with a sufficient approximation, the actual car dynamic behaviour (see [4, 167, 227]). Thus, Eqs. (5.44)–(5.60) are expected to be useful at least at the preliminary design stage of a car.

5.4 Actively Suspended Vehicle System Optimisation

The method described in Sect. 3.7.1 has been employed to compute the Pareto-optimal solutions (objective functions and suspension design variables).

5.4.1 Optimal Performances and Suspension Design Variables

1 d.o.f. Model: Pareto-optimal Set

The mathematical procedure described in Sect. 3.7.1 has been used to optimise both the objective functions (i.e. performance indices) and the design variables of the suspension of a road vehicle represented by the simple 1 d.o.f. system model in Fig. 5.14 (See Sect. 5.3.2)

The design variables to be optimised are the gains G_1 and G_2 of the active suspension system. The objective functions to be optimised are $\sigma_{\ddot{z}_2}$ (discomfort) and $\sigma_{z_2-\xi}$ (working space).

The following subsections describe in detail the derivation of the Pareto-optimal set.

Optimal $\sigma_{\ddot{z}_2}$,$\sigma_{z_2-\xi}$ and Optimal G_1, G_2 (1S-PSD)

By applying the procedure (based on constraints method) presented in Sect. (3.7.1), we find the Pareto-optimal set both in the design variables and in the objective functions space.

The Pareto-optimal set (Figs. 5.15 and 5.16) in the objective functions space is given by

$$\overline{\sigma}^{*2}_{\ddot{z}_2} = \frac{27}{64} \frac{1}{(\overline{\sigma}^{*2}_{z_2-\xi})^3} \tag{5.61}$$

the corresponding expression in the design variable space (Fig. 5.17) is

$$G_2^* = \sqrt{2m_2 G_1^*}; \tag{5.62}$$

The relationship between optimal G_1^* and G_2^* is identical to the one reported in [16, 111, 124].

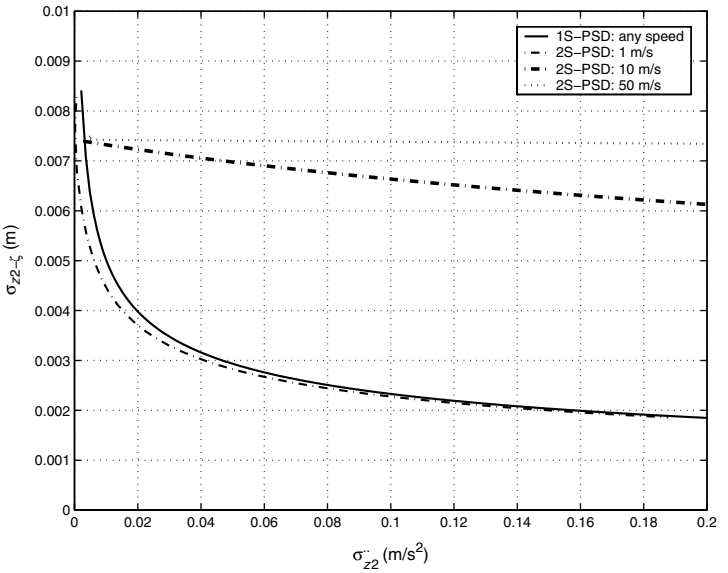


Fig. 5.15. 1 d.o.f. system model. Pareto-optimal set in the discomfort ($\sigma_{\ddot{z}_2}$) and working space ($\sigma_{z_2-\xi}$) space. Vehicle data in Table 5.3, running condition data in Table 5.2

Table 5.3. Data of the reference road vehicle taken into consideration

Design variable		Passive suspension	Active suspension
m_1	(kg)	229	229
m_2	(kg)	31	31
k_1	(N/m)	120,000	120,000
$k_2(reference)$	(N/m)	20,000	0
$r_2(reference)$	(Ns/m)	1,200	0

Optimal $\sigma_{\ddot{z}_2}$, $\sigma_{z_2-\xi}$ and Optimal G_1, G_2 (2S-PSD)

The Pareto-optimal set in the objective functions space (Figs. 5.15 and 5.16) is given by

$$\begin{aligned}\overline{\sigma}^{*2}_{\ddot{z}_2} = {} & 1/64(-4\omega_c(\overline{\sigma}^{*2}_{z_2-\xi})\sigma_1^{1/4} + 16\omega_c^2(\overline{\sigma}^{*2}_{z_2-\xi})^2\sigma_1^{1/4} - 3\sigma_1^{1/4} - \sigma_1^{3/4} - \sigma_2) \\ & (2\omega_c(\overline{\sigma}^{*2}_{z_2-\xi}) - 1)^2(12\omega_c(\overline{\sigma}^{*2}_{z_2-\xi})\sigma_1^{1/4} + 3\sigma_1^{1/4} + \sigma_2 + \sigma_1^{3/4}) \\ & (3\sigma_1^{1/4} + 4\omega_c(\overline{\sigma}^{*2}_{z_2-\xi})\sigma_1^{1/4} + \sigma_1^{3/4} + \sigma_2)^2/(\sigma_1^{1/4}(\overline{\sigma}^{*2}_{z_2-\xi})^3 \\ & (4\omega_c(\overline{\sigma}^{*2}_{z_2-\xi})\sigma_1^{1/4} - \sigma_2 - 3\sigma_1^{1/4} - \sigma_1^{3/4})(-\sigma_1^{1/4} + 4\omega_c(\overline{\sigma}^{*2}_{z_2-\xi})\sigma_1^{1/4} \\ & +\sigma_1^{3/4} + \sigma_2)^2) \end{aligned} \tag{5.63}$$

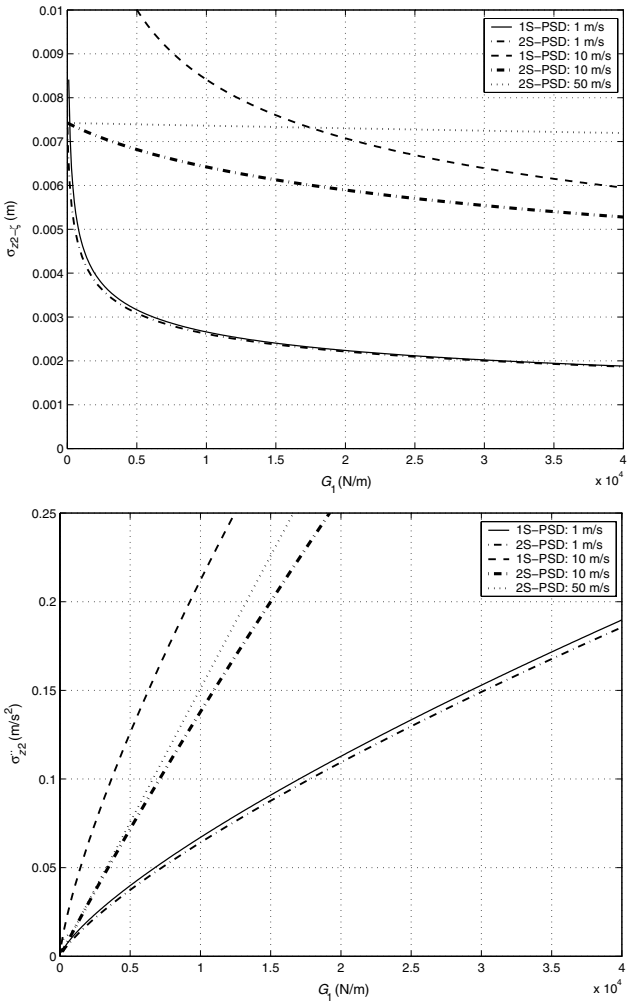


Fig. 5.16. 1 d.o.f. system model. Pareto-optimal set in the discomfort ($\sigma_{\ddot{z}_2}$) and working space ($\sigma_{z_2-\xi}$) spaces as function of G_1. Vehicle data in Table 5.3, running condition data in Table 5.2

where

$$\sigma_1 = 9 + 8\omega_c(\overline{\sigma}^{*2}_{z_2-\xi}) + 16\omega_c^2(\overline{\sigma}^{*2}_{z_2-\xi})^2$$

$$\begin{aligned}\sigma_2 = 2^{\frac{1}{2}}(&9\sigma_1^{\frac{1}{2}} + 16\sigma_1^{\frac{1}{2}}\omega_c(\overline{\sigma}^{*2}_{z_2-\xi}) - 64\sigma_1^{\frac{1}{2}}\omega_c^3(\overline{\sigma}^{*2}_{z_2-\xi})^3 \\ &-64\omega_c^2(\overline{\sigma}^{*2}_{z_2-\xi})^2 - 64\omega_c^3(\overline{\sigma}^{*2}_{z_2-\xi})^3 + 27 + 60\omega_c(\overline{\sigma}^{*2}_{z_2-\xi}) \\ &-256\omega_c^4(\overline{\sigma}^{*2}_{z_2-\xi})^4)^{\frac{1}{2}}\end{aligned}$$

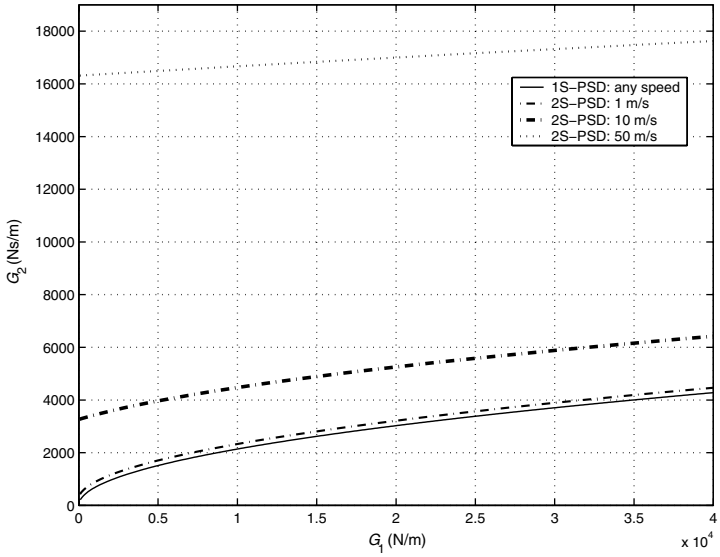


Fig. 5.17. 1 d.o.f. system model. Pareto-optimal set for the discomfort ($\sigma_{\ddot{z}_2}$) – working space ($\sigma_{z_2-\xi}$) problem represented onto the G_1G_2 space (relationship between gains G_1 and G_2 for obtaining the best compromise between discomfort and working space). Vehicle data in Table 5.3, running condition data in Table 5.2

The corresponding expression in the design variable space could not be simplified to a proper extent, so it has been derived only numerically and plotted in Fig. 5.17.

For the 1S-PSD the optimal gains G_1 and G_2 do not depend on vehicle speed. For the 2S-PSD, G_1 and G_2 do depend on vehicle speed in a remarkable way.

2 d.o.f. Model: Pareto-optimal Set

The mathematical procedure described in Sect. 3.7.1 has been used to optimise the design variables of the suspension of an actively suspended road vehicle described by the 2 d.o.f. model in Fig. 5.14 (see Sect. 5.3.2). The design variables to be optimised are the active suspension controller gains G_1, G_2 and the tyre radial stiffness k_1. The objective functions to be optimised are $\sigma_{\ddot{z}_2}$ (discomfort), σ_{F_z} (road holding) and $\sigma_{z_2-z_1}$ (working space).

By analysing the expressions given by Eqs. (5.52), (5.58) and Eqs. (5.54), (5.59) one can notice that $\sigma_{\ddot{z}_2}$, σ_{F_z}, $\sigma_{z_2-z_1}$ increase monotonically with k_1 (if reasonable variations for the suspension design variables are considered). Thus, the tyre radial stiffness k_1 has to be kept at the lower bound of the admissible range. The problem can be simplified by considering only two design variables G_1 and G_2.

The following subsections describe the derivation of Pareto-optimal sets for different combinations (2×2) of the three objective functions that have been considered ($\sigma_{\ddot{z}_2}$, σ_{F_z}, $\sigma_{z_2-z_1}$). The two controller gains G_1 and G_2 have been varied.

Optimal $\sigma_{\ddot{z}_2}$,σ_{F_z} (1S-PSD)

By applying the procedure based on the constraints method (see Sect. 3.7.1), we find the suspension setting for optimal $\sigma_{\ddot{z}_2} - \sigma_{F_z}$, leaving $\sigma_{z_2-z_1}$ free to vary.

The Pareto-optimal set (Fig. 5.18) is given by

$$\overline{\sigma}^{*2}_{F_z} = \frac{3(\overline{\sigma}^{*2}_{\ddot{z}_2})^2 m_2^4 + 4(\overline{\sigma}^{*2}_{\ddot{z}_2})^2 m_2^3 m_1 + 4k_1^3 m_1}{4\overline{\sigma}^{*2}_{\ddot{z}_2} m_2^2}$$
$$k_1\sqrt{\frac{k_1 m_1(3m_2+4m_1)}{m_2}} < \overline{\sigma}^{*2}_{F_z} < \infty, \quad 0 < \overline{\sigma}^{*2}_{\ddot{z}_2} < 2k_1\sqrt{\frac{k_1 m_1}{m_2^3(3m_2+4m_1)}} \tag{5.64}$$

The corresponding expression in the design variable space (as shown in Fig. 5.22) is

$$G_1^* = \frac{2k_1 m_2}{m_2 + 2m_1} \quad G_2^* >= \sqrt{\frac{k_1 m_2^3(4m_1 + 3m_2)}{m_1(m_2 + 2m_1)^2}} \tag{5.65}$$

The expression of the Pareto-optimal set referring to a passively suspended vehicle (Fig. 5.18) is already found in Sect. 5.1.2. Sensible improvements can be obtained by adopting the active system at all speeds.

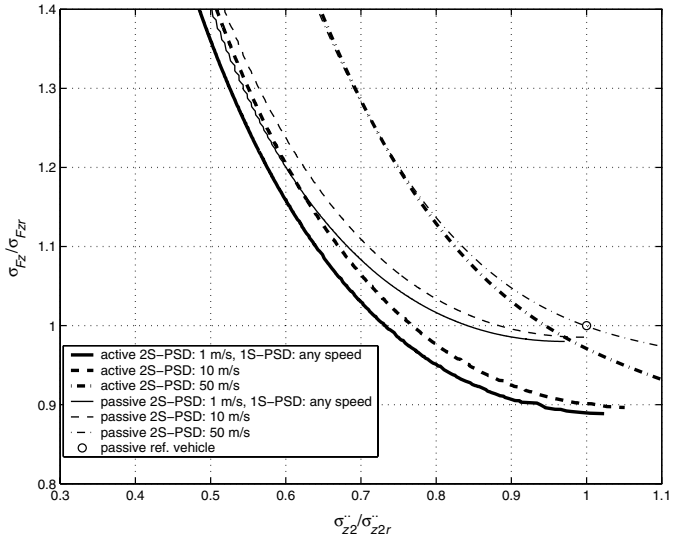


Fig. 5.18. 2 d.o.f. system model. Pareto-optimal sets for the $\sigma_{\ddot{z}_2}$–σ_{F_z} problem. Plot in non-dimensional form ($\sigma_{\ddot{z}_2\,r}$ and $\sigma_{F_z\,r}$ refer to the passively suspended reference vehicle), vehicle data in Table 5.3, running condition data in Table 5.2

It is important to notice that Eq. (5.64), if taken without the constraints referring to $\overline{\sigma}_{\ddot{z}_2}{}^2$ and $\overline{\sigma}_{F_z}{}^2$, defines both Pareto-optimal solutions and non-Pareto-optimal solutions. In order to compute correctly the Pareto-optimal set, the inequalities have to be considered.

Optimal $\sigma_{\ddot{z}_2}$,$\sigma_{z_2-z_1}$ (1S-PSD)

The Pareto-optimal set (Fig. 5.19) for optimal $\sigma_{\ddot{z}_2} - \sigma_{z_2-z_1}$ is given by

$$\overline{\sigma}^{*2}_{z_2-z_1} = \frac{k_1(m_1+m_2)}{\overline{\sigma}^{*2}_{\ddot{z}_2} m_2^2} \tag{5.66}$$

Equation (5.66) is a limit curve being the corresponding parameters unbounded. The curve contains the whole Pareto-optimal set being $\overline{\sigma}_{\ddot{z}_2}{}^2{}_{(min)} = 0$ corresponding to $\overline{\sigma}_{z_2-z_1}{}^2 = \infty$ and $\overline{\sigma}_{z_2-z_1}{}^2{}_{(min)} = 0$ corresponding to $\overline{\sigma}_{\ddot{z}_2}{}^2 = \infty$.

The corresponding expression of the Pareto-optimal set referring to a passive suspension (Fig. 5.19) can be found in Sect. 3.7.1.

It is important to notice that the Pareto-optimal set obtainable by considering a passive suspension is identical to the one obtainable with the active

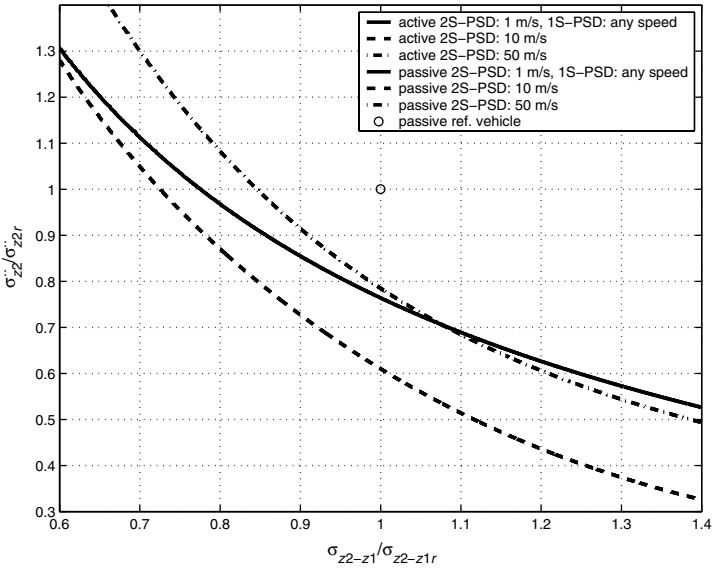


Fig. 5.19. 2 d.o.f. system model. Pareto-optimal sets for the $\sigma_{\ddot{z}_2}$–$\sigma_{z_2-z_1}$ problem. Plot in non-dimensional form ($\sigma_{\ddot{z}_2\,r}$ and $\sigma_{z_2-z_1\,r}$ refer to the passively suspended reference vehicle), vehicle data in Table 5.3, running condition data in Table 5.2. The passively and the actively suspended vehicles perform in the same way; this is due to fact that $k_2 = 0$ for the optimal passively suspended vehicle

system. This can be done only by setting a vanishing stiffness k_2 for the passive suspension system. This configuration is obviously of no practical interest. Lower performances can be obtained by setting a stiffness value (k_2) different from zero for the passive system.

Optimal $\sigma_{z_2-z_1}$,σ_{F_z} (1S-PSD)

The analytical derivation of the Pareto-optimal set for the $\sigma_{z_2-z_1}$,σ_{F_z} problem is impractical. The non-linear equations involved do not allow to simplify the Pareto-optimal set expression to a compact analytical form. So the Pareto-optimal set has been computed numerically. The Pareto-optimal set has been approximated by applying the definition of Pareto-optimal solution (see expression (2.11)). The Pareto-optimal set is shown in Fig. 5.20 into the objective functions space and in Fig. 5.22 into the design variable space.

Optimal $\sigma_{\ddot{z}_2}$,σ_{F_z},$\sigma_{z_2-z_1}$ (1S-PSD)

The MOP problem can be reformulated by using the constraints method (see Sect. 3.7.1) as

$$\begin{array}{c} \min_{G1,G2} \sigma_{F_z}^2 = \varepsilon_1 \\ \sigma_{z_2-z_1}^2 \leq \varepsilon_2 \\ \sigma_{\ddot{z}_2}^2 \leq \varepsilon_3 \end{array} \tag{5.67}$$

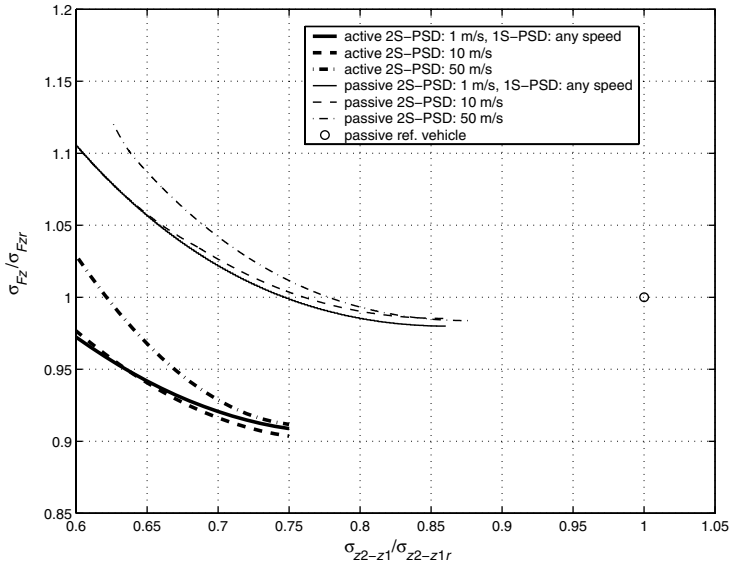


Fig. 5.20. 2 d.o.f. system model. Pareto-optimal sets for the $\sigma_{z_2-z_1}$–σ_{F_z} problem. Plot in non-dimensional form ($\sigma_{z_2-z_1\,r}$ and $\sigma_{F_z\,r}$ refer to the passively suspended reference vehicle), vehicle data in Table 5.3, running condition data in Table 5.2

(k_1 is not considered because it should be kept always at its minimum value) $\sigma^2_{F_z}$ (Eq. (5.54)) decreases monotonically with both G_1 and G_2 (if reasonable variations for the suspension design variables are considered). According to the monotonicity theorem[4] there must exist one active constraint with opposite monotonicity with respect to G_1 ($\sigma^2_{\ddot{z}_2}$, see Eq. (5.52))) and one active constraint with opposite monotonicity with respect to G_2 ($\sigma^2_{z_2-z_1}$, see Eq. (5.56)). So the two inequalities constraints in Eq. (5.67) are active (i.e. the equality holds) and the problem (5.67) is reduced to the solution of

$$\begin{array}{c} \sigma^2_{F_z} = h_1 \\ \sigma^2_{z_2-z_1} = h_2 \\ \sigma^2_{\ddot{z}_2} = h_3 \end{array} \tag{5.68}$$

The analytical expression of the surface which contains the Pareto-optimal set in the $\sigma_{\ddot{z}_2}$, σ_{F_z}, $\sigma_{z_2-z_1}$ space is then

$$\begin{aligned} \overline{\sigma}^{*2}_{F_z} = k_1^2 \overline{\sigma}^{*2}_{z_2-z_1} - \frac{1}{m_2 \overline{\sigma}^{*2}_{\ddot{z}_2}} k_1^3 - k_1 S_Q - \frac{2}{m_2} k_1 S_Q m_1 \\ + m_2^2 \overline{\sigma}^{*2}_{\ddot{z}_2} + 2 m_2 \overline{\sigma}^{*2}_{\ddot{z}_2} m_1 + \overline{\sigma}^{*2}_{\ddot{z}_2} m_1^2 \end{aligned} \tag{5.69}$$

where

$$S_Q = \sqrt{\overline{\sigma}^{*2}_{z_2-z_1} m_2^2 \overline{\sigma}^{*2}_{\ddot{z}_2} - k_1 m_1 - k_1 m_2}$$

The Pareto-optimal set (Eq. (5.69)) is plotted in the $\sigma_{\ddot{z}_2}$, σ_{F_z}, $\sigma_{z_2-z_1}$ space (objective functions space) in Fig. 5.21.

The Pareto-optimal sets related to the problem formulated by considering only two objective functions are obviously on the border of the projection onto a bidimensional space (plane) of the surface which represents the Pareto-optimal set related to the problem with three objective functions. Moreover, they constitute the boundaries of the Pareto-optimal surface.

The Pareto-optimal set into the design variable space is shown as the shaded area in Fig. 5.22. It is limited by the lines referring to the Pareto-optimal sets related to the problems formulated by considering two objective functions.

Optimal $\sigma_{\ddot{z}_2}$,σ_{F_z} (2S-PSD)

When the road irregularity is defined by the 2S-PSD the analytical derivation of the Pareto-optimal set is rather impractical. So the Pareto-optimal set has been computed numerically. The Pareto-optimal solutions have been selected by applying directly the definition of Pareto-optimality (see Definition 2.11). The approximated Pareto-optimal set is plotted in Fig. 5.18 into the objective functions space. Three different vehicle speeds (1, 10, 50 m/s) have been considered.

[4] If the variable x_i is explicitly represented in the objective function to be minimised, then there exists at least one active constraint with opposite monotonicity with respect to x_i [165, 200]

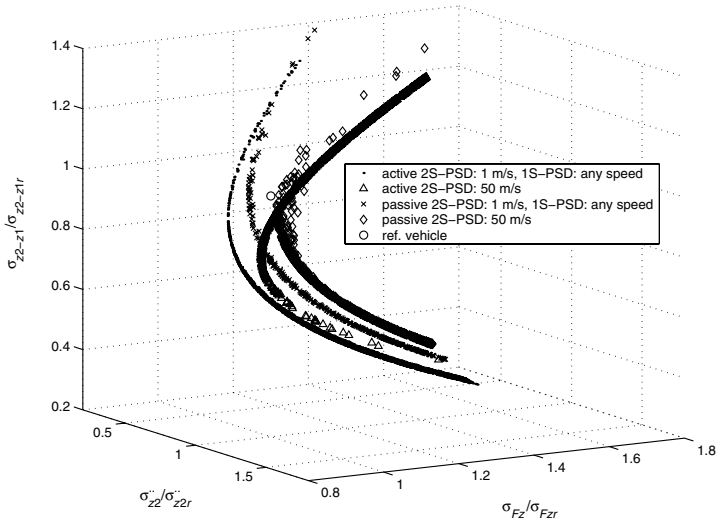


Fig. 5.21. 2 d.o.f. system model. Pareto-optimal set in the discomfort ($\sigma_{\ddot{z}_2}$) – road holding (σ_{F_z}) – working-space ($\sigma_{z_2-z_1}$) space. Plot in non-dimensional form ($\sigma_{\ddot{z}_2\,r}$, $\sigma_{z_2-z_1\,r}$ and $\sigma_{F_z\,r}$ refer to the passively suspended reference vehicle), vehicle data in Table 5.3, running condition data in Table 5.2. The 1S-PSD and 2 S-PSD are considered. The reference vehicle is represented by the small circle

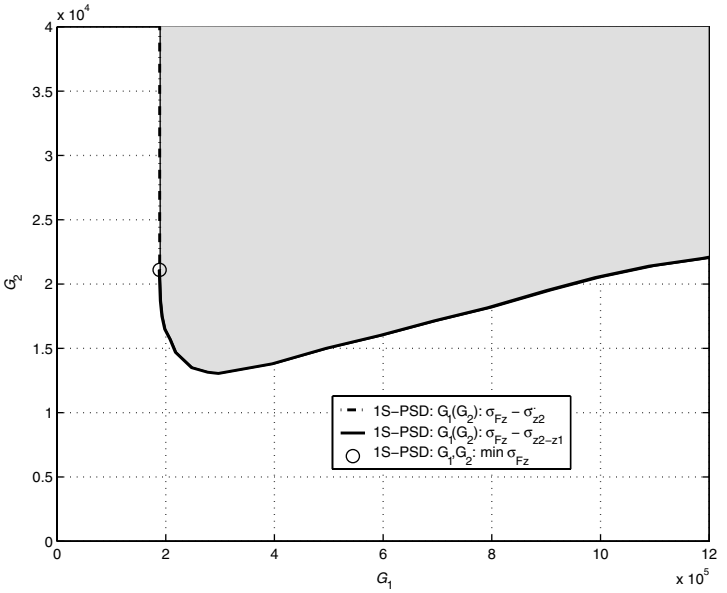


Fig. 5.22. Pareto-optimal set into the G_1, G_2 design variable space considering the 1S-PSD excitation. The Pareto-optimal set for the $\sigma_{\ddot{z}_2}$, σ_{F_z}, $\sigma_{z_2-z_1}$ problem is the shaded area. The Pareto-optimal sets for the $\sigma_{\ddot{z}_2}$, σ_{F_z} and σ_{F_z}, $\sigma_{z_2-z_1}$ problems are reported as dash–dot and continuous line respectively. Vehicle data in Table 5.3

Optimal $\sigma_{\ddot{z}_2}$, $\sigma_{z_2-z_1}$ (2S-PSD)

The numerical procedure introduced in Sect. 5.4.1 has been used to compute the Pareto-optimal set referring to $\sigma_{\ddot{z}_2}$ and $\sigma_{z_2-z_1}$.

The shape of the curve (Fig. 5.19) that represents the Pareto-optimal set $\sigma_{\ddot{z}_2}$–$\sigma_{z_2-z_1}$ is different from the curve obtained by considering the 1S-PSD.

Optimal $\sigma_{z_2-z_1}$–σ_{F_z} (2S-PSD)

The Pareto-optimal set considering $\sigma_{z_2-z_1}$ and σ_{F_z} is shown in Fig. 5.20. Three different vehicle speeds (1, 10, 50 (m/s)) have been considered.

Optimal $\sigma_{\ddot{z}_2}$–σ_{F_z}–$\sigma_{z_2-z_1}$ (2S-PSD)

A numerical search algorithm has been designed [86] and used to approximate the Pareto-optimal set considering three objective functions, namely $\sigma_{\ddot{z}_2}$, σ_{F_z}, and $\sigma_{z_2-z_1}$. Fig. 5.21 shows the comparison considering two different vehicle speeds (1, 50 m/s).

The Pareto-optimal set into the design variable space (G_1, G_2) is reported in Fig. 5.23 considering two different vehicle speeds (1, 50 m/s).

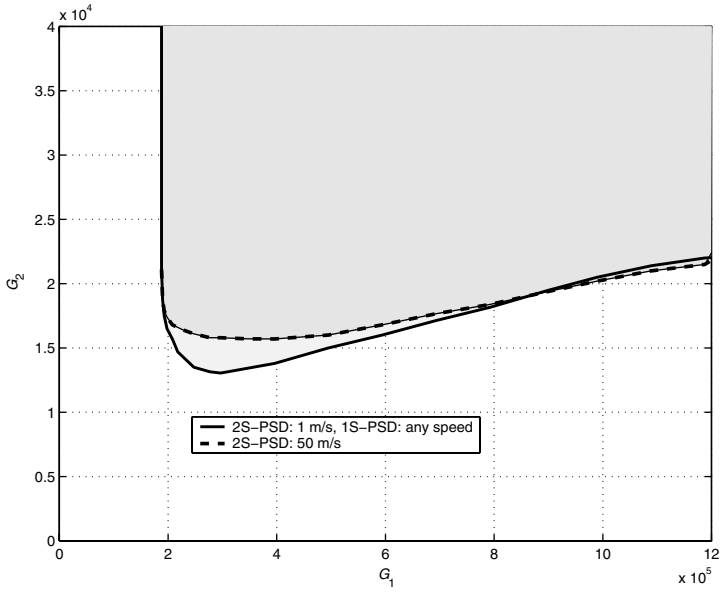


Fig. 5.23. Pareto-optimal set into the G_1, G_2 design variable space considering the 2S-PSD excitation. Vehicle data in Table 5.3, running condition data in Table 5.2

5.5 Conclusion

Simple analytical formulae have been derived symbolically in order to describe the active safety and ride comfort of actively and passively suspended road vehicles. In particular, the response to random excitation generated by the vertical road irregularity has been studied. Two different road irregularity spectra have been considered (1S-PSD and 2S-PSD). The derived analytical formulae should estimate with reasonable accuracy the dynamic behaviour of an actual road vehicle running on rough road.

Depending both on the power spectral density of the road irregularity (1S-PSD or 2S-PSD) and on the vehicle speed, the sensitivity of objective functions to suspension parameters variations is dramatically high.

By using the derived analytical formulae, a theoretically rigourous method – based multi-objective programming (MOP) (see Chap. 3) and monotonicity analysis (see Sect. 3.7.1) – has been applied to find the best trade-off among conflicting objective functions such as discomfort, road holding and working space.

If the excitation is given by a simplified road spectrum irregularity (1S-PSD), simple analytical formulae have been derived symbolically for the optimal compromise among discomfort, road holding and working space. Correspondingly, the optimal design variable parameters for the actively and the passively suspended vehicle have been derived symbolically, too.

If the excitation is defined by the simple road spectrum (1S-PSD) the optimal suspension settings do not depend on vehicle speed. The opposite occurs considering the (2S-PSD) excitation spectrum. For the passively suspended vehicle the best compromise between discomfort and road holding is obtained by increasing (or decreasing) both the suspension stiffness and damping. The relationship between stiffness and damping depends strongly on the vehicle speed if the best compromise between road holding and working space has to be obtained.

For a passively suspended vehicle when discomfort, road holding and working space have to be optimised contemporarily, the optimal suspension design variables (stiffness and damping) have to be set within well-defined ranges which have been derived (numerically) in the chapter. The tyre radial stiffness has always to be kept at its minimum value.

For the actively suspended vehicle Sect. 5.4 highlights the direct dependence of the gains on vehicle speed at least when simple car models are involved. The optimal active suspension system gains can vary significantly as function of the vehicle speed, and the variation can be set according to the preferred compromise among road holding, comfort and working space.

The comparison between the performances of the optimal active and the optimal passive suspension systems could be made on the basis of the derived analytical formulae. This has given the mathematical proof (relevant mostly from an academic point of view) that sensible improvements on the vehicle dynamic behaviour are obtainable by adopting active suspensions. The extents

of these improvements (referring to comfort, road holding and working space) are also defined in mathematical analytical form. In other terms, by the derived analytical formulae, the theoretical limit values of the performance of actual active suspension systems have been established.

5.6 Appendix: Tabulated Values of the Integral Form

The tabulated values of I_k for $k = 4$ and $k = 5$ reported in [180] are as follows:

$$I_k = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \frac{N(j\omega)\, N(-j\omega)}{D(j\omega)\, D(-j\omega)} ds$$

where

$$N(j\omega) = n_{k-1}(j\omega)^{k-1} + \cdots + n_0$$
$$D(j\omega) = d_k(j\omega)^k + \cdots + d_0$$

I_4 $(k = 4)$ reads

$$I_4 = \frac{n_3^2 c_m + (n_2^2 - 2n_1 n_3) d_0 d_1 d_4 + (n_1^2 - 2n_0 n_2) d_0 d_3 d_4 + n_0^2(-d_1 d_4^2 + d_2 d_3 d_4)}{2d_0 d_4(-d_0 d_3^2 - d_1^2 d_4 + d_1 d_2 d_3)}$$

where

$$c_m = -d_0^2 d_3 + d_0 d_1 d_2$$

.

I_5 $(k = 5)$ reads

$$I_5 = \frac{n_4^2 c_{m0} + (n_3^2 - 2n_2 n_4) c_{m1} + n_{m0} c_{m2} + (n_1^2 - 2n_0 n_2) c_{m3} + n_0^2 c_{m4}}{2d_0(d_1 c_{m4} - d_3 c_{m3} + d_5 c_{m2})}$$

where

$$n_{m0} = n_2^2 - 2n_1 n_3 + 2n_0 n_4$$
$$c_{m0} = 1/d_5(d_3 c_{m1} - d_1 c_{m2})$$
$$c_{m1} = -d_0 d_3 + d_1 d_2$$
$$c_{m2} = -d_0 d_5 + d_1 d_4$$
$$c_{m3} = 1/d_0(d_2 c_{m2} - d_4 c_{m1})$$
$$c_{m4} = 1/d_0(d_2 c_{m3} - d_4 c_{m2})$$

.

6

Optimal Handling and Active Safety of Road Vehicles

A basic study on the optimisation of the handling behaviour and active safety of road vehicles, derived from [157], is presented. The aim of the chapter is to describe how to tune the front and rear tyre cornering stiffness of a vehicle in order to obtain a preferred compromise among the many conflicting objective functions (i.e. performance indices) which describe the handling behaviour and active safety.

In the literature many papers – and even books – exist (see, e.g. – one for all – [167]) dealing with the problem of tyre stiffness selection, but they are mostly devoted to the presentation of system models, capable only to predict, more or less accurately, the vehicle dynamic behaviour. Indeed the tyre stiffness selection problem has hardly been addressed and solved as a pure optimisation problem.

To solve the addressed optimisation problem, a mathematical model of a road vehicle – with some unreferenced features – has been derived and used. A simple linear model has been used to derive in analytic mathematical form the dynamic response of the vehicle at a given step steer input. In the literature, a number of papers and books exist on this topic [46, 167, 193, 262], but here the specific objective functions relevant for performing the optimisation process have been derived. The objective functions derived in analytical form refer to a step-steering manoeuvre and represent the sideslip angle gain, the yaw velocity peak response time, the yaw velocity overshoot and the initial yaw acceleration. All of these metrics have shown a good and rather comprehensive correlation with subjective driver ratings [9, 33–35, 74, 103, 133, 217, 224].

From the linear vehicle model, a non-linear one has been derived by introducing the non-linear tyre characteristics in a piecewise linear form. In the literature non-linearities are mostly dealt with numerically [79, 167]. The piecewise linear form has been chosen to simplify as much as possible the complex non-linear tyre characteristic enabling the linear response of the vehicle being related to the non-linear one.

The optimisation has been performed on the basis of MOP (see Sect. 2.10); actually the objective functions referring to the metrics mentioned above are

conflicting [154,166]. In references [87,93,144,159,161] the adoption of multi-objective programming methods has been used effectively to solve complex problems related to vehicle handling.

In a complex optimisation problem such as the one dealt with in the chapter, the Pareto-optimal set has to be defined by computing its boundaries in the design variable space [154,166]; this computation is rather involved. A new numerical method [160] to compute the boundaries of the Pareto-optimal set has been used.

In the first part of the chapter, two system models (linear and non-linear) are introduced for the study of the handling performances of road vehicles. In the second part of the chapter, given a vehicle, the tyre cornering stiffness is tuned in order to get the best compromise among (the previously mentioned) four conflicting indices describing the handling behaviour. A validation of the optimisation process is proposed at the end of the chapter.

6.1 System Model

The adopted system model is shown in Fig. 6.1. It represents the well-known single-track vehicle model used to simulate the vehicle-handling behaviour [167]. Symbols appearing in Fig. 6.1 are reported in the chapter nomenclature.

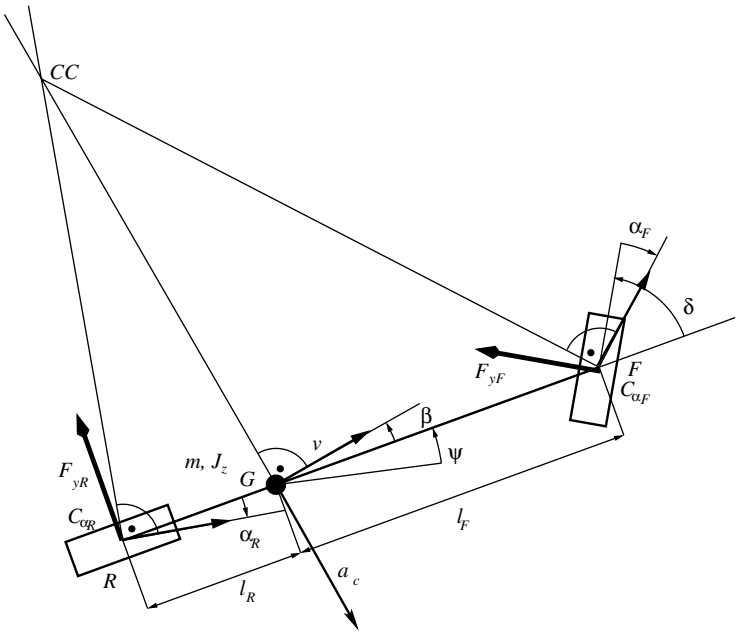


Fig. 6.1. Single track vehicle model

The lateral force F_y at an axle is considered either as a linear or a non-linear function of the slip angle α. Thus, two models will be considered

- a linear model (defined by linear lateral forces) and
- a non-linear model (defined by piecewise linear lateral forces)

The linear lateral forces at front and at rear axles (F_{yF}, F_{yR}) read respectively

$$\begin{aligned} F_{yF} &= C_{\alpha F}\alpha_F \\ F_{yR} &= C_{\alpha R}\alpha_R \end{aligned} \tag{6.1}$$

The non-linear lateral forces at front and at rear axles (F_{yF}, F_{yR}) are assumed to be piecewise linear. F_{yF}, F_{yR} are plotted in Fig. 6.2. They can be written as

$$\begin{aligned} F_{yF} = C_{\alpha F}\alpha_F \text{ if } & \quad |\alpha_F| < \alpha_{Fl} \\ F_{yF} = F_{yFl} \quad \text{if } & \quad |\alpha_F| \geq \alpha_{Fl} \\ F_{yR} = C_{\alpha R}\alpha_R \text{ if } & \quad |\alpha_R| < \alpha_{Rl} \\ F_{yR} = F_{yRl} \quad \text{if } & \quad |\alpha_R| \geq \alpha_{Rl} \end{aligned} \tag{6.2}$$

where F_{yFl} and F_{yRl} are respectively

$$\begin{aligned} F_{yFl} &= \mu_F F_{zF} = \mu_F mgl_R/l \\ F_{yRl} &= \mu_R F_{zR} = \mu_R mgl_F/l \end{aligned} \tag{6.3}$$

The angles α_F and α_R can be considered as function of β and $\dot{\psi}$ (Fig. 6.1). For small angles the following formulae hold [167]:

$$\begin{aligned} \alpha_F &= -\beta + \delta - l_F\frac{\dot{\psi}}{v} \\ \alpha_R &= -\beta + l_R\frac{\dot{\psi}}{v} \end{aligned} \tag{6.4}$$

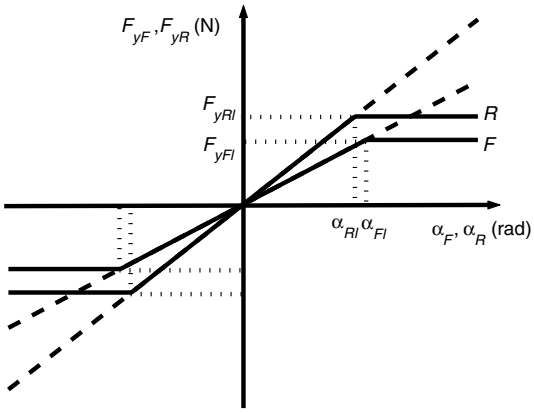


Fig. 6.2. Lateral forces transmitted by the tyres at front (F) and rear (R) axle as function of the lateral slip angle α

The curvature κ of the c.g. trajectory is

$$\kappa = \frac{1}{\rho} = \frac{d(\beta + \psi)}{d\xi} \tag{6.5}$$

where ξ is the traveled distance of the c.g. When $v = d\xi/dt$, the centripetal acceleration can be written as

$$a_c = \frac{v^2}{\rho} = v^2 \frac{(\dot{\beta} + \dot{\psi})}{v} = v(\dot{\beta} + \dot{\psi}) \tag{6.6}$$

6.1.1 Vehicle Response to a Steering Step Input: Linear Model

At first, a linear model is considered. If the speed of the vehicle is constant, considering Eqs. (6.1), (6.4) and (6.6), the equations of motion of the vehicle can be written as

$$mv\dot{\beta} + (C_{\alpha F} + C_{\alpha R})\beta + [mv^2 - (C_{\alpha R} l_R - C_{\alpha F} l_F)]\frac{\dot{\psi}}{v} = C_{\alpha F}\delta \tag{6.7}$$

$$J_z\ddot{\psi} - (C_{\alpha R} l_R - C_{\alpha F} l_F)\beta + (C_{\alpha F} l_F^2 + C_{\alpha R} l_R^2)\frac{\dot{\psi}}{v} = C_{\alpha F} l_F \delta \tag{6.8}$$

The steady-state conditions (ss) are derived from the above system by setting $\dot{\beta}$ and $\ddot{\psi}$ equal to zero

$$\beta_{ss} = \frac{C_{\alpha F}(C_{\alpha R} l_R l - m l_F v^2)\delta}{C_{\alpha F} C_{\alpha R} l^2 + mv^2(C_{\alpha R} l_R - C_{\alpha F} l_F)} \tag{6.9}$$

$$\dot{\psi}_{ss} = \frac{v C_{\alpha F} C_{\alpha R} l \delta}{C_{\alpha F} C_{\alpha R} l^2 + mv^2(C_{\alpha R} l_R - C_{\alpha F} l_F)} \tag{6.10}$$

$$a_{c\,ss} = \frac{v^2 C_{\alpha F} C_{\alpha R} l \delta}{C_{\alpha F} C_{\alpha R} l^2 + mv^2(C_{\alpha R} l_R - C_{\alpha F} l_F)}$$

By considering Eqs. (6.4) the steady-state values of α_F and α_R can be derived

$$\alpha_{F\,ss} = \frac{mv^2 C_{\alpha R} l_R \delta}{C_{\alpha F} C_{\alpha R} l^2 + mv^2(C_{\alpha R} l_R - C_{\alpha F} l_F)} \tag{6.11}$$

$$\alpha_{R\,ss} = \frac{mv^2 C_{\alpha F} l_F \delta}{C_{\alpha F} C_{\alpha R} l^2 + mv^2(C_{\alpha R} l_R - C_{\alpha F} l_F)} \tag{6.12}$$

It is well known that the solutions of Eqs. (6.7) and (6.8), when $\delta = 0$, are $\beta = \bar{\beta} e^{st}$ and $\dot{\psi} = \bar{\psi} e^{st}$. Considering the free motion of the system model, a solution of Eqs. (6.7) and (6.8) exists only if the characteristic polynomial reads [167]

$$s^2 + 2\varsigma s + \nu^2 = 0 \tag{6.13}$$

where

$$\varsigma = \frac{m(C_{\alpha F} l_F^2 + C_{\alpha R} l_R^2) + J_z(C_{\alpha F} + C_{\alpha R})}{2 J_z m v} \tag{6.14}$$

$$\nu^2 = \frac{C_{\alpha F} C_{\alpha R} l^2 + m v^2 (C_{\alpha R} l_R - C_{\alpha F} l_F)}{J_z m v^2} \tag{6.15}$$

ς is always positive. If ν^2 is positive, the real part of the two solutions s_1 and s_2 of the characteristic polynomial (6.13) are negative so the system is stable. So the system is stable for every speed v if $(C_{\alpha R} l_R - C_{\alpha F} l_F) > 0$, $\nu^2 > 0$. But if $(C_{\alpha R} l_R - C_{\alpha F} l_F) < 0$ there is a critical speed v_{cr} at which the vehicle motion becomes unstable

$$v_{cr}^2 = \frac{C_{\alpha F} C_{\alpha R} l^2}{m(C_{\alpha F} l_F - C_{\alpha R} l_R)} \tag{6.16}$$

To study the response to a steering step input ($\delta = \delta_{steady}\text{step}(t)$), Eqs. (6.7) and (6.8) have been considered. The steady-state gain of yaw velocity $\dot{\psi}$ and sideslip angle β are respectively

$$K_{\dot{\psi}} = \left(\frac{\dot{\psi}_{s\,s}}{\delta_{steady}} \right)$$

$$K_{\beta} = \left(\frac{\beta_{s\,s}}{\delta_{steady}} \right)$$

The Laplace transformation of Eqs. (6.7) and (6.8) is

$$(mvs + C_{\alpha F} + C_{\alpha R}) B(s) + [mv^2 - (C_{\alpha R} l_R - C_{\alpha F} l_F)] \frac{\dot{\Psi}(s)}{v} = C_{\alpha F} \Delta(s) \tag{6.17}$$

$$-(C_{\alpha R} l_R - C_{\alpha F} l_F) B(s) + (J_z v s + C_{\alpha F} l_F^2 + C_{\alpha R} l_R^2) \frac{\dot{\Psi}(s)}{v} = C_{\alpha F} l_F \Delta(s) \tag{6.18}$$

with $B(s)$, $\dot{\Psi}(s)$ and $\Delta(s)$ being the Laplace transformation of β, $\dot{\psi}$ and δ respectively. This system of equations can be solved and the Laplace transformation of sideslip angle β and yaw velocity $\dot{\psi}$ can be obtained respectively in the form

$$B(s) = K_{\beta} \frac{1 + T_{z1} s}{1 + \frac{2\varsigma}{\nu^2} s + \frac{1}{\nu^2} s^2} \Delta(s)$$

$$\dot{\Psi}(s) = K_{\dot{\psi}} \frac{1 + T_{z2} s}{1 + \frac{2\varsigma}{\nu^2} s + \frac{1}{\nu^2} s^2} \Delta(s) \tag{6.19}$$

where ς and ν are given by Eqs. (6.14) and (6.15) and

$$T_{z1} = \frac{J_z v}{C_{\alpha R} l_R l - l_F m v^2} \tag{6.20}$$

$$T_{z2} = \frac{m v l_F}{C_{\alpha R} l} \tag{6.21}$$

To analyse the response to a steering step input the Laplace transformation of the steering angle has been considered

$$\Delta(s) = \frac{\delta_{steady}}{s} \tag{6.22}$$

The inverse Laplace transformation of Eq. (6.19) gives

$$\begin{aligned} \frac{\beta(t)}{\delta_{steady}} &= K_\beta \left[1 + \frac{s_1 + 2\varsigma - T_{z1}\nu^2}{-s_1 + s_2} e^{s_1 t} + \frac{-s_2 - 2\varsigma + T_{z1}\nu^2}{-s_1 + s_2} e^{s_2 t}\right] \\ \frac{\dot{\psi}(t)}{\delta_{steady}} &= K_{\dot{\psi}} \left[1 + \frac{s_1 + 2\varsigma - T_{z2}\nu^2}{-s_1 + s_2} e^{s_1 t} + \frac{-s_2 - 2\varsigma + T_{z2}\nu^2}{-s_1 + s_2} e^{s_2 t}\right] \end{aligned} \tag{6.23}$$

where

$$s_1 = -\varsigma - \sqrt{\varsigma^2 - \nu^2} \tag{6.24}$$

$$s_2 = -\varsigma + \sqrt{\varsigma^2 - \nu^2} \tag{6.25}$$

for $t = 0$

$$\frac{\dot{\psi}(0)}{\delta_{steady}} = 0 \tag{6.26}$$

There are different responses depending on the sign of ς, ν and $\varsigma^2 - \nu^2$.

Case 1. For $\nu^2 > 0$ and $\varsigma^2 - \nu^2 > 0$, s_1 and s_2 from (6.24) and (6.25) are all real and negative, so the vehicle is stable and $\lim_{t\to\infty} \dot{\psi}(t)/\delta = K_{\dot{\psi}}$. In this case the yaw velocity reaches the maximum value at a time equal to ∞.

Case 2. For $\nu^2 > 0$ and $\varsigma^2 - \nu^2 < 0$, from Eqs. (6.24) and (6.25), s_1 and s_2 are complex conjugate. The vehicle is also stable because the real part of s_1 and s_2 are negative. Equation (6.23) can be written as

$$\frac{\dot{\psi}(t)}{\delta_{steady}} = K_{\dot{\psi}} \left[1 - e^{-\varsigma t}\left(\cos(\sqrt{\nu^2 - \varsigma^2}\, t) + \frac{\varsigma - T_{z2}\nu^2}{\sqrt{\nu^2 - \varsigma^2}} \sin(\sqrt{\nu^2 - \varsigma^2}\, t)\right)\right] \tag{6.27}$$

Case 3. For $\nu^2 < 0$ ($\sqrt{\varsigma^2 - \nu^2} > \varsigma$), the solution s_1 is real positive and the solution s_2 is real negative. So the vehicle is unstable because $\lim_{t\to\infty} \dot{\psi}(t) = \infty$.

In the present chapter only vehicles having $C_{\alpha R} l_R - C_{\alpha F} l_F > 0$ have been considered. In this case the vehicle is stable for every speed (actually in this case $\nu^2 > 0$ so *case 1* or *case 2* holds).

6.1.2 Vehicle Response to a Steering Step Input: Non-Linear Model

To analyse the motion of the non-linear model, different system models have been considered. Actually, the non-linearities in Eq. (6.2) refer to a piecewise linear tyre characteristic. In Fig. 6.3 the non-linear relationships among α_F, α_R, β and $\dot{\psi}$ are plotted (see Eq. (6.4)). Referring to Fig. 6.3, domain (1) with $|\alpha_F| < \alpha_{Fl}$ and $|\alpha_R| < \alpha_{Rl}$ has been defined. In this domain the motion of the system model is obtained by integrating the linear system of Eqs. (6.7) and (6.8). In domains (2^+) and (2^-) ($|\alpha_F| > \alpha_{Fl}$ and $|\alpha_R| < \alpha_{Rl}$), the system model is linear but the front tyres reach the maximum force level. In this domain the front tyre forces are constant. In domains (3^+) and (3^-) ($|\alpha_F| < \alpha_{Fl}$ and $|\alpha_R| > \alpha_{Rl}$), the rear tyres reach the maximum force level and these forces are constant. In domains $(4^{++}), (4^{+-}), (4^{-+})$ and (4^{--}) ($|\alpha_F| > \alpha_{Fl}$ and $|\alpha_R| > \alpha_{Rl}$), both front and rear forces do not change by varying the lateral slip angles α_F and α_R. Referring to the step-steering input ($\delta = \delta_{steady}\text{step}(t)$) the following considerations hold

- *Domains* (1). The motion of the system model is obtained by integrating Eqs. (6.7) and (6.8). The response of the system model depends on the sign of ς and ν as explained in Sect. 6.1.1. At steady state ($\delta = \delta_{steady}$), to keep $\beta_{s\,s}$ and $\dot{\psi}_{s\,s}$ within domain (1), from Eqs. (6.11) and (6.12) with $\alpha_{Fl} =$

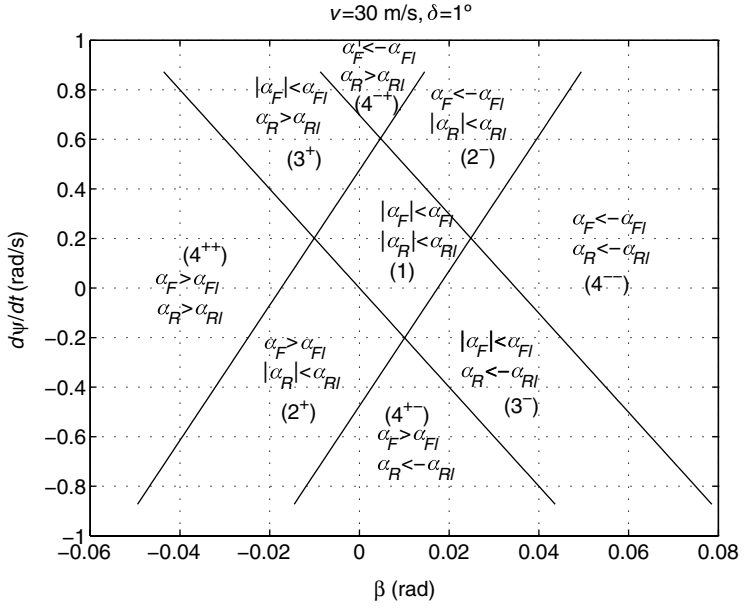


Fig. 6.3. Lateral slip at front and rear axles (α_F, α_R) as a function of sideslip body angle β and yaw velocity $\dot{\psi}$

$F_{ylF}/C_{\alpha F} = \mu_F mgl_R/(C_{\alpha F}l)$ and $\alpha_{Rl} = F_{ylR}/C_{\alpha R} = \mu_R mgl_F/(C_{\alpha R}l)$, it follows

$$|\delta_{steady}| < \frac{\min\{\mu_F, \mu_R\} g|C_{\alpha F}C_{\alpha R}l^2 + mv^2(C_{\alpha R}l_R - C_{\alpha F}l_F)|}{lC_{\alpha F}C_{\alpha R}v^2} = \tilde{\delta} \tag{6.28}$$

- *Domains* (2^+), (2^-). The motion of the system model in this domain is obtained by integrating the system of equations

$$mv\dot{\beta} + C_{\alpha R}\beta + [mv^2 - C_{\alpha R}l_R]\frac{\dot{\psi}}{v} = F_{ylF}\text{sign}(\alpha_F) = \frac{\mu_F mgl_R}{l}\text{sign}(\alpha_F) \tag{6.29}$$

$$J_z\ddot{\psi} - C_{\alpha R}l_R\beta + C_{\alpha R}l_R^2\frac{\dot{\psi}}{v} = F_{ylF}l_F\text{sign}(\alpha_F) = \frac{\mu_F mgl_R l_F}{l}\text{sign}(\alpha_F) \tag{6.30}$$

The steady-state condition of this system is given by (the subscript refers to domains (2^+) and (2^-))

$$\begin{aligned} \beta_{ss\,2} &= -F_{yFl}\text{sign}(\alpha_F)\frac{l_F mv^2 - C_{\alpha R}ll_F}{C_{\alpha R}l_R mv^2} \\ &= -\mu_F g\text{sign}(\alpha_F)\frac{l_F mv^2 - C_{\alpha R}ll_F}{C_{\alpha R}lv^2} \end{aligned} \tag{6.31}$$

$$\begin{aligned} \dot{\psi}_{ss\,2} &= F_{yFl}\text{sign}(\alpha_F)\frac{l}{l_R mv} \\ &= \mu_F g\text{sign}(\alpha_F)/v \\ |a_{c\,ss\,2}| &= \mu_F g \end{aligned} \tag{6.32}$$

and the values of $\alpha_{F\,ss}$ and $\alpha_{R\,ss}$ are given by the equations

$$\begin{aligned} \alpha_{F\,ss\,2} &= F_{yFl}\text{sign}(\alpha_F)\frac{mv^2 l_F - C_{\alpha R}l^2}{C_{\alpha R}l_R mv^2} + \delta \\ &= \mu_F g\text{sign}(\alpha_F)\frac{mv^2 l_F - C_{\alpha R}l^2}{C_{\alpha R}lv^2} + \delta \end{aligned} \tag{6.33}$$

$$\begin{aligned} \alpha_{R\,ss\,2} &= F_{yFl}\text{sign}(\alpha_F)\frac{l_F}{C_{\alpha R}l_R} \\ &= \mu_F mg\text{sign}(\alpha_F)\frac{l_F}{C_{\alpha R}l} \end{aligned} \tag{6.34}$$

In this domain, $\nu^2 > 0$ for every speed. The eigenvalues associated with this motion have all negative real part so the system model is stable. At steady state ($\delta = \delta_{steady}$), to keep $\beta_{s\,s\,2}$ and $\dot{\psi}_{s\,s\,2}$ within domains (2^+) and (2^-), from Eq. (6.33) and with $|\alpha_{F\,ss\,2}| > \alpha_{Fl}$, it follows

$$\delta > \frac{\alpha_{Fl}[C_{\alpha F}C_{\alpha R}l^2 + mv^2(C_{\alpha R}l_R - C_{\alpha F}l_F)]}{C_{\alpha R}l_R mv^2} = \frac{\mu_F g[C_{\alpha F}C_{\alpha R}l^2 + mv^2(C_{\alpha R}l_R - C_{\alpha F}l_F)]}{C_{\alpha F}C_{\alpha R}lv^2}$$

$$\text{or } \delta < -\frac{\alpha_{Fl}[C_{\alpha F}C_{\alpha R}l^2 + mv^2(C_{\alpha R}l_R - C_{\alpha F}l_F)]}{C_{\alpha R}l_R mv^2} = \frac{\mu_F g[C_{\alpha F}C_{\alpha R}l^2 + mv^2(C_{\alpha R}l_R - C_{\alpha F}l_F)]}{C_{\alpha F}C_{\alpha R}lv^2} \tag{6.35}$$

Additionally, it follows that with $|\alpha_{R\,ss\,2}| < \alpha_{Rl}$ the requisite $\mu_F < \mu_R$ must be satisfied ($\alpha_{Rl} = \mu_R mgl_F/(C_{\alpha R}l)$).

- *Domains* (3^+), (3^-). The motion of the system model is obtained by integrating the system of equations

$$\begin{aligned} mv\dot{\beta} + C_{\alpha F}\beta + [mv^2 + C_{\alpha F}l_F]\frac{\dot{\psi}}{v} &= C_{\alpha F}\delta + F_{ylR}\text{sign}(\alpha_R) \\ &= C_{\alpha F}\delta + \frac{\mu_R mgl_F}{l}\text{sign}(\alpha_R) \\ J_z\ddot{\psi} + C_{\alpha F}l_F\beta + C_{\alpha F}l_F^2\frac{\dot{\psi}}{v} &= C_{\alpha F}l_F\delta - F_{ylR}l_R\text{sign}(\alpha_R) \\ &= C_{\alpha F}l_F\delta - \frac{\mu_R mgl_F l_R}{l}\text{sign}(\alpha_R) \end{aligned}$$

The steady-state solution of this linear system is given by (the subscript refers to domains (3^+) and (3^-))

$$\begin{aligned} \beta_{ss\,3} &= -F_{yRl}\text{sign}(\alpha_R)\frac{l_R mv^2 + C_{\alpha F}ll_F}{C_{\alpha F}l_F mv^2} + \delta \\ &= -\mu_R g\text{sign}(\alpha_R)\frac{l_R mv^2 + C_{\alpha F}ll_F}{C_{\alpha F}lv^2} + \delta \qquad (6.36) \\ \dot{\psi}_{ss\,3} &= F_{yRl}\text{sign}(\alpha_R)\frac{l}{l_F mv} = \mu_R g\text{sign}(\alpha_R)/v \qquad (6.37) \\ |a_{c\,eq\,3}| &= \mu_R g \end{aligned}$$

and the values of α_F and α_R are given by

$$\begin{aligned} \alpha_{F\,ss\,3} &= F_{yRl}\text{sign}(\alpha_R)\frac{l_R}{C_{\alpha F}l_F} = \mu_R mg\text{sign}(\alpha_R)\frac{l_R}{C_{\alpha F}l} \\ \alpha_{R\,ss\,3} &= F_{yRl}\text{sign}(\alpha_R)\frac{mv^2 l_R + C_{\alpha F}l^2}{C_{\alpha F}l_F mv^2} - \delta \\ &= \mu_R g\text{sign}(\alpha_R)\frac{mv^2 l_R + C_{\alpha F}l^2}{C_{\alpha F}lv^2} - \delta \end{aligned}$$

In this domain, $\nu^2 < 0$ for every speed. The system model is unstable. At steady state ($\delta = \delta_{steady}$), to keep $\beta_{s\,s\,2}$ and $\dot{\psi}_{s\,s\,2}$ within domains (3^+) and

(3^-), the conditions $|\alpha_{F\,ss\,3}| < \alpha_{Fl}$ and $|\alpha_{R\,ss,\,3}| > \alpha_{Rl}$ must be satisfied respectively. This is equivalent to impose the condition $\mu_R < \mu_F$. It must be also verified that

$$\delta < \frac{\alpha_{Rl}[C_{\alpha F}C_{\alpha R}l^2 + mv^2(C_{\alpha R}l_R - C_{\alpha F}l_F)]}{C_{\alpha F}l_F mv^2} = \frac{\mu_R g[C_{\alpha F}C_{\alpha R}l^2 + mv^2(C_{\alpha R}l_R - C_{\alpha F}l_F)]}{C_{\alpha F}C_{\alpha R}lv^2}$$

$$\text{or} \quad \delta > -\frac{\alpha_{Rl}[C_{\alpha F}C_{\alpha R}l^2 + mv^2(C_{\alpha R}l_R - C_{\alpha F}l_F)]}{C_{\alpha R}l_R mv^2} \frac{\mu_R g[C_{\alpha F}C_{\alpha R}l^2 + mv^2(C_{\alpha R}l_R - C_{\alpha F}l_F)]}{C_{\alpha F}C_{\alpha R}lv^2} \tag{6.38}$$

- *Domains* (4^{++}), (4^{+-}), (4^{-+}), (4^{--}). In this case, the equations of motion are

$$mv\dot{\beta} + mv^2\frac{\dot{\psi}}{v} = F_{ylF}\text{sign}(\alpha_F) + F_{ylR}\text{sign}(\alpha_R) \tag{6.39}$$

$$J_z\ddot{\psi} = F_{ylF}l_F\text{sign}(\alpha_F) - F_{ylR}l_R\text{sign}(\alpha_R) \tag{6.40}$$

This case is rather complicated to be discussed in detail, and in any case there are generally no stable equilibria.

The analysis of the non-linear system model behaviour is summarised in Table 6.2 for $(C_{\alpha R}l_R - C_{\alpha F}l_F) > 0$.

Table 6.1. Data of the reference vehicle taken under consideration

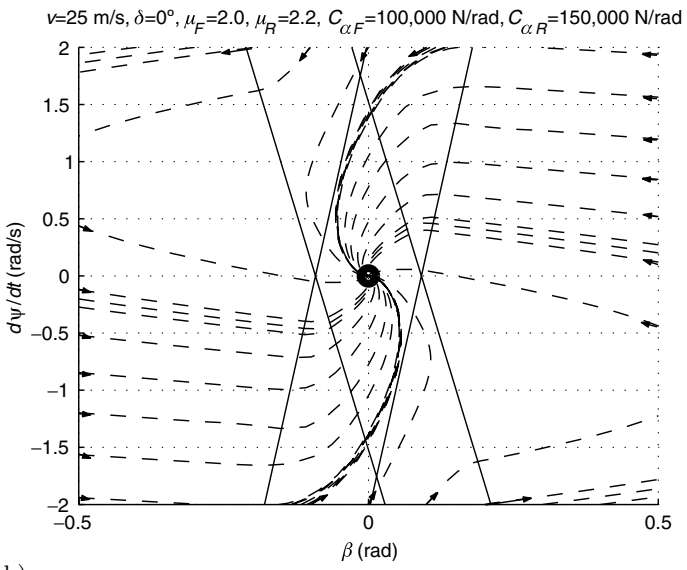
Design variable	Reference value	
m	1,100	(kg)
J_z	1,800	(kgm^2)
l_F	1.5	(m)
l_R	1.1	(m)
l	2.6	(m)
$C_{\alpha Fmin}$	50,000	(N/rad)
$C_{\alpha Fmax}$	200,000	(N/rad)
$C_{\alpha Rmin}$	50,000	(N/rad)
$C_{\alpha Rmax}$	300,000	(N/rad)

Figures 6.4 and 6.5 show typical phase plane plots for $\mu_F < \mu_R$ and $\mu_F > \mu_R$ respectively.

6.1.3 Objective Functions

The objective functions that will be considered for the optimisation of the handling behaviour of a vehicle are related to the response of the vehicle to a

a)



b)

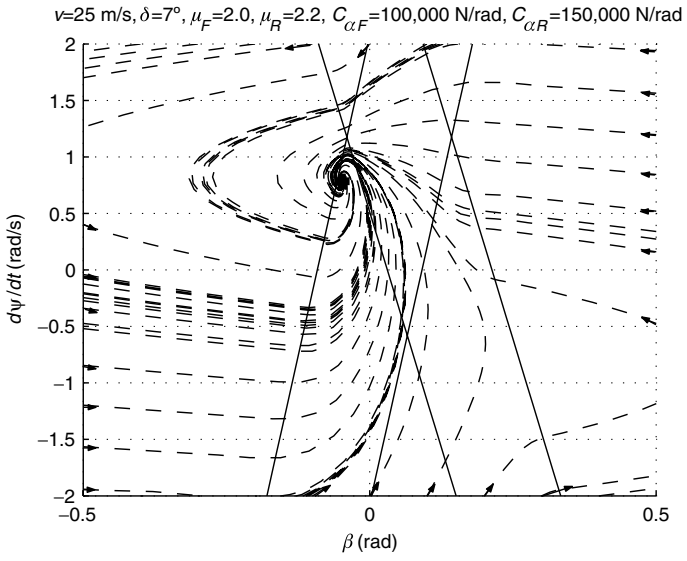
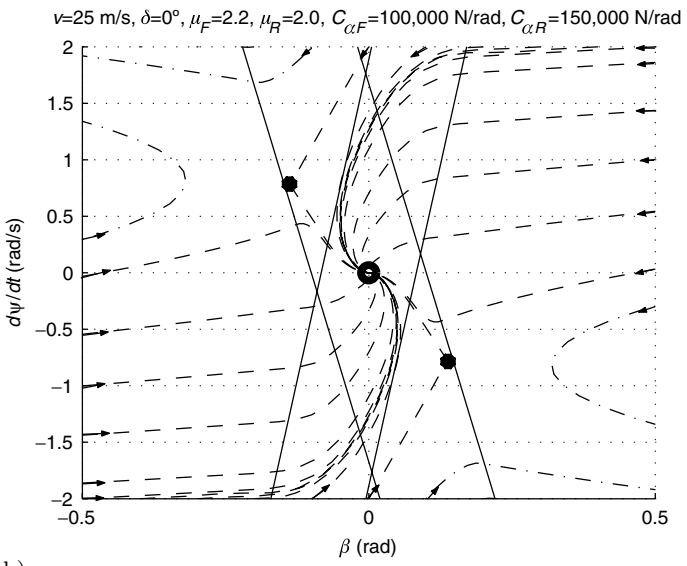


Fig. 6.4. Phase plane plot, $\mu_F < \mu_R$. Vehicle running at $v = 25$ m/s. (**a**) $\delta = 0° < \tilde{\delta}$ and (**b**) $\delta = 7° > \tilde{\delta}$ ($\tilde{\delta}$ is given by Eq. (6.28)). Stable equilibrium point '$\circ$' is given by Eqs. (6.9) and (6.10). Stable equilibrium point '+' is given by Eqs. (6.31) and (6.32). Vehicle data are reported in Table 6.1

a)



b)

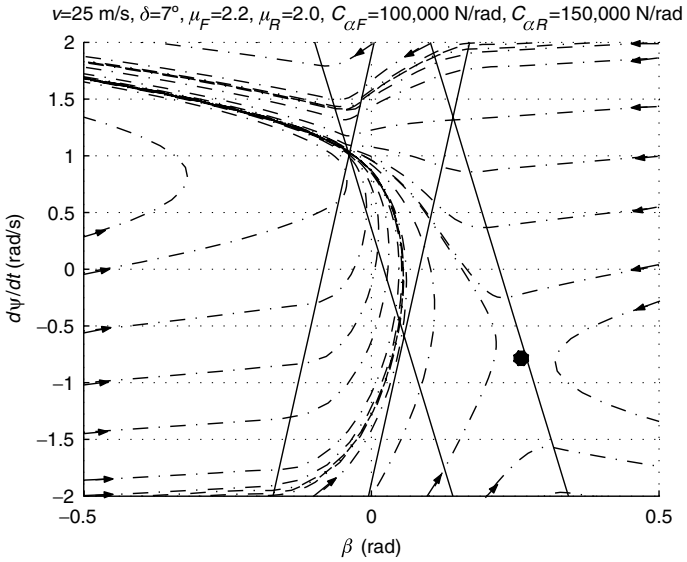


Fig. 6.5. Phase plane plot, $\mu_F > \mu_R$. Vehicle running at $v = 25\,\mathrm{m/s}$. (**a**) $\delta = 0° < \tilde{\delta}$ and (**b**) $\delta = 7° > \tilde{\delta}$ ($\tilde{\delta}$ is given by Eq. (6.28)) Stable equilibrium point '$\circ$' is given by Eqs. (6.9) and (6.10). Unstable equilibrium points '*' are given by Eqs. (6.36) and (6.37). Vehicle data are reported in Table 6.1

Table 6.2. Summary of the dynamic behaviour of the non-linear vehicle system model ($(C_{\alpha R}l_R - C_{\alpha F}l_F) > 0$, $\delta > 0$, $\tilde{\delta}$ is given by Eq. (6.28))

	$\delta < \tilde{\delta}$	$\delta > \tilde{\delta}$
$\mu_F < \mu_R$	1 stable equilibrium point in domain (1) (Eqs. (6.9), (6.10), Fig. 6.4 (a))	1 stable equilibrium point in domain (2^+) (Eqs. (6.31), (6.32) for $\alpha_F > 0$, Fig. 6.4 (b))
$\mu_F > \mu_R$	1 stable equilibrium point in domain 1 (Eqs. (6.9), (6.10), Fig. 6.5 (a)) 2 unstable equilibrium points in domains (3^+) and (3^-) (Eqs. (6.36), (6.37), Fig. 6.5 (a))	1 unstable equilibrium point in domain (3^-) (Eqs. (6.36), (6.37) for $\alpha_R < 0$, Fig. 6.5 (b))

steering step input ($\delta = \delta_{steady}\text{step}(t)$) performed at different vehicle speed. Similar objective functions are considered in [161, 167].

The steering step input manoeuvre refers to the ISO standard 7401 [245]. The considered objective functions are as follows:

- Steady-state sideslip angle at centre of gravity (gain) $K_\beta = \beta_{s\,s}/\delta_{steady}$; the steady-state sideslip angle at centre of gravity is strictly related to the driver feeling [167, 245]; drivers should prefer $\beta_{s\,s}$ close to zero. For the linear system model this objective function is given by

$$K_\beta^2 = \left(\frac{C_{\alpha F}(C_{\alpha R}l_R l - m l_F v^2)}{C_{\alpha F}C_{\alpha R}l^2 + m v^2 (C_{\alpha R}l_R - C_{\alpha F}l_F)} \right)^2 \tag{6.41}$$

- Peak response time of the yaw velocity $T_{\dot{\psi}}$ (Fig. 6.6), a typical yaw velocity response to a steering step input is shown; the peak response time ($T_{\dot{\psi}}$) is highlighted in Fig. 6.6, low values are considered good
- The overshoot of the yaw velocity $O_{\dot{\psi}}$, the expression of the yaw velocity overshoot for the linear system model (see Fig. 6.6) is

$$O_{\dot{\psi}} = \frac{\dot{\psi}_{max} - \dot{\psi}_{s\,s}}{\dot{\psi}_{s\,s}} \tag{6.42}$$

  The overshoot defines the precision of response of the vehicle, low values are considered good
- Yaw acceleration at $t = 0$ or $\ddot{\psi}(0)/\delta$; the initial yaw acceleration response to a steering step input has been considered because it is strictly correlated to the maximum yaw moment for this manoeuvre, low values are considered good

Deriving Eq. (6.23) with respect to time, the yaw acceleration can be obtained as follows:

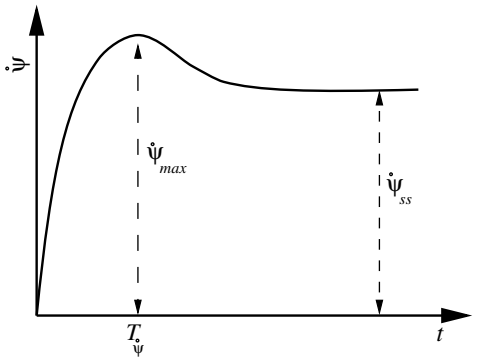


Fig. 6.6. Peak response time, $\dot{\psi}_{max}$ and $\dot{\psi}_{ss}$ definition with reference to the yaw velocity response

$$\frac{\ddot{\psi}(t)}{\delta_{steady}} = K_{\dot{\psi}} \left[s_1 \frac{s_1 + 2\varsigma - T_{z2}\nu^2}{-s_1 + s_2} e^{s_1 t} + s_2 \frac{-s_2 - 2\varsigma + T_{z2}\nu^2}{-s1 + s2} e^{s_2 t} \right] \tag{6.43}$$

for $t = 0$

$$\frac{\ddot{\psi}(0)}{\delta_{steady}} = K_{\dot{\psi}} T_{z2}\nu^2 = \frac{C_{\alpha F} l_F}{J_z} \tag{6.44}$$

Equation (6.23) can be used to obtain respectively the analytical formula of the peak response time and the overshoot of the yaw velocity ($T_{\dot{\psi}}$ and $O_{\dot{\psi}}$). There are different expressions of $T_{\dot{\psi}}$ and $O_{\dot{\psi}}$ depending on the cases considered in Sect. 6.1.1

Case 1. In this case the yaw velocity reaches the maximum value for a time equal to ∞. The peak response time ($T_{\dot{\psi}}$) is ∞ and the overshoot ($O_{\dot{\psi}}$) is 0.

Case 2. In this case the function $\dot{\psi}(t)$ (Eqs.(6.27)) is oscillating and it has finite relative maxima for

$$T_{\dot{\psi}} = \left(\arctan \frac{\sqrt{\nu^2 - \varsigma^2}\, T_{z2}}{\varsigma T_{z2} - 1} + k\pi \right) \frac{1}{\sqrt{\nu^2 - \varsigma^2}}, \qquad k = 0, 1, 2 \ldots \tag{6.45}$$

One may always verify that the absolute maximum occurs for the minimum positive $T_{\dot{\psi}}$. The overshoot value (Eqs. (6.27) and (6.42)) is equal to

$$O_{\dot{\psi}} = e^{-\varsigma T_{\dot{\psi}}} \sqrt{\nu^2 T_{z2}^2 - 2\varsigma T_{z2} + 1} \tag{6.46}$$

For high values of velocity ($v \to \infty$), the following approximation holds for $T_{\dot{\psi}}$ and $O_{\dot{\psi}}$

$$T_{\dot{\psi}} \approx \frac{\pi}{2}\sqrt{\frac{J_z}{C_{\alpha R}\, l_R - C_{\alpha F}\, l_F}} \tag{6.47}$$

$$O_{\dot{\psi}} \approx \frac{m l_F}{l C_{\alpha R}}\sqrt{\frac{C_{\alpha R} l_R - C_{\alpha F} l_F}{J_z}}\, v \tag{6.48}$$

Case 3. The vehicle is unstable because $\lim_{t\to\infty} \dot{\psi}(t) = \infty$ and this case is not to be considered for $(C_{\alpha R} l_R - C_{\alpha F} l_F) > 0$.

As indicated above β has to be preferably equal to zero, so the first objective function that is considered is $F_1 = K_\beta^2$. $F_2 = T_{\dot{\psi}}$, $F_3 = O_{\dot{\psi}}$ and $F_4 = \ddot{\psi}(0)/\delta_{steady}$ are the other three functions. The objective functions are defined in Table 6.3.

Table 6.3. Objective functions considered in the optimisation process

Objective functions			
		Linear vehicle model	Non-linear vehicle model
f_1	Sideslip angle gain (K_β^2)	Eq. (6.41)	Numerically computed
f_2	Yaw velocity peak response time ($T_{\dot{\psi}}$)	Eq. (6.45)	
f_3	Yaw velocity overshoot ($O_{\dot{\psi}}$)	Eq. (6.46)	
f_4	Initial yaw acceleration ($\ddot{\psi}(0)/\delta_{steady}$)	Eq. (6.44)	

The computation of the objective functions in the non-linear case is performed by integration of the equations of motion. Only the stable solutions having $\dot{\beta}$ and $\ddot{\psi}$ equal to zero, when $t \to +\infty$, are considered.

6.2 Results of the Optimisation

The mathematical procedure described in Sect. 3.6 has been used to optimise the handling performance of the system model in Fig. 6.1. The tyres cornering stiffness have been varied (see Sect. 6.1). The design variables to be optimised are the front and rear cornering stiffness at front axle $C_{\alpha F}$ and at rear axle $C_{\alpha R}$ respectively. The values of $C_{\alpha F}$ and $C_{\alpha R}$ are defined in the respective ranges

$$C_{\alpha F min} \le C_{\alpha F} \le C_{\alpha F max}$$
$$C_{\alpha R min} \le C_{\alpha R} \le C_{\alpha R max}$$

Additionally, the following condition has to be satisfied (see Sect. 6.1.1)

$$C_{\alpha R} l_R - C_{\alpha F} l_F > 0 \tag{6.49}$$

The objective functions to be optimised are given in Table 6.3.

6.2.1 Analytical Solution (Linear Case)

The optimisation problem can be solved analytically if a very high vehicle speed ($v \to \infty$) is considered. As $v \to \infty$ the objective functions read

$$\min \begin{cases} f_1 = & K_\beta^2 & = \left(\frac{C_{\alpha F} l_F}{C_{\alpha R} l_R - C_{\alpha F} l_F}\right)^2 \\ f_2 = & T_{\dot\psi} & = \frac{\pi}{2}\sqrt{\frac{J_z}{C_{\alpha R}\, l_R - C_{\alpha F}\, l_F}} \\ f_3 = & O_{\dot\psi} & = \frac{m l_F}{l C_{\alpha R}}\sqrt{\frac{C_{\alpha R} l_R - C_{\alpha F} l_F}{J_z}}\, v \\ f_4 = \ddot\psi(0)/\delta_{steady} & & = C_{\alpha F} l_F / J_z \end{cases} \tag{6.50}$$

If a bounded domain of the design variables is considered $[C_{min}, C_{max}]$, the absolute minimum of each objective function is obtained when $C_{\alpha F}$ and $C_{\alpha R}$ get the values reported in Table 6.4.

Figure 6.7 shows the Pareto-optimal set (denoted by symbol '+') in the design variable space for the case $v \to \infty$. In Fig. 6.7, the curve γ reads

$$C_{\alpha R} = \frac{C_{\alpha F} l_F C_{\alpha R max}}{(C_{\alpha R max} l_R - C_{\alpha F} l_F)} \tag{6.51}$$

Equation (6.51) (γ–curve) is found by applying the constraints method (see Sect. 3.4.10) and solving the optimisation problem analytically as follows.

Table 6.4. Values of the cornering stiffness defining each single absolute minimum of the four objective functions that have been taken into consideration for the optimisation of the handling behaviour of a road vehicle

Objective function		**Optimal cornering stiffness values**	
		$C_{\alpha F}$	$C_{\alpha R}$
f_1	K_β^2	$C_{\alpha F min}$	$C_{\alpha R max}$
f_2	$T_{\dot\psi}$	$C_{\alpha F min}$	$C_{\alpha R max}$
f_3	$O_{\dot\psi}$	$\frac{l_R}{l_F} C_{\alpha R}$	$\frac{l_F}{l_R} C_{\alpha F}$
f_4	$\ddot\psi(0)/\delta$	$C_{\alpha F min}$	whichever $\in [C_{\alpha R min}, C_{\alpha R max}]$

Objective function		**Objective function minimum values**	
		$v =$ **low, medium**	$v =$ **high** ($v \to \infty$)
f_1	K_β^2	$f_1(C_{\alpha F min}, C_{\alpha R max})$	$\left(\frac{C_{\alpha F min} l_F}{C_{\alpha R max} l_R - C_{\alpha F min} l_F}\right)^2$
f_2	$T_{\dot\psi}$	$f_2(C_{\alpha F min}, C_{\alpha R max})$	$\frac{\pi}{2}\sqrt{\frac{J_z}{C_{\alpha R max}\, l_R - C_{\alpha F min}\, l_F}}$
f_3	$O_{\dot\psi}$	0	0
f_4	$\ddot\psi(0)/\delta$	$f_4(C_{\alpha F min})$	$\frac{C_{\alpha F min} l_F}{J_z}$

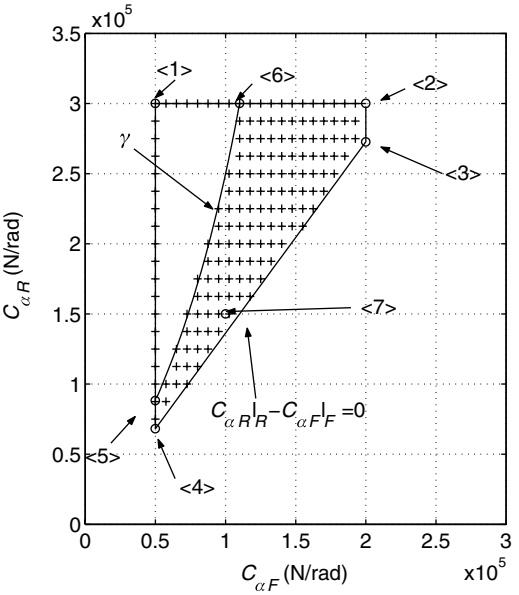


Fig. 6.7. Pareto-optimal set plotted in the design variable space. Linear vehicle model. The Pareto-optimal set is defined by symbol '+'. Boundaries have been computed analytically. Vehicle data are given in Table 6.1. γ is defined by Eq. (6.51). The time responses for cases $< 1 > - < 7 >$ are plotted in Fig. 6.9

The γ–curve can be found relatively easily if the objective functions f_i $(i = 1, \ldots, 4)$ are re-written, respectively, as the following new objective functions f_i', $(i = 1, \ldots 4)$[1]

$$f_1' = \left(\frac{C_{\alpha F}}{C_{\alpha R} l_R - C_{\alpha F} l_F} \right)^2$$

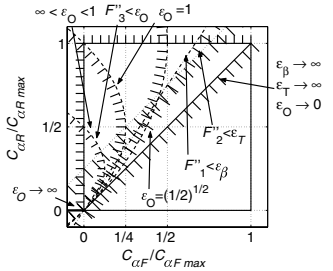
$$f_2' = \frac{1}{\sqrt{C_{\alpha R} l_R - C_{\alpha F} l_F}}$$

$$f_3' = \frac{\sqrt{C_{\alpha R} l_R - C_{\alpha F} l_F}}{C_{\alpha R}}$$

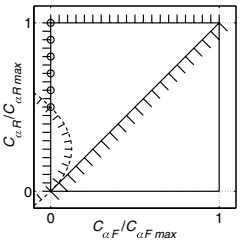
$$f_4' = C_{\alpha F}$$

In order to give a simple but general formulation of the optimisation problem, the design variables $C_{\alpha F}$ and $C_{\alpha R}$ can be set in non-dimensional form as $x_1 = C_{\alpha F}/C_{\alpha F max}$, $x_2 = C_{\alpha R}/C_{\alpha R max}$, additionally, without introducing any important restriction to the optimisation problem, we can set $C_{\alpha F min} = 0$, $C_{\alpha R min} = 0$ and $C_{\alpha R max} l_R = C_{\alpha F max} l_F$.

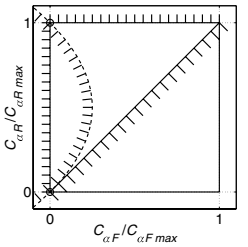
[1] Actually a set of non-dominated (i.e. optimal) solutions (see Sect. 2.11) remains non-dominated by multiplying each objective function by a scale factor (different scale factors for different objective functions are allowed)



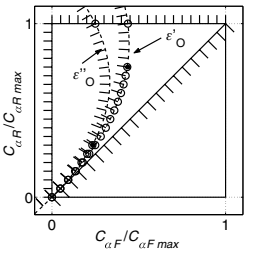
a)



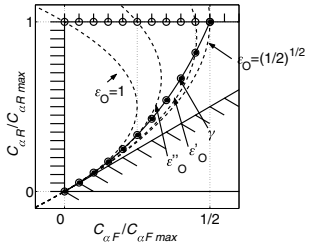
b) $0 < \varepsilon_\beta < \infty$, $0 < \varepsilon_T < \infty$, $1 < \varepsilon_O < \infty$



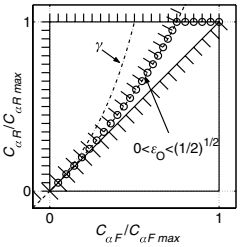
c) $0 < \varepsilon_\beta < \infty$, $0 < \varepsilon_T < \infty$, $\varepsilon_O = 1$



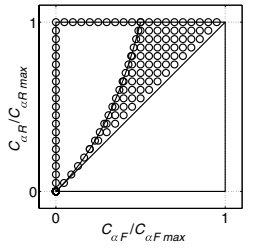
d) $0 < \varepsilon_\beta < \infty$, $0 < \varepsilon_T < \infty$, $(1/2)^{1/2} < \varepsilon'_O < 1$, $(1/2)^{1/2} < \varepsilon''_O < 1$, $\varepsilon'_O < \varepsilon''_O$



e) $0 < \varepsilon_\beta < \infty$, $0 < \varepsilon_T < \infty$, $(1/2)^{1/2} < \varepsilon_O < 1$



f) $0 < \varepsilon_\beta < \infty$, $0 < \varepsilon_T < \infty$, $0 < \varepsilon_O < (1/2)^{1/2}$



g) $0 < \varepsilon_\beta < \infty$, $0 < \varepsilon_T < \infty$, $0 < \varepsilon_O < \infty$

Fig. 6.8. Solution of the problem (6.52) '∘' for different values of the parameters ε_O, ε_β and ε_T. Different values of these parameters determine the shape of the feasible domain of the problem (6.52). The union of points '∘' obtained for the problem (6.52) solved for different values $(\varepsilon_O, \varepsilon_\beta, \varepsilon_T)$ allows to obtain the Pareto-optimal set of the problem considered

The objective functions f_i' $(i = 1, \ldots, 4)$ can be re-written (once again) in non-dimensional form as

$$
\begin{aligned}
f_1'' &= \left(\frac{C_{\alpha F}/C_{\alpha F max}}{(C_{\alpha R} l_R - C_{\alpha F} l_F)/(C_{\alpha R max} l_R)} \right)^2 \\
f_2'' &= \frac{1}{\sqrt{(C_{\alpha R} l_R - C_{\alpha F} l_F)/(C_{\alpha R max} l_R)}} \\
f_3'' &= \frac{\sqrt{(C_{\alpha R} l_R - C_{\alpha F} l_F)/(C_{\alpha R max} l_R)}}{(C_{\alpha R})/(C_{\alpha R max})} \\
f_4'' &= C_{\alpha F}/C_{\alpha F max}
\end{aligned}
$$

and substituting the non-dimensional design variables x_1, x_2

$$
\begin{aligned}
f_1'' &= \left(\frac{x_1}{x_2 - x_1} \right)^2 \\
f_2'' &= \frac{1}{\sqrt{x_2 - x_1}} \\
f_3'' &= \frac{\sqrt{x_2 - x_1}}{x_2} \\
f_4'' &= x_1
\end{aligned}
$$

To solve analytically the addressed optimisation problem, the constraints method will be applied (see Sect. 3.7.1), so

$$
\begin{aligned}
&\min_{x_1, x_2} f_4''(x_1) \\
&f_1''(x_1, x_2) \leq \varepsilon_\beta \\
&f_2''(x_1, x_2) \leq \varepsilon_T \\
&f_3''(x_1, x_2) \leq \varepsilon_O \\
&0 \leq x_1 \leq 1 \\
&0 \leq x_2 \leq 1 \\
&x_1 - x_2 < 0
\end{aligned}
\tag{6.52}
$$

In explicit form, the above mathematical problem reads

$$
\begin{aligned}
&\min_{x_1, x_2} x_1 \\
&x_2 \geq x_1(\sqrt{\varepsilon_\beta} + 1)/\sqrt{\varepsilon_\beta} \\
&x_2 \geq x_1 + 1/\varepsilon_T^2 \\
&\varepsilon_O^2 x_2^2 - x_2 + x_1 \geq 0 \\
&0 \leq x_1 \leq 1 \\
&0 \leq x_2 \leq 1 \\
&x_1 - x_2 < 0
\end{aligned}
\tag{6.53}
$$

It can be noticed that the last three inequalities in (6.53), shown in Fig. 6.8(a), are given as the area shaped as a triangle whose edges are defined by points (0,0), (0,1), (1,1). The points outside the triangle correspond to solutions that have to be discarded. The optimal solutions have to be searched into the triangle. The optimal solutions area is defined by applying the first three inequalities in (6.53)). Let us assume (Fig. 6.8(a)) that the boundaries defined by the first two inequalities ($f_1'' \leq \varepsilon_\beta$ and $f_2'' \leq \varepsilon_T$) can be represented as lines (Fig. 6.8), whereas the boundary given by the third inequality ($f_3'' \leq \varepsilon_O$) is a parabola whose axis is directed towards the x_1–axis. For all the values of ε_O, the point $x_1 = 0, x_2 = 0$ belongs to the given parabola and in this point it is tangent to the line $x_2 = x_1$.

The first three inequalities in (6.53) depend on $\varepsilon_\beta, \varepsilon_T$ and ε_O. The Pareto-optimal solutions (points) in the design variable space are found, (according to the constraints method), by varying the constraints $\varepsilon_\beta, \varepsilon_T$ and ε_O within their domain and then searching for the minimum x_1. The domain of the constraints is $0 < \varepsilon_\beta < \infty$, $0 < \varepsilon_T < \infty$ and $0 < \varepsilon_O < \infty$. The tics marks in Fig. 6.8(a) define the borders of the excluded area in which optimal solutions are not to be searched for.

For ease of presentation, and according to the constraints method, at first ε_O will be varied from ∞ to zero, subsequently, each value of ε_O, ε_β and ε_T will be varied within their respective ranges.

- Let us consider $1 < \varepsilon_O < \infty$. Figure 6.8(a) shows that as $\varepsilon_O \to \infty$ the parabola is shrunk to a line corresponding to the negative x_1–axis. For $\varepsilon_O = 1$ the point (0,1) belongs to the parabola. Let us consider Fig. 6.8(b) in which a generic parabola ($1 < \varepsilon_O < \infty$) is plotted. Let us vary ε_β in the range $0 < \varepsilon_\beta < \infty$; this defines a set of lines as those plotted in Fig. 6.8(a). Each couple of constraints $\varepsilon_O, \varepsilon_\beta$ define an area in which the point with the minimum x_1 is always found at $x_2 = 0$. The variation of ε_T in the range $0 < \varepsilon_T < \infty$ produces the same effect. For the case considered in Fig. 6.8(b), the Pareto-optimal points are those indicated by the symbols 'o'.
  For $1 < \varepsilon_O < \infty$, the Pareto-optimal set is the x_2-axis (from $x_2 = 0$ to $x_2 = 1$).
- Let us consider $\varepsilon_O = 1$ (Fig. 6.8(c)). The points $x_1 = 0$, $x_2 = 0$, $x_1 = 0$, $x_2 = 1$ belong to the parabola and in this case they represent the domain of optimal x_1, x_2.
- Let us consider $\sqrt{2}/2 \leq \varepsilon_O < 1$ (Fig. 6.8(d)). The domain containing optimal x_1, x_2 is defined again by the points denoted by the symbol 'o'. The two parabolas in Fig. 6.8(d) have been plotted in order to show how the optimal points vary as ε_O varies. The points denoted by black dots in Fig. 6.8(d) can be found as follows. Given the abscissa x_1, the two x_2 values of the parabola $\varepsilon_O^2 x_2^2 - x_2 + x_1 = 0$ are $x_2 = (1 \pm \sqrt{1 - 4\varepsilon_O^2 x_1})/(2\varepsilon_O^2)$, where the minus sign denotes the black dot. Setting $x_2 = 1 = (1 + \sqrt{1 - 4\varepsilon_O^2 x_1})/(2\varepsilon_O^2)$ and solving this equation

with respect to ε_O^2 gives $\varepsilon_O^2 = 1 - x_1$, which represent the value of ε_O when the point $x_2 = 1$ belongs to the parabola.

The value of the ordinate of the black dot is found by substituting $\varepsilon_O^2 = 1 - x_1$ in $x_2 = (1 - \sqrt{1 - 4\varepsilon_O^2 x_1})/(2\varepsilon_O^2)$ which gives

$$x_2 = \frac{x_1}{1 - x_1} \tag{6.54}$$

This relationship represents the locus of points denoted by black dots in Fig. 6.8(e). This curve (named γ–curve in (6.51)) defines the set of optimal solutions when $\sqrt{2}/2 \leq \varepsilon_O \leq 1$. The points of the line $x_2 = 1, 0 \leq x_1 \leq 1/2$ do belong to the Pareto-optimal set; in fact, in Fig. 6.8(d), the minimum x_1 is obtained at the black dots and at the corresponding points having the same abscissa but ordinate $x_2 = 1$. By introducing the design variables in dimensional form in (6.54), the expression of the γ–curve given by (6.51) can be obtained.

- Let us consider $0 < \varepsilon_O < \sqrt{2}/2$. The Pareto-optimal set is denoted by symbols 'o' in Fig. 6.8(f).
- In Fig. 6.8(g) the whole set of Pareto-optimal solutions is presented, the main hypotheses are that the considered model is linear and its speed is high.

Removing the hypotheses $C_{\alpha F min} = 0$, $C_{\alpha R min} = 0$ and $C_{\alpha R max} l_R = C_{\alpha F max} l_F$ does not modify the solution of the problem. In particular, the γ–curve do maintain the same analytical form (see Eq. (6.51)). The γ–curve bounds the domain in Fig. 6.7 and establishes a relationship between $C_{\alpha F}$ and $C_{\alpha R}$.

The Pareto-optimal set in Fig. 6.7 is multiply connected. Figure 6.9 shows the different responses to a step steering input of the seven vehicle models whose front and rear cornering stiffness are defined with the numbers <1> to <7> in Fig. 6.7. Figure 6.10 shows the trends of each objective function, the tradeoffs among objective functions can be assessed by inspection and comparison of the four plots.

6.2.2 Numerical Solutions (Linear Case)

If the vehicle speed is not very high, the Pareto-optimal set has to be computed numerically, and obviously, the Pareto-optimal set changes as the vehicle speed varies. The numerical solution is obtained by using the method described in Sect. 3.6.

Different Pareto-optimal sets at different vehicle speeds are shown in Fig. 6.11. The Pareto-optimal set is influenced only by the vehicle forward velocity and not by the steering angle δ. As the speed increases, the Pareto-optimal sets in Fig. 6.11 approach the shape of the Pareto-optimal set in Fig. 6.7. So the analytical solution (γ–curve) can be considered as a reference solution, in particular if the response at high speed is of interest. For the reference vehicle, the Pareto-optimal set at $v = 35$ m/s is similar to the one obtained at $v \to \infty$.

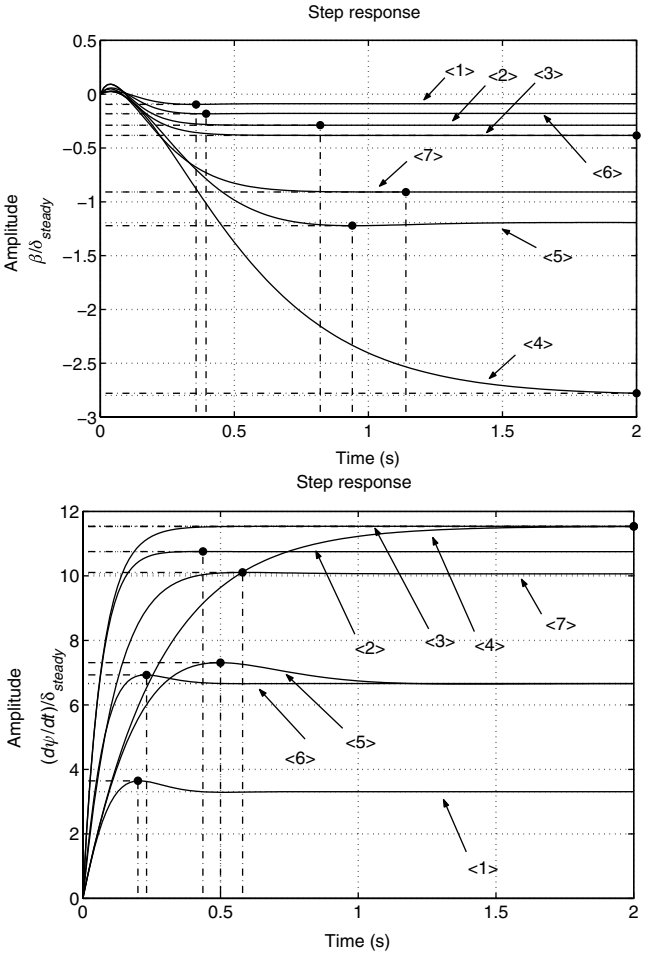


Fig. 6.9. Time responses to a step steering angle input of cars (design solutions) denoted by numbers $< 1 >$ to $< 7 >$ in Fig. 6.7. The symbol $\bullet$ indicates the peak response time. The forward speed considered is $v = 30$ m/s. The solution $< 1 >$ has the best $T_{\dot{\psi}}$ and K_{β}^2 (see Table 6.4, $\min F_1, F_2$), the solutions $< 3 >$ and $< 4 >$ have the best $O_{\dot{\psi}}$ (see Table 6.4, $\min F_3$) and the solutions $< 1 >$, $< 5 >$ and $< 4 >$ have the best $\ddot{\psi}/\delta_{steady}(0)$ (see Table 6.4, $\min F_4$)

6.2.3 Numerical Solution (Non-Linear Case)

If the non-linear model is considered, the effect of the steering angle on the Pareto-optimal set becomes important. Only the solutions for which the vehicle motion is stable will be considered ($\lim_{t\to\infty} \dot{\psi} \neq \infty$ and $\lim_{t\to\infty} \beta \neq \infty$).

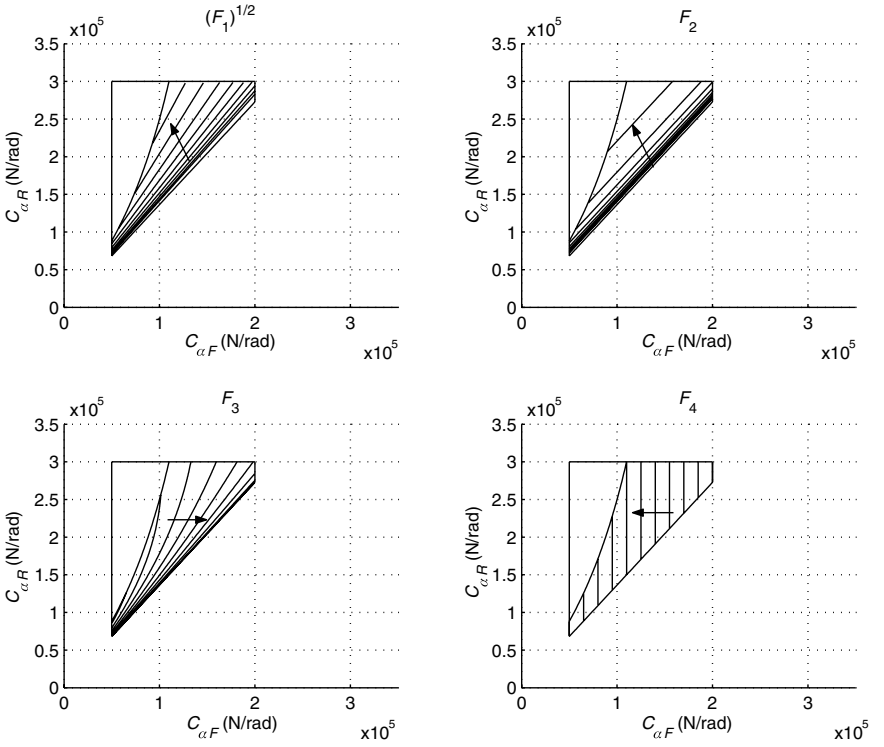


Fig. 6.10. Pareto-optimal set plotted in the design variable space for the linear vehicle model as in Fig. 6.7. The arrows indicate decreasing values of each objective function, see Table 6.3. Vehicle data are given in Table 6.1

The results of the simulations are shown in Figs. 6.12–6.15. These plots show how the Pareto-optimal set can be influenced by the steering angles and by the values of μ_F and μ_R.

If $\mu_R < \mu_F$ (Fig. 6.12), the area of Pareto-optimal set in design variable space is shrunk because many design variables combination define an unstable motion of the system model. In this case for high steering angles and high speed ($\delta > \tilde{\delta}$ is given by Eq. (6.38)) the motion of the model is unstable even if the condition $(C_{\alpha R} l_R - C_{\alpha F} l_F) > 0$ is satisfied. If $\mu_R < \mu_F$ cars are commonly considered dangerous and the results presented in Fig. 6.12 have only academic interest.

If $\mu_R > \mu_F$ (Figs. 6.13–6.15), all the design variables combinations that satisfy the condition $(C_{\alpha R} l_R - C_{\alpha F} l_F) > 0$ refer to a stable motion of the system model. For some values of the design variables, and for high speed and high steering angles (where $\delta > \tilde{\delta}$ is given by Eq. (6.28)) the dynamic behaviour of the vehicle changes with respect to the linear model. Figures 6.13–6.15 show the shape of the Pareto-optimal set together with the trends

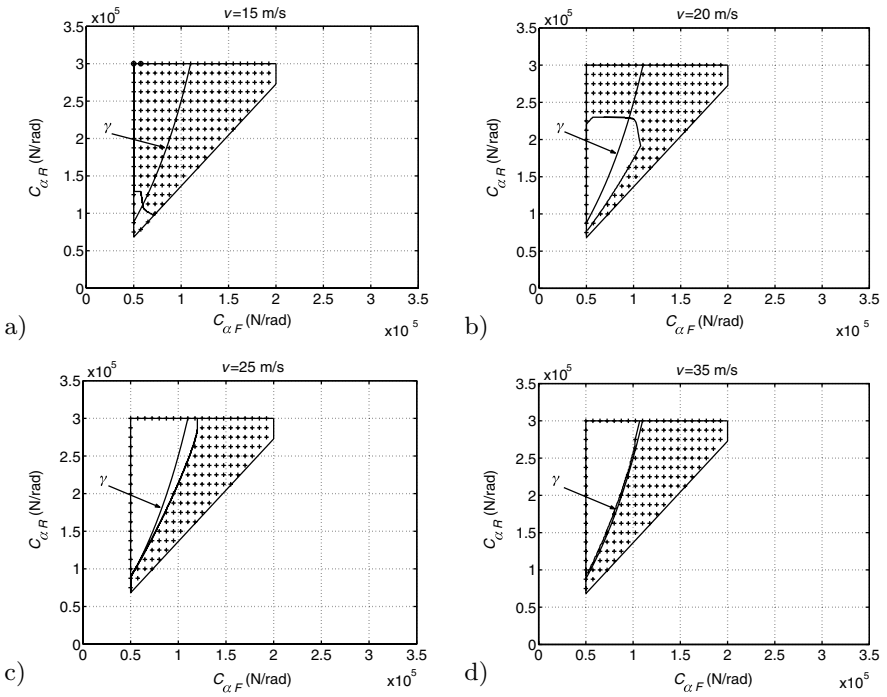


Fig. 6.11. Influence on the Pareto-optimal set of the vehicle speed. Linear vehicle model as in Fig. 6.7. (**a**) $v = 15\,\mathrm{m/s}$, (**b**) $v = 20\,\mathrm{m/s}$ (**c**) $v = 25\,\mathrm{m/s}$ and (**d**) $v = 35\,\mathrm{m/s}$. The Pareto-optimal set is defined by symbols '+'. Vehicle data are given in Table 6.1

of the objective functions. The change in the shape of the Pareto-optimal solution set is due to change in monotonicity of the objective functions, as highlighted by the trends of the objective functions shown in Figs. 6.13–6.15. The Pareto-optimal set in these cases is not connected because the design solutions that are stationary may become globally or locally Pareto-optimal.

6.3 Validation

6.3.1 Validation of the Model

The handling performance of an actual vehicle depends on many parameters (e.g. 100 and more). So only a complex vehicle system model can be used to assess accurately the handling behaviour. However, it is well known that even a 2 d.o.f. linear model such as the one shown in Fig. 6.1 can represent fairly well the handling behaviour of a road vehicle at relatively low lateral acceleration levels [167]. Thus, the fundamental study presented in this chapter is based on a system model of a road vehicle which represents a very good compromise

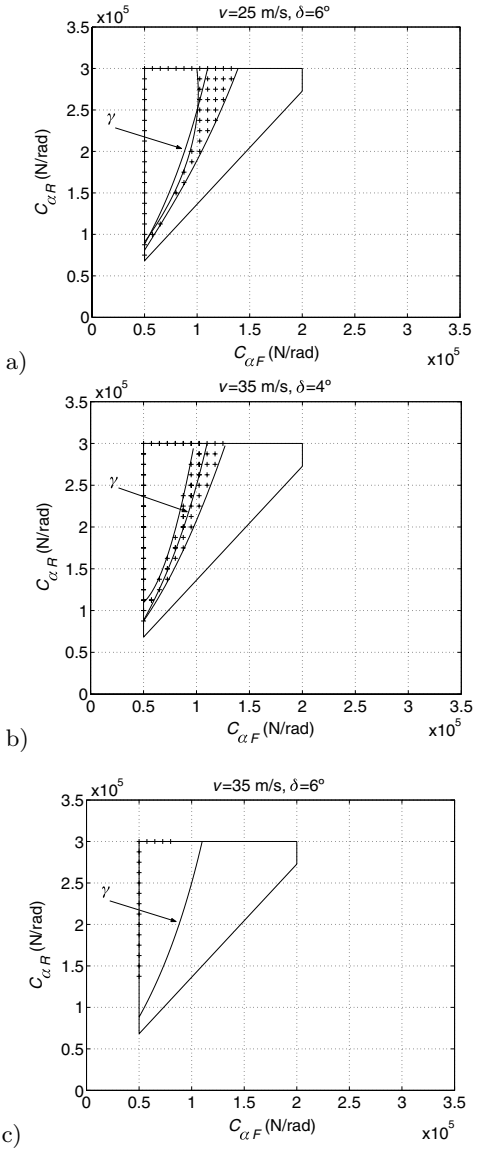


Fig. 6.12. Pareto-optimal set plotted in the design variable space. non-linear vehicle model: $\mu_R = 2.0$ and $\mu_F = 2.2$. Influence of speed and of steering input angles: (**a**) $v = 25\,\mathrm{m/s}$, $\delta = 6^\circ$, (**b**) $v = 35\,\mathrm{m/s}$, $\delta = 4^\circ$ and (**c**) $v = 35\,\mathrm{m/s}$ and $\delta = 6^\circ$. The Pareto-optimal set is defined by symbols '+'. Vehicle data are given in Table 6.1

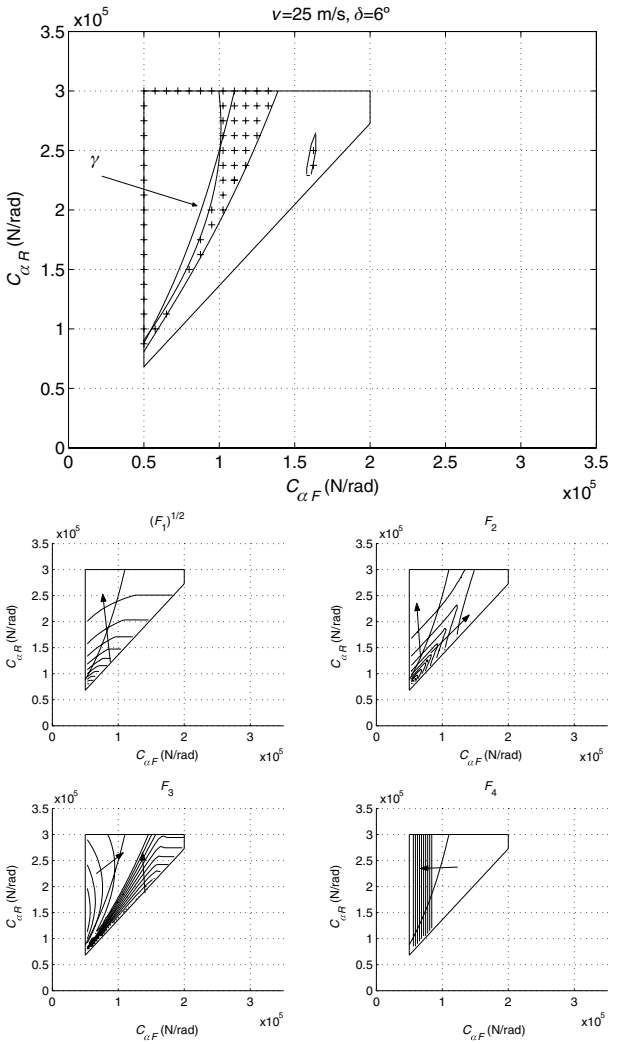


Fig. 6.13. Pareto-optimal set plotted in the design variable space. non-linear vehicle model, $\mu_R = 2.2$ and $\mu_F = 2.0$ ($v = 25\,\mathrm{m/s}$ and $\delta = 6^\circ$). The Pareto-optimal set is defined by symbols '+'. *Bottom*: Trends of the objective functions (the *arrows* show decreasing directions of the objective functions). Vehicle data are given in Table 6.1

between accuracy of numerical computations and simplicity of mathematical formulation. In other words the adopted models allow to capture the essential information on the actual dynamic behaviour of a road vehicle.

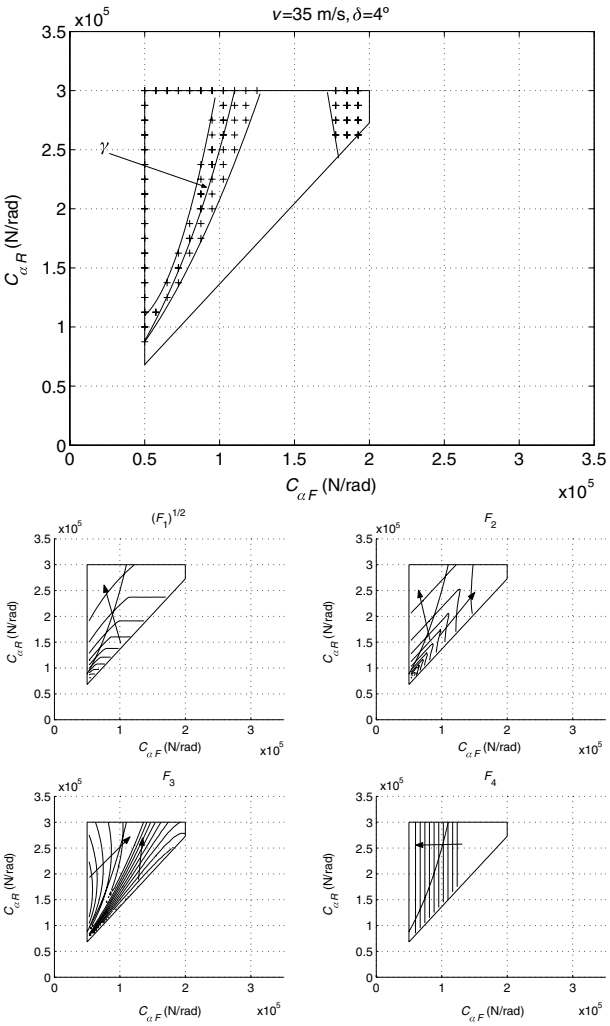


Fig. 6.14. Pareto-optimal set plotted in the design variable space. non-linear vehicle model, $\mu_R = 2.2$ and $\mu_F = 2.0$. ($v = 35\,\mathrm{m/s}$ and $\delta = 4°$). The Pareto-optimal set is defined by symbols '+'. *Bottom:* Trends of the objective functions (the arrows show decreasing directions of the objective functions). Vehicle data are given in Table 6.1

6.3.2 Validation of the Optimisation Process

The validation of an optimisation process such as that one presented here is rather involved. Actually, in principle, one should verify that the whole set of Pareto-optimal solutions are truly optimal, which would require the testing on a proving ground of a huge number of vehicles. A different approach has been used here to establish a relationship between the results of the presented optimisation process and actual applications. A number of cars (sports, sedan,

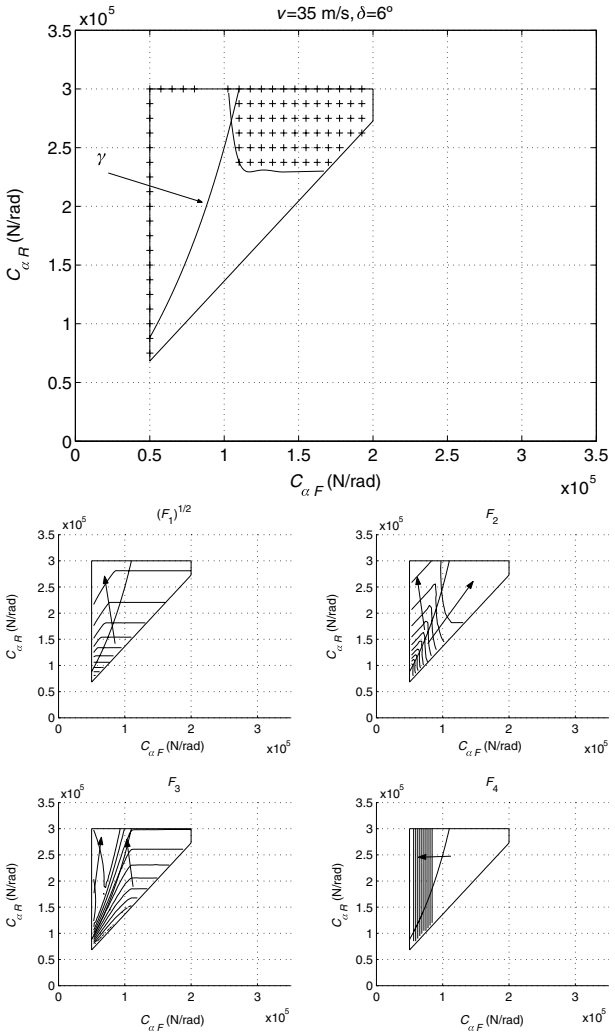


Fig. 6.15. (*Left*) Pareto-optimal set plotted in the design variable space. non-linear vehicle model, $\mu_R = 2.2$ and $\mu_F = 2.0$ ($v = 35\,\mathrm{m/s}$ and $\delta = 6^\circ$). The Pareto-optimal set is defined by symbols '+'. (*Bottom*) Trends of the objective functions (the *arrows* show decreasing directions of the objective functions). Vehicle data are given in Table 6.1

suvs, ...) under present or former production have been considered. The tyre cornering stiffness of these cars have been plotted in Fig. 6.16 and are reported in Table 6.5.

The graph shows the Pareto-optimal set as a function of $C_{\alpha F} l_F / l$ and $C_{\alpha R} l_R / l$ to allow the comparison of vehicles of different size and mass distribution. All of the points in Fig. 6.16 referring to the considered vehicles lie

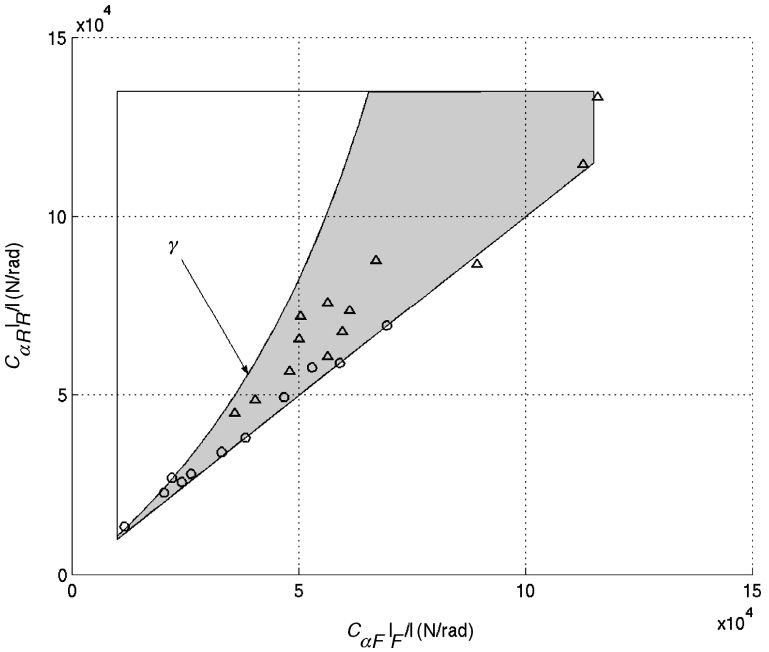


Fig. 6.16. Pareto-optimal set (*grey area*) in the design variable space as a function of $C_{\alpha F}l_F/l$ and $C_{\alpha R}l_R/l$. Data denoted by '$\circ$' were derived from the literature [46, 268, 269, 271], data denoted by '$\triangle$'were available to the authors and measured by a number of tyre manufacturers

in the optimal domain (linear vehicle, Fig. 6.7). Of course, it is difficult to state, on the basis of this matter of fact, that the presented method allows to tune the tyre stiffness of actual cars, however it is also true that, referring to the significant sample taken into consideration, no cars have cornering stiffness that do not belong to the derived Pareto-optimal domain. Thus, one may conclude that the presented optimisation method can help in avoiding to analyse the dynamic behaviour of cars whose cornering stiffnesses do not belong to the mentioned Pareto-optimal domain.

The optimal cornering stiffness values found by this method can be used as a first approximation, useful for starting further researches and/or understanding the results found after ground tests.

6.4 Conclusion

The chapter presents a fundamental study on the optimisation of the handling performances of road vehicles. The aim was that of giving, with reference to a simple, but relatively accurate mathematical model, a scientific contribution

Table 6.5. Vehicle data in Fig. 6.16

	m (kg)	l (m)	l_R/l–l_F/l (%)	$C_{\alpha F}l_F/l$ (N/rad)	$C_{\alpha R}l_R/l$ (N/rad)
Saloon [46]	727	2.04	62–38	11,550	13,500
Saloon [46]	1045	2.40	57–43	24,200	25,900
Coupé [46]	1175	2.58	55–45	20,250	22,900
Saloon [46]	1435	2.77	53–47	32,950	34,150
Saloon [46]	1945	3.07	49–51	46,700	49,450
Sport [46]	1600	2.50	50–50	38,200	38,200
Formula 1 [46]	625	2.72	40–60	59,000	59,000
Sport Prototype [46]	1020	2.91	41–59	69,350	69,350
Saloon [271]	1480	2.68	61–39	52,900	57,800
Saloon [268]	1600	3.2	56–44	26,250	28,100
Saloon [269]	1067	2.5	60–40	22,000	27,000
Saloon	1096	2.46	61–39	35,900	45,000
Saloon	1096	2.46	61–39	50,350	72,100
Saloon	1404	2.60	61–39	56,350	76,400
Saloon	1404	2.60	61–39	50,050	65,750
Saloon Station Wagon	1461	2.59	59–41	47,900	56,800
Saloon Station Wagon	1461	2.59	59–41	56,400	60,850
Saloon	1667	2.70	59–41	59,500	67,700
Saloon	1554	2.70	61–39	40,350	48,850
Saloon	1890	2.80	57–43	61,200	73,650
Sport	1820	2.66	54–46	112,550	114,500
Sport	1628	2.45	42–58	89,200	86,700
Sport	1620	2.60	44–56	66,950	87,850
Racing	1100	2.60	42–58	115,700	133,300

on how to tune the front and rear tyre cornering stiffness of a road vehicle, in order to obtain the best handling and active safety performances.

The system model that has been adopted is the well-known 2 d.o.f. model of a two axle road vehicle cornering on a flat horizontal surface.

The tyre characteristics have been introduced in two different ways: at first a linear relationship between lateral slip and lateral force at a tyre was introduced, then, to account for non-linearities, a piecewise linear characteristic has been used. By means of these two mathematical models of a road vehicle the response to a step steering input has been computed analytically whenever possible.

In particular, the relevant objective functions for the handling optimisation process were derived in analytical form by exploiting the linear model. The handling objective functions taken into consideration referred to a step steering manoeuvre are the sideslip angle gain, the yaw velocity peak response time, the yaw velocity overshoot, and the initial yaw acceleration.

The selected handling objective functions are conflicting, i.e. improving one implies the worsening of another. By using both the linear and the non-linear road vehicle models, the optimal compromise among the selected handling objective functions has been searched for. The front and the rear tyre cornering stiffness have been considered as the design variables to be optimised.

The optimisation has been performed on the basis of multi-objective programming. The values of the tyre cornering stiffness which minimise each single objective function have been computed in analytical form for the linear model of a vehicle running at high speed.

Together with numerical computations, an analytical formulation of the boundary of the Pareto-optimal set in the design variable space (γ–curve) has been derived.

The cornering stiffness computed by the optimisation process have been compared to those ones pertaining to 24 cars under present or former production. All of the values were inside the Pareto-optimal domain whose boundaries were computed analytically. This means that the presented optimal theory allows at least to discard the non-optimal values of the front and rear cornering stiffness of a road vehicle. Additionally, the boundaries of the Pareto set in the design variable space do have a simple analytical form and this allows a simple reasoning during engineering activities. Of course, the presented problem solution cannot allow an accurate tuning of tyre cornering stiffnesses of a road vehicle, anyway it has been shown that, referring to a significant set of actual production cars, the tyre cornering stiffnesses that have been definitely adopted in the engineering practice do belong to the Pareto-optimal set computed in the optimisation process proposed in this chapter.

7

Optimal Design of the Tyre-Suspension System of a Racing Car

A racing car should be designed to improve the performance of the driver–vehicle system. Unfortunately, at present no readily exploitable driver models seem to be available. So the optimisation of a racing car refers mainly to the car itself. Contemporary tuning of both tyre characteristics and suspesnion system is of crucial importance. One of the main technical challenges is to achieve maximum cornering speed while balancing tyre forces at front and rear axles, improving handling ability.

In order to design effectively a racing car, one may be requested to tune the entire set of design variables related to tyres, aerodynamics and chassis characteristic (stiffness, damping and kinematics of the suspension system). This is not a straightforward problem especially if one exploits very sophisticated models defined by many parameters. In fact, when many design variables have to be changed, their tuning has to be performed by using adequate optimisation theories (see Sect. 2.10) and related software tools. Traditionally, the process of tuning the car has been performed by a trial-and-error approach based on professional drivers' feedback. A sensitivity analysis [32] is of limited help to the designer if the number of design variables is high.

A more rational approach compared to a trial-and-error procedure is desirable. Multi-objective optimisation is the most effective way for optimising complex systems, i.e. systems, which are described by many design variables and which perform many functions [149] (see Chap. 3).

In the literature, several papers [93,94,141] have dealt with the problem of the optimisation of a car suspension system by means of the multi-objective approach.

This chapter, derived from [161], presents a procedure that can be used for the integrated design (tuning) of tyres and suspensions of a racing car.

A complete model of a racing car has been developed and implemented through a series of programming codes. Elasto-kinematics of the suspension system, powertrain and braking systems have been described in detail. Tyre characteristics have been identified and implemented by means of

Pacejka's Magic Formulæ. The aerodynamic characteristics of the body have been measured in wind tunnel and included into the model.

A number of ground tests, according to the ISO standards, have been performed to validate the vehicle model. The main dynamic responses of the car have been recorded and compared with the computed responses, with fully satisfactory results.

Proper objective functions have been defined after a subjective–objective correlation analysis. Many different driving situations (steady-state, J-turn, lane-change, power on–off while steering, braking on a bend, passing over a kerb while steering) have been considered. By means of a multi-objective (Sect. 2.10) approach, 26 objective functions, defined after a subjective–objective correlation analysis, have been optimised by changing 18 design variables related with the suspension system and the tyre characteristics. A global approximation model (see Sect. 4) has been followed (the original physical car model has been substituted by another purely numerical model), allowing fast optimisations.

7.1 System Model

A vehicle model, divided into modules, has been constructed on the basis of one presented in [93]. Each module represents a vehicle sub-system (Fig. 7.1) which interacts with the other sub-systems by means of input/output variables.

The result is an 18 d.o.f.s model that simulates satisfactorily the actual vehicle dynamic behaviour (Table 7.2).

7.1.1 Vehicle Model

Car chassis is considered as a rigid body. Tyre forces are transmitted to the front and rear axle modules respectively (Fig. 7.1). Front and rear axle modules apply two resultant forces and two resultant torques to points A and B respectively (Fig. 7.1). Aerodynamic forces are modeled assuming front lift, rear lift and drag coefficients depending on front and rear vertical position of points A and B with respect to the ground.

Front and rear independent suspension systems have a double wishbone layout, different layouts can be easily considered referring to available libraries [158]. Shock absorbers are modeled as non-linear components; forces transmitted by the shock absorber are function of the deformation speed. Non-linearity due to bump-stop characteristics has been included.

The torque-speed characteristic of the differential has been derived experimentally and accurately modeled. The braking system is modeled by considering the different distribution of braking force between front and rear axle.

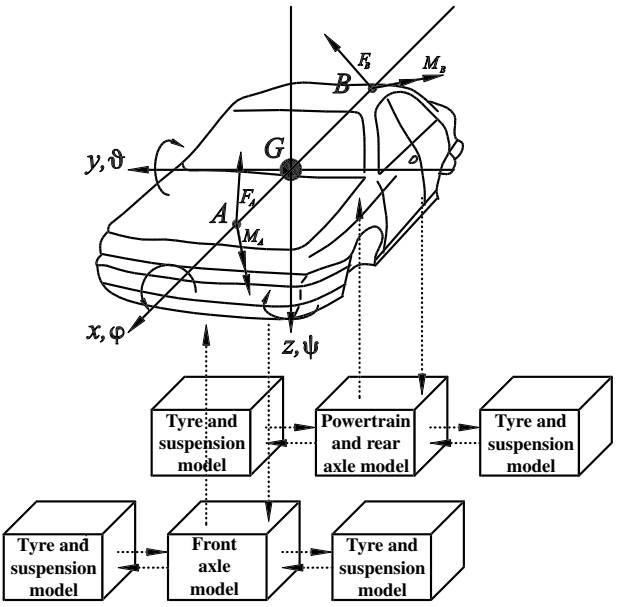


Fig. 7.1. Vehicle model. Dotted arrows refer to input–output variables describing the dynamics of each module

7.1.2 Tyre Model

A model that allows to estimate tyre forces with a good accuracy has been implemented on the basis of *Pacejka's Magic Formulæ* [10, 110, 194–196]. Tyre characteristics have been measured on a flat-track machine [208]. Experimental measurements have been used to identify Pacejka's model coefficients by applying the procedure reported in [187]. The model enables to simulate tyre behaviour at high slip angles and high level of longitudinal slip. The transient tyre behaviour is given by

$$\chi\left(F_z\right)\frac{dF_y}{dt} + v_w\,F_y = v_w\,F_{y,stat}\left(\alpha, F_z, \gamma\right) \tag{7.1}$$

This formulation allows to include the *relaxation length* effect. Combined effects of longitudinal and lateral slip have been considered [225].

Four d.o.f.s are used to simulate vertical motions of unsprung masses and four d.o.f.s refer to the rolling of the wheels.

7.1.3 Validation

Results given by simulations have been compared with track test measures on a test vehicle. The test vehicle is equipped with a data acquisition computer, wheel speed sensors, throttle position sensor, gear position sensor, brake circuit pressure sensors, steering wheel angle sensor, vehicle body

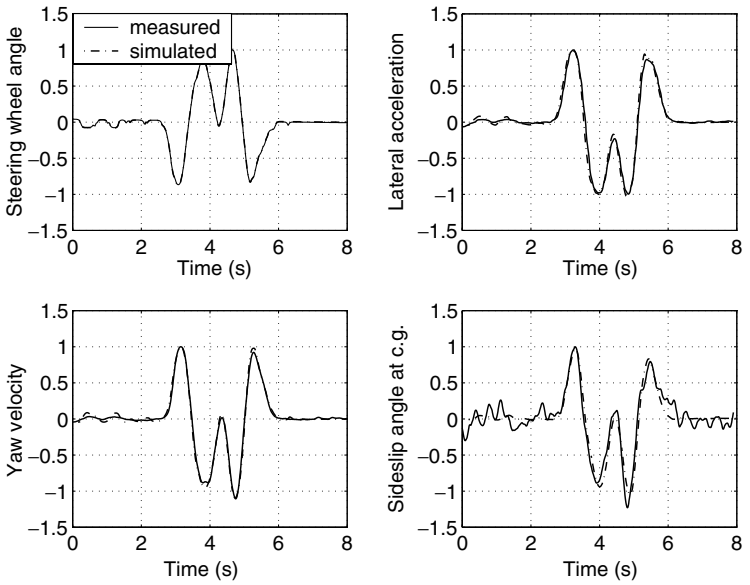


Fig. 7.2. Severe lane-change manoeuvre, numerical and experimental data. The measured steering wheel angle is used as input for the simulation code (throttle and selected gear are constant during the manoeuvre). Non-dimensional values on the y-axes ($y_{actual}/y_{max,measured}$)

accelerometers and gyroscopes, optical vector speed sensor. Measurements are matched directly with computed data. A parameter identification procedure is not needed. Comparisons considering a severe lane-change manoeuvre (ISO 3888) are shown in Fig. 7.2. The agreement is good. A test run on track is shown in Fig. 7.3. Again the model proves to represent fairly the dynamic behaviour of the car under consideration.

7.2 Design Variables

Eighteen design variables have been considered for the optimisation process. Design variables are related to suspension springs, anti-roll bars, hydraulic dampers, static toe angles, static camber angles and tyre characteristics. These design variables have been chosen as they are reputed to influence the behaviour of the vehicle taken under consideration. Moreover, these design variables can be easily modified on the considered vehicle.

Three types of optimisation strategies have been performed:

(1) Changing the stiffness of the front and rear suspension springs, the front anti-roll bar and the static toe and camber angles of front and rear wheels according to a grid (see Sect. 3.4.2).

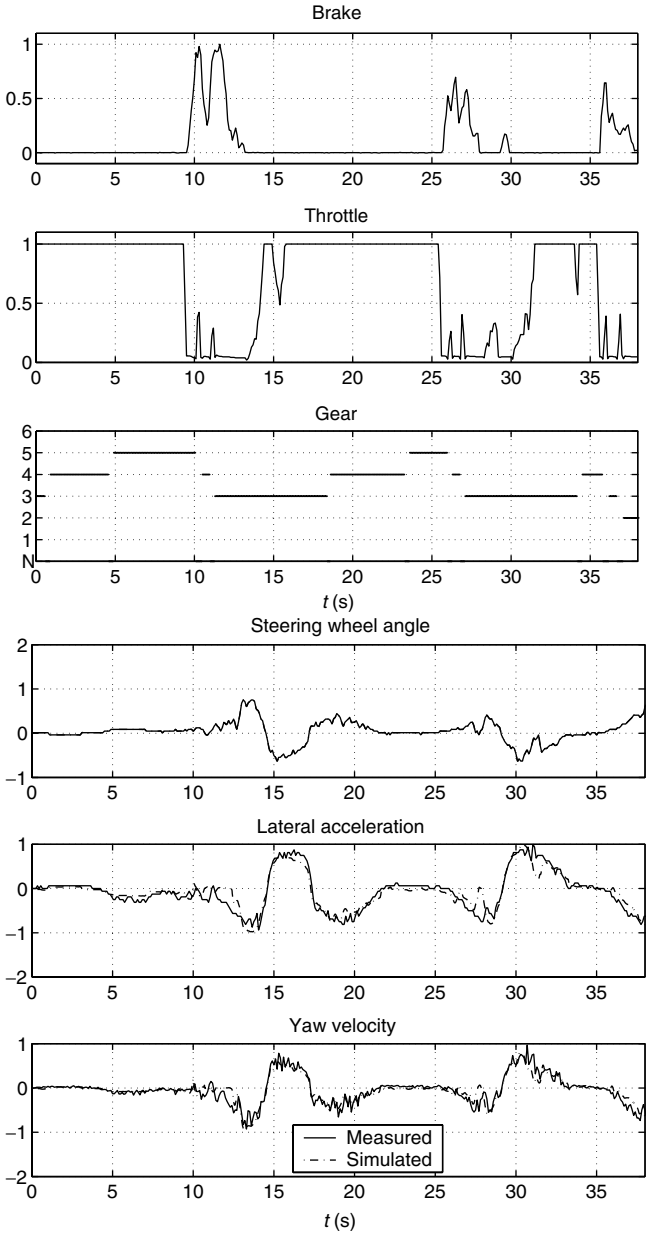


Fig. 7.3. Test run on track. The measured steering wheel angle, throttle position, gear and brake position are used as inputs for the simulation code. Non-dimensional values on y-axes ($y_{actual}/y_{max,measured}$). The only relevant discrepancy between measured and computed data (at $t = 10$–$13\,\mathrm{s}$) is due to the hitting of a kerb

(2) Changing suspension characteristics (stiffness, damping, toe and camber) by varying design variables continuously within specified ranges.
(3) Changing tyre and suspension characteristics by varying design variables continuously within specified ranges.

The design variables variations are shown in Table 7.1.

Table 7.1. Variation of the design variables values with respect to the reference vehicle

Design variable	1st optimisation	2nd optimisation	3rd optimisation
$k_{s,F}$	-20%, ref.	$[-50\%, +10\%]$	$[-50\%, +10\%]$
$k_{s,R}$	-20%, ref.	$[-60\%, +10\%]$	$[-60\%, +10\%]$
$k_{r,F}$	-40%, ref.	$[-80\%, +60\%]$	$[-80\%, +60\%]$
$k_{r,R}$	Ref.	$[-80\%, +100\%]$	$[-80\%, +100\%]$
$r_{b,l,F}$	Ref.	$[-60\%, +90\%]$	$[-60\%, +90\%]$
$r_{b,l,R}$	Ref.	$[-60\%, +90\%]$	$[-60\%, +90\%]$
$r_{r,l,F}$	Ref.	$[-50\%, +50\%]$	$[-50\%, +50\%]$
$r_{r,l,R}$	Ref.	$[-50\%, +50\%]$	$[-50\%, +50\%]$
$\delta_{st,F}$	-130%, -65%, ref., $+65\%$, $+130\%$	$[-130\%, +130\%]$	$[-130\%, +130\%]$
$\delta_{st,R}$	-80%, -40%, ref., $+40\%$, $+80\%$	$[-80\%, +80\%]$	$[-80\%, +80\%]$
$\gamma_{st,F}$	-30%, -15%, ref., $+15\%$, $+30\%$	$[-30\%, +30\%]$	$[-30\%, +30\%]$
$\gamma_{st,R}$	-30%, -15%, ref., $+15\%$, $+30\%$	$[-30\%, +30\%]$	$[-30\%, +30\%]$
$p_{1,F}$	Ref.	Ref.	$[-15\%, +15\%]$
$p_{1,R}$	Ref.	Ref.	$[-15\%, +15\%]$
$p_{2,F}$	Ref.	Ref.	$[-15\%, +15\%]$
$p_{2,R}$	Ref.	Ref.	$[-15\%, +15\%]$
C_F	Ref.	Ref.	$[-15\%, +15\%]$
C_R	Ref.	Ref.	$[-15\%, +15\%]$

Three optimisation strategies are considered. Design variables referring to the first optimisation have only discrete values ('ref.' means 'reference value')

7.2.1 Suspension System

Twelve design variables (system's model parameters) are tuned (Table 7.1) referring to

– the suspension spring characteristic (front and rear);
– the anti-roll bar characteristic (front and rear);
– the characteristic of the hydraulic dampers (front and rear) (four design variables);
– the suspension geometry (front and rear) (four design variables)

Table 7.2. List of main parameters values used in the presented simulations

l_F	Distance ofbody centre of mass from the front axle	1.497 m
$l_F + l_R$	Wheelbase	2.568 m
c_F/c_R	Half track front/rear	0.85 m/0.80 m
h	Height of the vehicle body centre of mass	0.428 m
m	Vehicle body mass	1,020 kg
m_F/m_R	Wheel mass (right + left) front and rear	80 kg/100 kg
$J_x/J_y/J_z$	Moments of inertia of the vehicle body about x-, y-, z-axis	330 kgm^2, 1,600 kgm^2, 1,800 kgm^2
C_dS	Drag	0.71 m^2
ET_{max}	Engine maximum torque	400 Nm@6,000 rpm
$C_{\alpha,F}/C_{\alpha,R}$	Cornering stiffness front/rear tyres	121,000 N/rad, 182,000 N/rad

Suspension Springs

Front and rear stiffness of the suspension springs, have been varied. The values of the design variables during the optimisation process are shown in Table 7.1.

Anti-roll Bars

Stiffness of front and rear anti-roll bar have been varied. These design variables during optimisation process are shown in Table 7.1.

Hydraulic Dampers

The force transmitted as a function of damper speed is modeled as shown in Fig. 7.4. Forces are given by

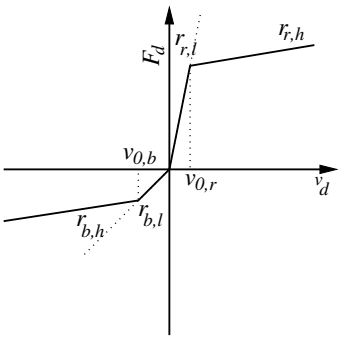


Fig. 7.4. Non-linear hydraulic damper characteristic, force vs. damper speed

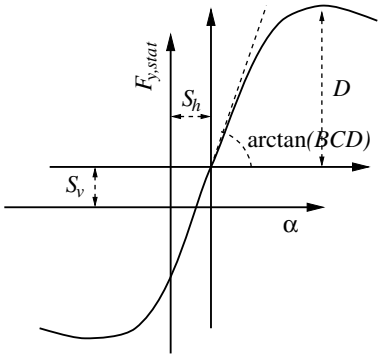


Fig. 7.5. A typical tyre characteristic (see Eq. (7.3))

$$\begin{cases} F_d = r_{r,h}\,(v_d - v_{0,r}) + r_{r,l}\,v_{0,r} & v_d > v_{0,r} \\ F_d = r_{r,l}\,v_d & v_d > 0,\ v_d \le v_{0,r} \\ F_d = r_{b,l}\,v_d & v_d > v_{0,b},\ v_d \le 0 \\ F_d = r_{b,h}\,(v_d - v_{0,b}) + r_{b,l}\,v_{0,b} & v_d \le v_{0,b} \end{cases} \tag{7.2}$$

Four design variables $r_{b,l}$, $r_{r,l}$, $r_{b,h}$, $r_{r,h}$ describe the shock absorber characteristic [87]. Velocities $v_{0,b}$ and $v_{0,r}$ are kept fixed, only $r_{b,l}$ and $r_{r,l}$ are varied during the optimisation process. Values assumed by these design variables are shown in Table 7.1.

Suspension Geometry

Suspension geometry is optimised by changing the static values of camber and toe angles. Values assumed by these design variables are shown in Table 7.1.

7.2.2 Tyre Characteristic

For pure lateral slip *Pacejka's Magic Formulæ* express the side force $F_{y,stat}$ as a function of the side slip angle α. The general form for a given value of vertical load and camber reads (Fig. 7.5)

$$F_{y,stat} = D \sin\{C \arctan[B(\alpha + S_h) - E\,(B(\alpha + S_h) - \arctan\,(B(\alpha + S_h)))]\} + S_v \tag{7.3}$$

Figure 7.5 shows the meaning of coefficients BCD and D for a typical tyre force characteristic. Coefficient D represents the peak value of the lateral force (if $S_v = 0$) and the product BCD corresponds to slope at the origin of the curve (if $S_h = 0$ and $S_v = 0$). The factor C controls the domain of the sine function in Eq. (7.3), and determines the shape of the resulting curve. The function of the cornering stiffness BCD reads

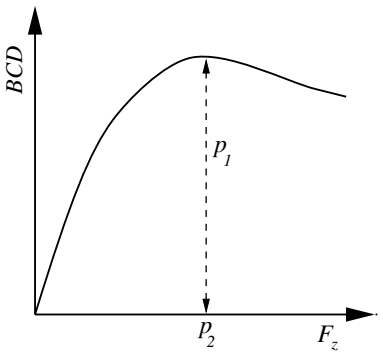


Fig. 7.6. Cornering stiffness as a function of vertical load according to expression (7.4) at camber zero

$$BCD = p_1 \sin\left[2\arctan\left(\frac{F_z}{p_2}\right)\right](1 - p_3\,|\gamma|) \tag{7.4}$$

For zero camber, the cornering stiffness has a maximum p_1 at $F_z = p_2$. Figure 7.6 shows the relationship given in (7.4).

C, p_1 and p_2 are chosen as design variables their variations are reported in Table 7.1.

7.3 Running Situations and Objective Functions

The vehicle–driver system and the external environment constitute a complex closed-loop system. The evaluation of the handling behaviour is made difficult by the human–vehicle interaction. Only a limited number of standard manoeuvres and running conditions have been defined by ISO and generally no evaluation criteria are given. In the following application a total number of six running situations (26 objective functions) are at first introduced.

7.3.1 Steady-state Turning

The simulation of the steady-state turning manoeuvre is based on ISO 4138 standard. The vehicle has to be driven on a known circular path at different speeds. This manoeuvre can be used to assess the tendency of the vehicle to understeer–oversteer by computing *understeer/oversteer gradient*:

$$\frac{d\delta}{da_c} \tag{7.5}$$

This parameter is strictly related to driver perception. Oversteering vehicles at high levels of lateral acceleration will be discarded during the search procedure. Considered objective functions are reported in Table 7.3.

Table 7.3. Optimisation scheme

Symbol	Objective function	Goal
Steady-state turning		
$a_{c,50m}$	Maximum a_c ($\rho = 50\,\mathrm{m}$)	Maximise
$\beta_{1.3\,g,50m}$	β ($a_c = 13\,\mathrm{m/s^2}$, $\rho = 50$ m)	Minimise
$P_{1.3\,g,50m}$	Power required ($a_c = 13\,\mathrm{m/s^2}$, $\rho = 50\,\mathrm{m}$)	Minimise
$a_{c,120m}$	Maximum a_c ($\rho = 120\,\mathrm{m}$)	Maximise
$\beta_{1.3\,g,120m}$	β ($a_c = 13\,\mathrm{m/s^2}$, $\rho = 120\,\mathrm{m}$)	Minimise
$P_{1.3\,g,120m}$	Power required ($a_c = 13\,\mathrm{m/s^2}$, $\rho = 120\,\mathrm{m}$)	Minimise
J-turn		
$t_{\dot{\psi}}$	$\dot{\psi}$ peak response time	Minimise
$O_{\dot{\psi}}$	$\dot{\psi}$ overshoot	Minimise
$\dot{\psi}_{s,s}$	Steady-state $\dot{\psi}$	Maximise
t_β	β peak response time	Minimise
O_β	β overshoot	Minimise
β_{max}	Maximum β	Minimise
$\beta_{s,s}$	Steady-state β	Minimise
t_{a_c}	a_c peak response time	Minimise
O_{a_c}	a_c overshoot	Minimise
$a_{c,max}$	Maximum a_c	Maximise
$a_{c,s,s}$	Steady-state a_c	Maximise
$RMS(\dot{\beta})_{J\,turn}$	Root mean-square of $\dot{\beta}$	Minimise
y_{path}	deviation from straight path	Maximise
$t_{a_c} - t_{\dot{\psi}}$	time delay between a_c and $\dot{\psi}$	Minimise
TB	TB factor	Minimise
Power on–off		
$RMS(\dot{\beta})_{Pwr\,on-off}$	*Root mean-square of* $\dot{\beta}$	Minimise
Braking on a bend		
$F_{z,max,left}$	Minimum vertical load on left rear wheel	Maximise
$F_{z,max,right}$	Minimum vertical load on right rear wheel	Maximise
Passing on a kerb while steering		
$RMS(F_{z,f,left})$	Root mean-square of the front left wheel vertical load	Minimise
$RMS(F_{z,r,left})$	Root mean square of the right left wheel vertical load	Minimise

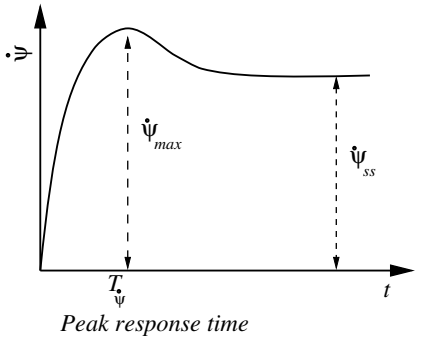


Fig. 7.7. Values for defining the peak response time and the overshoot with reference to yaw velocity response

7.3.2 J-Turn

The simulation of the J-turn manoeuvre (based on ISO 7401 standard with some minor adaptations due to the fact that a racing car is considered) provides information on transient vehicle response. The time histories of the centripetal acceleration, of the sideslip angle and of the yaw rate are considered as responses to a standard steering-wheel ramp input. The objective functions are the peak response time of the responses to a steering wheel step input and the overshoots of the responses (related to damping level of the system). In Fig. 7.7 a typical yaw velocity response to a steering step input is shown. Peak response time ($t_{\dot{\psi}}$) is highlighted. The expression of the overshoot is

$$O_{\dot{\psi}} = \frac{\dot{\psi}_{Max} - \dot{\psi}_{steady\ state}}{\dot{\psi}_{steady\ state}} \tag{7.6}$$

The peak response time and the overshoot for β and a_c are defined similarly as in (7.6). The TB factor [167] is an important objective function referring to a J-turn manoeuvre. TB is calculated as the product of the yaw velocity peak response time and the steady-state sideslip angle at the vehicle centre of gravity:

$$TB = t_{\dot{\psi}} \cdot \beta_{s,s} \tag{7.7}$$

A low value of the TB factor seems implying good vehicle handling characteristics. Objective functions for this manoeuvre are reported in Table 7.3.

7.3.3 Power On–Off while Steering

This manoeuvre is included in order to analyse the vehicle reactions to a power on–off input during turning. This type of input could excite undesirable

reactions that are perceived negatively by the driver. The vehicle runs into a bend in a steady-state condition at a high level of lateral acceleration and suddenly the throttle is fully opened (< 0.3 s), after 3 s it is fully released and then reopened (< 0.3 s). The steering-wheel angle is kept constant during the whole manoeuvre. Root mean square of $\dot{\beta}$ is the variable taken as a objective function during this manoeuvre (Table 7.3). This variable shows the tendency of the vehicle to oscillate around the centre of gravity.

7.3.4 Braking into a Bend

A typical braking manoeuvre before negotiating a bend is simulated. Load transfer in this running situation drives to unstability. This manoeuvre is composed by a sudden release of the throttle and subsequent full brake application. During the turning the brake is released gradually. The steer, brake and throttle inputs are shown in Fig. 7.8. The objective functions for this manoeuvre are reported in Table 7.3.

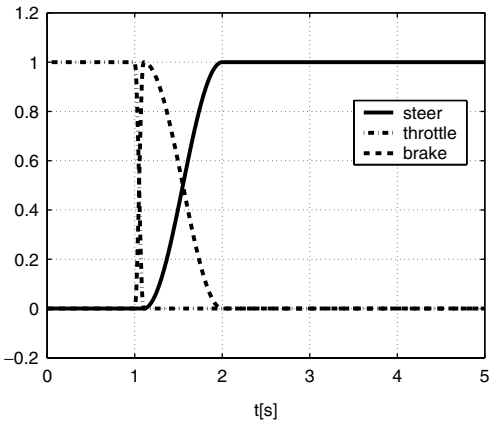


Fig. 7.8. Braking into a bend steer, brake and throttle inputs (non-dimensional values on y-axes (y_{actual}/y_{max}))

7.3.5 Passing over a Kerb While Steering

Passing over a kerb while steering provides information of both low-frequency and high-frequency vehicle responses. The obstacle is asymmetric and excites the roll motion of the vehicle body. The maximum height of the obstacle is 50 mm. The vertical shape of the kerb is shown in Fig. 7.9. The objective functions are indicated in Table 7.3.

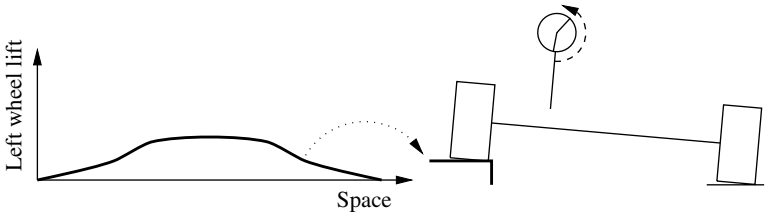


Fig. 7.9. Passing over a kerb

7.4 Search Method

The optimisation method [87,88] is based on two different mathematical models of the same car whose design variables have to be optimised. The first is a physical model, i.e. it is derived on the basis of a mechanical model strictly related to the vehicle system (the model is described in Sect. 7.1.1). The second model ('global approximation' see Chap. 4) is just a set of functions that are able to approximate many different relationships that exist between the design variables and the objective functions, describing the vehicle system responses.

The global approximation model has been constructed by means of an artificial neural network. Multi-layer perceptron neural networks have been used to approximate (within a very small computation time) complex non-linear functions with an arbitrary degree of accuracy [88, 113] (see Chap. 4). To avoid overtraining, a cross-validation procedure (see Chap. 4) [127] has been implemented.

7.4.1 Reduction of Objective Functions

N ($<10^3$) solutions have been analysed in order to reduce the number of objective functions that have to be taken into account. This has been done by using the 'Spearman rank correlation coefficient' (see Chap. 3) [182]. Some strongly correlated objective functions were found (even non-linear correlations were detected, Fig. 7.10), so the existence of redundancy was discovered (see Table 7.4) and the number of objective functions could be reduced [87, 182].

7.4.2 Pareto-optimal Solutions

The 'global approximation' model is used during the search procedure for the computation of the Pareto-optimal set (see Sect. 2.10).

The main advantage of multi-objective programming is that it provides a rational approach to decision making in the presence of conflicting criteria.

The global approximation allows a quick computation of the optimal sets of design variables, generally much faster than the computation by using the physical model.

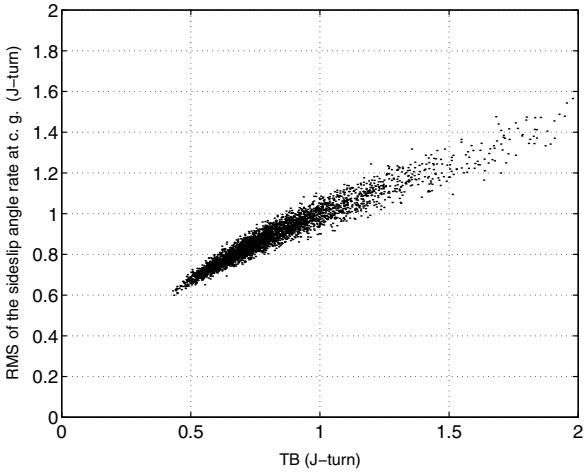


Fig. 7.10. Example of correlation between two objective functions in non-dimensional form. Spearman Rank Correlation Coefficient = 0.971

Table 7.4. Objective functions eliminated by using the Spearman rank correlation analysis

Eliminated objective function	Correlated objective function	r_s
$\beta_{1.3\,g,50\,m}$	$\beta_{1.3\,g,120\,m}$	0.997
$P_{1.3\,g,50\,m}$	$P_{1.3\,g,120\,m}$	0.996
$t_{\dot{\psi}}$	TB	0.926
t_{β}	TB	0.950
β_{max}	TB	0.964
$\beta_{s,s}$	TB	0.963
$a_{c,max}$	$\dot{\psi}_{s,s}$	0.947
$a_{c,s,s}$	$\dot{\psi}_{s,s}$	0.991
$RMS(\dot{\beta})_{Jturn}$	TB	0.971
y_{path}	$\dot{\psi}_{s,s}$	0.974
$F_{z,max,right}$	$F_{z,max,left}$	0.997

A large number ($>10^6$) of uniformly distributed solutions can be generated by the neural net approximation within a short time (see Chap. 4). Condition (2.11) can be checked and the Pareto-optimal solutions can be stored.

The designer can choose a preferred solution among those (and only those) belonging to the Pareto-optimal set.

7.5 Results

Two thousand uniformly distributed design variable vectors have been generated and used to compute the responses of the physical model.

By applying the procedure described in Sect. 7.4.1 only 15 objective functions were selected and considered for the subsequent optimisation. In fact they resulted fully representative of the 26 objective functions initially considered.

A multi-layer perceptron neural network has been trained approximating the 15 objective functions as a function of 18 design variables. The mean error of the trained neural network is less than 3%.The network architecture has two layers of 70 and 40 neurons respectively. The training set is composed of only 1,000 responses of the physical model out of the 1,200 generated. A number of candidate optimal solutions have been selected from the Pareto-optimal set.

Figures 7.11 and 7.12 show two projections of the Pareto-optimal set onto two planes. Data are reported in non-dimensional form. Improvements are obtained for all the objective functions.

A preferred solution has been chosen from the Pareto-optimal set and it has been denoted as *optimised car*. The results of the third optimisation (Table 7.1) are reported, because the improvements obtainable by varying a reduced set of design variables (first and second optimisation) were not fully satisfactory. The improvements with respect to the reference car are shown in Table 7.5.

The mean improvement (on all the 26 objective functions) is about 10%. The handling behaviour is significantly improved. The reference car does not seem belonging to the Pareto-optimal set. Comparisons between reference car and optimised car are shown in Figs. 7.13–7.15, referring to a J-turn manoeuvre, a steady-state turning manoeuvre and a power on–off while steering manoeuvre. Steady-state turning manoeuvres (Fig. 7.14) show higher level of centripetal acceleration that can be achieved by the optimised car with respect to the reference car. This is due to a better load transfer distribution between front and rear axles. Moreover, in this manoeuvre the sideslip angle is reduced because of the higher cornering stiffness of the rear tyres. The power required is reduced (up to 12% at the highest lateral acceleration) because the slippages of the optimised vehicle are less than the slippage of the reference vehicle. J-turn manoeuvre reveals (Fig. 7.13) a more damped motion of the optimised vehicle with respect the reference vehicle due to a different stiffness and damping distribution between front and rear axles. This is done by maintaining the TB factor at about the same level. The same damping effect holds for Power on–off manoeuvre (Fig. 7.15). $F_{z,max,left}$ is improved because $r_{b,l,F}$ is augmented and $r_{r,l,R}$ is reduced. $RMS(F_{z,r,left})$ is less because damping of the rear shock absorbers are greatly decreased, being $k_{s,R}$ equal.

7.5.1 Comparison of the Performances of Global Approximation Methods

Three approximation methods, polynomial approximation, RBFNN and MLPNN (see Chap. 4) are trained using a sample of 2,023 point obtained from a $(t, 16)$-sequence in base 17 (see Sect. 3.4.2). The design variables and the objective functions are normalised before the application of the methods. The set of 2,023 design points is split into two set $\mathcal{D}(n_D)$ and $\mathcal{V}(n_V)$ to train

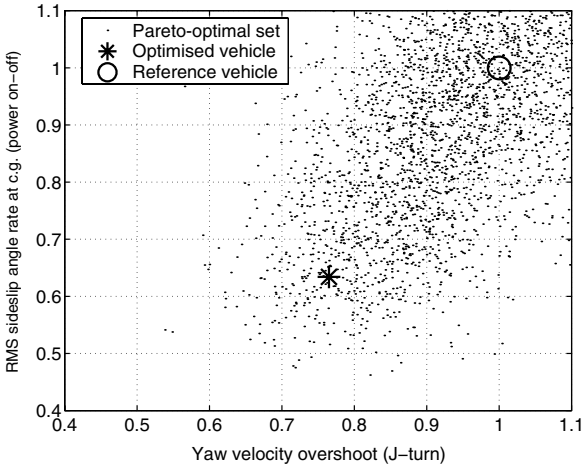


Fig. 7.11. Pareto-optimal set projection onto $(RMS(\dot{\beta})_{Pwr\,on-off}, O_{\dot{\psi}})$ plane. Data are plotted in non-dimensional form

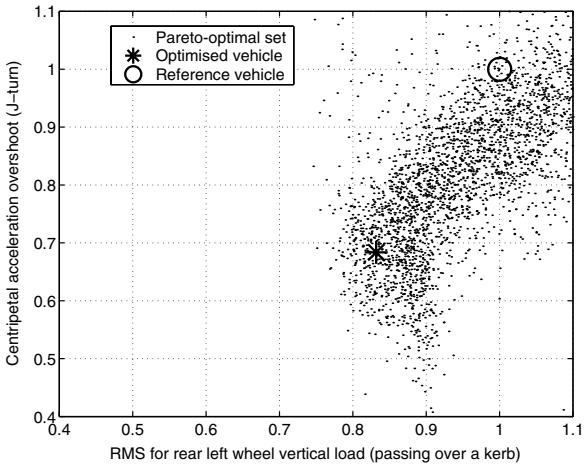


Fig. 7.12. Pareto-optimal set projection onto $(O_{a_c}, RMS(F_{z,r,left}))$ plane. Data are plotted in non-dimensional form

Table 7.5. Design variables and objective functions of a preferred car denoted as 'optimised vehicle'

Design variable	Variation with respect to the reference car (%)
$k_{s,F}$	−10
$k_{s,R}$	0
$k_{r,F}$	−15
$k_{r,R}$	+73
$r_{b,l,F}$	+83
$r_{b,l,R}$	−38
$r_{r,l,F}$	−18
$r_{r,l,R}$	−49
$\delta_{st,F}$	−43
$\delta_{st,R}$	+31
$\gamma_{st,F}$	+9
$\gamma_{st,R}$	+18
$p_{1,F}$	+1
$p_{1,R}$	+8
$p_{2,F}$	+12
$p_{2,R}$	−2
C_F	+3
C_R	+10

Objective function index	Improvement with respect to the reference car (%)
$a_{c,\,50\,m}$	3
$a_{c,\,120\,m}$	3
$\beta_{1.3\,g,120\,m}$	6
$P_{1.3\,g,120\,m}$	2
$O_{\dot{\psi}}$	24
$\dot{\psi}_{s,s}$	1
O_{β}	16
t_{a_c}	0
O_{a_c}	32
$t_{a_c} - t_{\dot{\psi}}$	1
TB	0
$RMS(\dot{\beta})_{Pwr\,on-off}$	36
$F_{z,max,left}$	15
$RMS(F_{z,f,left})$	0
$RMS(F_{z,r,left})$	16

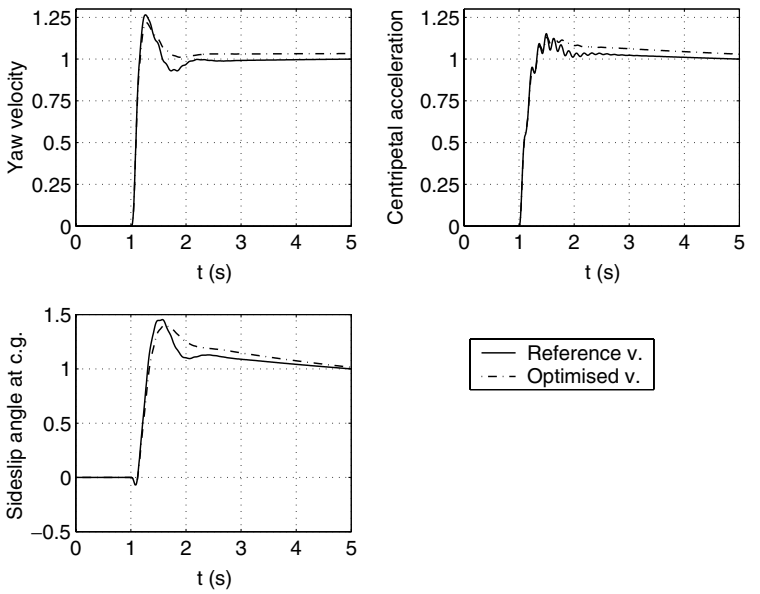


Fig. 7.13. J-turn manoeuvre. Comparisons between reference and optimised vehicle. (Data are normalised with respect to the steady-state values of the reference vehicle)

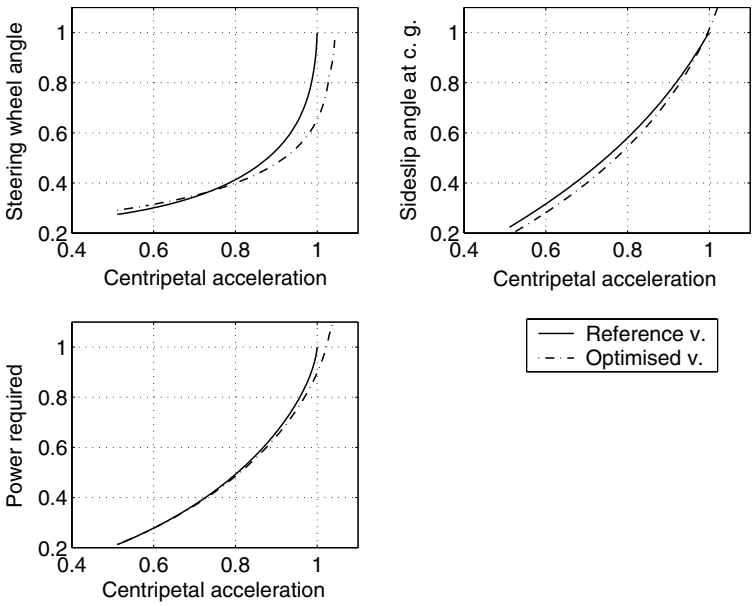


Fig. 7.14. Steady-state turning. Comparisons between reference and optimised vehicle. (Data are normalised with respect to the maximum values of the reference vehicle).

the multi-layer perceptron neural network. The $\mathcal{D}(N_D)$ is the training set and $n_D = 1{,}734$, $\mathcal{V}(N_V)$ is the validation set used for cross-validation (see Sect. 4.6.1) and $N_V = 289$. The models are tested on a set $\mathcal{T}(N_T)$ of $N_T = 14{,}739$ points.

Different types of error estimators are considered.

1. The *mean relative error* is defined by

$$\Delta_j = \frac{1}{N_T} \sum_{i=1}^{N_T} \frac{|\hat{y}_{ij} - y_{ij}|}{|y_{ij}|} \tag{7.8}$$

   lower Δ_j refers to a good approximation.
2. The *normalised standard deviation* is defined by

$$\Sigma_j = \sqrt{\frac{\sum_{i=1}^{N_T} \left(\hat{y}_{ij} - y_{ij}\right)^2}{\sum_{i=1}^{N_T} \left(y_{ij} - \bar{\bar{y}}_j\right)^2}} \tag{7.9}$$

   where $\bar{\bar{y}}_j = \frac{1}{N_T} \sum_{i=1}^{N_T} y_{ij}$. Lower Σ_j refers to good a approximation.
3. The *correlation coefficient* between the estimated $\hat{y}_{ij}$ output and the test output y_{ij}

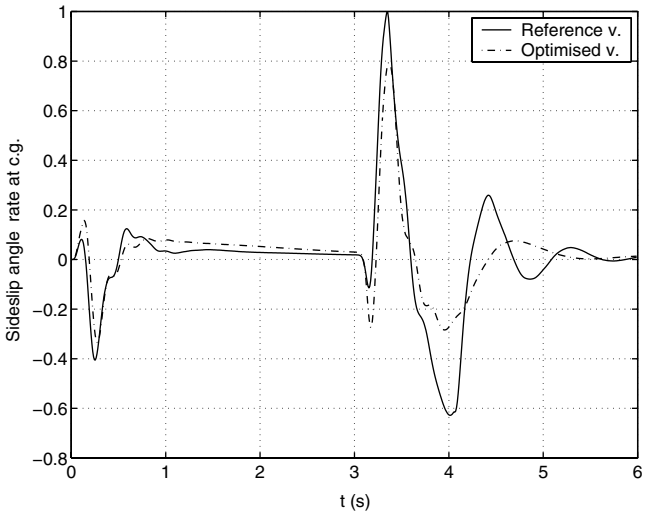


Fig. 7.15. Power on–off manoeuvre (see Sect. 7.3.3). Comparisons between reference and optimised vehicle. (Data are normalised with respect to the maximum values of the reference vehicle)

$$R_j = \frac{\sum_{i=1}^{N_T} \left(\hat{y}_{ij} - \hat{\bar{y}}_j\right)\left(y_{ij} - \bar{y}_j\right)}{\sqrt{\sum_{i=1}^{N_T} \left(\hat{y}_{ij} - \hat{\bar{y}}_j\right)^2 \sum_{i=1}^{N_T} \left(y_{ij} - \bar{y}_j\right)^2}} \tag{7.10}$$

where $\hat{\bar{y}}_j = \frac{1}{N_T} \sum_{i=1}^{N_T} \hat{y}_{ij}$. Value of R_j closer to 1 refers to a good approximation.

Four approximation models have been compared in terms of accuracy:

1. linear approximation (Sect. 4.3.1). The number r of function employed is 17 for every objective function;
2. polynomial quadratic approximation (Sect. 4.3.1). The number r of function employed is 153 for every objective function;
3. radial basis function neural network regularised and trained using regression trees (Sect. 4.6.2) with Gaussian basis functions. N_{min} is 2 (N_{min} is the unique parameter to be set by the user and refers to the number of points that are considered to stop the generation of the regression tree);
4. Radial basis function neural network regularised and trained using regression trees (Sect. 4.6.2) with thin plate spline basis functions. N_{min} is 2;
5. Multi-layer perceptron neural network (MLPNN) trained using early stopping (Sect. 4.6.1). The final network architecture is composed by two hidden layers with 98 and 105 neurons.

It is important to notice that the approximations model based on NN needs to be designed and trained properly. This procedure is time-consuming and the

generation of the approximation model requires more time than other methods (see Table 7.11). Actually, a trial-and-error procedure has to be completed for the setting of the number of neurons and hidden layers. In contrast to the complex setting of parameters referring to the architecture of MLPNN, the RBFNN requires to set only one parameter: N_{min} that defines the regression tree structure (see Sect. 4.6.2).

By inspection of Table 7.6, one may easily notice that the MLPNN is superior with respect to other methods for approximation purposes (see mean Δ_j values). The same could be stated for the other test criteria (Eqs. (7.9) and (7.10); Tables 7.7 and 7.8). In these cases RBFNN methods show performance (both with Gaussian basis functions and with thin plate spline basis functions) close to MLPNN, confirming the results presented in [229]. The results (Tables 7.6–7.8) show also that the linear approximation model is inadequate for the problem under consideration. The quadratic approximation model may be good for the approximation of some objective functions but for other objective functions the approximation is insufficient (see maximum values in Tables 7.6– and minimum values in Table 7.8)[1].

Table 7.6. Relative errors

Method	1(%)	2(%)
Linear	5.67	27.92
Quadratic	4.16	26.52
RBFNN gaussian	2.87	9.49
RBFNN thin plate spline	2.91	9.38
MLPNN	2.44	15.46

(1) Mean $\triangle_j$ over all the considered objective functions.
(2) Maximum $\triangle_j$ over all the objective functions

Table 7.7. Normalised standard deviation

Method	1	2
Linear	0.270	0.482
Quadratic	0.175	0.402
RBFNN gaussian	0.156	0.385
RBFNN thin plate spline	0.158	0.386
MLPNN	0.132	0.298

(1) Mean Σ_j over all the considered objective functions.
(2) Maximum Σ_j over all the objective functions

[1]We must emphasise that this is only the time needed for a single training of the MLPNN; MLPNN needs multiple training to determine the optimal network architecture

Table 7.8. Correlation coefficient between estimated $\hat{y}$ and real y of different methods

Method	1	2
Linear	0.898	0.732
Quadratic	0.948	0.809
RBFNN gaussian	0.971	0.877
RBFNN thin plate spline	0.970	0.881
MLPNN	0.980	0.960

(1) Mean R_j over all the considered objective functions.
(2) Minimum R_j over all the objective functions

Table 7.9. Pareto-optimal solutions classified correctly: solutions that in terms of design variables are identical to the ones directly computed by means of the vehicle model

Method	$PO(\%)$
Linear	66.60
Quadratic	67.60
RBFNN gaussian	73.01
RBFNN thin plate spline	74.55
MLPNN	77.10

PO is the ratio between the number of Pareto-optimal solutions obtained by means of the approximated model $\hat{f}$ and the number of Pareto-optimal solutions obtained by means of the vehicle model (expressed in percentage)

The ultimate purpose of multi-objective optimisation methods integrated by global approximation is to compute a good approximation of the Pareto-optimal set. The Pareto-optimal set has been computed for the test set $\mathcal{T}(N_T)$ by using Definition 2.7. The test set $\mathcal{T}(N_T)$ has been obtained directly by using the vehicle model. In the same way the approximated Pareto-optimal set has been obtained by using the approximated models by considering the same design variables vectors which define the test set $\mathcal{T}(N_T)$. Table 7.9 shows the comparisons. By using the MLPNN, 77.1% of the points belonging to the approximated Pareto-optimal set are the same Pareto-optimal points (in terms of design variables) found directly by using the vehicle model. This is the best performance even if the two RBFNN obtain 73.01 and 74.55%, not so far from MLPNN. The Pareto-optimal set has been computed also for a test set $\mathcal{F}(N_F)$, obtained from $\mathcal{T}(N_T)$ considering the solution that improve all the objective functions with respect to the reference vehicle ($N_F = 787$) (see Table 7.10). In this case for the MLPNN, 71.00% of the points belonging to the approximated Pareto-optimal set are coincident with the solutions

Table 7.10. Pareto-optimal solutions correctly classified on the subset of the set solutions that improve all the objective functions with respect to a reference vehicle: solutions that in terms of design variables are identical to the ones directly computed by means of the vehicle model

Method	$PO(\%)$
Linear	37.73
Quadratic	64.26
RBFNN gaussian	69.86
RBFNN thin plate spline	70.37
MLPNN	71.00

PO is the ratio between the number of Pareto-optimal solutions obtained by means of the approximated model $\hat{f}$ and the number of Pareto-optimal solutions obtained by means of the vehicle model (expressed in percentage)

obtained by using the vehicle model; in this case too this is the best performance even if the two RBFNN (69.86 and 70.37%) perform in similar way as MLPNN.

Finally, Table 7.11 shows respectively the time needed to generate the training set, the time needed to obtain the parameters of the approximation model to generate the approximated objective functions from the test set and the time needed to complete the optimisation process on a Pentium PIII 800 using a sample of 100,000 solutions. The time needed to generate the Pareto-optimal set from the test set by obtaining the objective functions directly from the vehicle model without using global approximation is more than $1 \cdot 10^7$ s. In all the cases the optimisation process obtained by using global approximation is by far cheaper in terms of computation time with respect to the optimisation process obtained without using the approximation method. The time needed to the global approximation methods is mainly due to the evaluation of the training set. For the MLPNN, the time needed for the determination of the net architecture is relevant. RBFNN can be successfully used when a trade-off between accuracy and computation time is required.

7.6 Conclusion

In order to achieve the best performances from a racing car, the contemporary tuning of tyre and suspension design variables has proved to be necessary. To optimise a racing car, the multi-objective optimisation approach has been followed (see Chap. 3). This approach requires both validated models and a well-defined experimental activity aimed to derive the design variables that influence the dynamic behaviour of the car, namely tyre characteristics and other chassis parameters.

Table 7.11. Computation time, expressed in seconds, obtained using a Pentium PIII 800 processor

Method	1 (s)	2 (s)	3 (s)	Total (s)
Linear	1.4×10^5	6	2.4×10^2	1.4×10^5
Quadratic	1.4×10^5	7.0×10^1	7.0×10^2	1.4×10^5
RBFNN Gaussian	1.4×10^5	1.9×10^4	1.3×10^3	1.6×10^5
RBFNN thin pl. spl.	1.4×10^5	2.5×10^4	1.3×10^3	1.7×10^5
MLPNN	1.4×10^5	1.5×10^4	1.7×10^3	$\simeq 2.1 \times 10^5$

(1) Time needed to generate the training set.
(2) Time needed to generate the approximation model.
(3) Time needed to generate the Pareto-optimal set using the Quasi-Monte Carlo method (100,000 evaluations). The time needed to generate the Pareto-optimal set using the quasi-Monte Carlo method without global approximation is $>10^7$ s

According to the global approximation procedure, the complex physical model of the vehicle has been substituted by an artificial neural network that is able to approximate very closely the relationships between tyre/chassis design variables and objective functions. By means of the neural network many ($>10^6$) simulations have been performed within an extremely short time (24 h). From these simulations the optimal solutions have been selected according to Pareto theory. The dynamic behaviour of a preferred optimal car has been discussed in detail. Sensible improvements of the whole objective functions describing the vehicle dynamic behaviour have been obtained for this optimised car, with respect to the reference one.

Professional drivers have tested a number of cars fitted with the optimal suspension/tyre settings. The simulation results have been confirmed by ground tests.

The presented results showed that multi-layer perceptron neural network gives the highest accuracy. A similar performance in approximating objective functions is provided by radial basis function neural network. This result seems very encouraging because the training of RBFNN require the user to set only one parameter, contrary to MLPNN which requires much expertise to set the number of hidden layers and the number of neurons in order to obtain the best global approximation performance. Obviously, the presented results are not intended to be a general comparison between the considered global approximation methods, although the testing has been performed on an actual and very complex engineering problem. The results can be reputed valid only for similar optimisation problems.

The global approximation procedure may be used to analyse a large number of optimal solutions. The approach based on the presented low discrepancy sequences allows not only to build good approximation models but also to improve the knowledge of the designer about the relationships between design variables and objective functions. From the analysis of the presented

simulations, it appears that the Pareto-optimal set can be computed adequately by using high accuracy approximators.

With reference to the design of a vehicle tyre/suspension system, the adopted procedure has proved to be reliable and effective for obtaining higher car performance reducing the time needed in expensive ground tests.

The method allowed both to synthesise and analyse the optimal solutions in order to improve not only the performances of the car under consideration but also the know-how and the skills of designers.

8

Integrated Controls for the Improvement of Ride, Comfort, Handling and Active Safety of Road Vehicles

Sophisticated control applications are increasingly used in automobiles. Active suspensions, active four-wheel steering, controlled four-wheel drive, anti-skid brake systems, traction control are commercially available. All these sophisticated controls have been developed separately, and now a method for their integration is proposed the basis on of multi-objective optimisation (MOO). MOO (see Chap. 3) is substantially a framework to help and guide a designer who has to handle (optimise) many objective functions (i.e. vehicle performances) at the same time. Many design variables (control gains and suspension settings) have to be tuned to achieve the desired vehicle performances during many different situations (curving, accelerating, braking on rough, curved, wet roads). This appears to be one of the main design problems of chassis systems (either active or passive) [143, 214]. The problem of the synthesis of a novel controller for integrated active chassis systems cannot be separated from the problem of trade-off, i.e. the problem of optimisation of complex systems [143]. The synthesis of this novel controller is based on innovations proposed by the designer after an in-depth analysis of trade-offs. In the literature, a number of attempts have been presented for the integration of active chassis systems [63, 114, 139, 148, 186, 221, 255, 264]. All these attempts have focused on the analysis of physical phenomena occurring during few special driving situations. Although these analyses are very useful to provide an insight to the problem of integrated controls, they are not sufficient for a final synthesis of a controller. Actually, a trade-off stage is necessary to account properly for the inherent complexity of this design task, as the following study will show. Similar studies to the one presented in this chapter have been proposed in [23, 66, 125, 126, 272]; they were based on multi-criteria optimisation, but they were mostly not concerned with integrated controls.

A numerical application, derived from [144], will illustrate the capabilities of the method.

8.1 System Models and Reference Driving Situations

8.1.1 System Models

Two models of the same automobile have been used for numerical simulations. The first model (whose equations of motion have been reported in [158]) is devoted essentially to simulate handling manoeuvres on smooth road. It has 18 d.o.f: 6 d.o.f. of the body, 4 d.o.f. of the wheels rotating around their axes, 8 d.o.f. of the vertical and lateral movements of the wheels.

The equations of motion read

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} = F(\mathbf{q}, \dot{\mathbf{q}})$$

where $\mathbf{q}$ is the vector of 36 coordinates. $\mathbf{M}$ is the mass matrix. $\mathbf{F}$ is the vector of non-linear forces acting on each body of the vehicle. The complete non-linear pneumatic tyre characteristics have been included in the model. They have been measured and the data have been processed according to the interpolation formulae reported in [195]. The tyre characteristics on wet road have been extrapolated from the previous ones referring to a dry surface. The extrapolation has been performed according to the indications reported in [29] (see Fig. 8.8).

The second model is devoted to simulate ride dynamics on rough road [85]. It is linear and has 12 d.o.f: five rigid movements of the body (displacement along x-axis is superimposed), four displacements of the wheels with respect to the body, the displacement of the driver seat with respect to the body, 2 d.o.f. to account for engine vibrations. The equations of motion contain the first-order-lag equations describing the lateral forces at the tyres [167]. During the linearisation of the equations of motion the actual (non-linear) suspension kinematics and damper characteristics are accounted for. The equations of motion read

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{R}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} = F$$

where $\mathbf{q}$, in this case, is the vector of lagrangian coordinates; $\mathbf{M}, \mathbf{R}, \mathbf{K}$ are the mass, damping and stiffness matrixes respectively; F is the vector of excitation forces due to road irregularities.

8.1.2 Reference Driving Situations

In order to provide sufficient information for assessing robustness of the proposed control schemes, a number of different driving situations have been selected for simulations. Braking actions (and problems related to throttle release, i.e. power-off) have not been considered here because the aim of this research work is to show the effectiveness of the proposed method, more than to provide an industrial solution to the problem of integrated controls.

The selected driving situations, each one of them corresponding to an objective function, are reported in Table 8.1. The way in which these driving

Table 8.1. Driving situations relevant for assessing the overall dynamic behaviour of an automobile

Driving situation	#	Objective function	Symbols	Model
	1	Discomfort	D	
Ride Comfort/road holding: running straight ahead on rough road	2	Std. dev. dynamic wheel load (front)	$\sigma_{F_z,F}$	
	3	Std. dev. dynamic wheel load (rear)	$\sigma_{F_z,R}$	12 d.o.f. linear
	4	Std. dev. relative displ. wheel–body (front)	$\sigma_{ws,F}$	
	5	Std. dev. relative displ. wheel–body (rear)	$\sigma_{ws,R}$	
	6	Ramp steering input on dry road	J_1	
	7	Ramp steering input on wet road	J_2	
Handling: steering on even road	8	Power-on while steering on dry road	P_1	
	9	Power-on while steering on wet road	P_2	18 d.o.f. non-linear
	10	Wind gust	W	
	11	Power-on μ–split	P_μ	
	12	Lane change on dry road	L_1	
	13	Lane change on wet road	L_1	

Each driving situation corresponds to an objective function.

situations have been characterised for the mathematical definition of the corresponding objective functions is now explained.

1. *Discomfort.* The discomfort index has been computed according to the indications reported in [1,167]. While the car runs straight ahead on rough road, the accelerations in each main direction at the driver seat and the accelerations at hands and feet are filtered to account for actual human perception. The computations are performed in the frequency domain

$$D^2 = \int_{-\infty}^{+\infty} P_{SD\,D}(\omega) d\omega$$

2, 3. *Standard deviation of dynamical wheel loads (front and rear).* With reference to the computation of power spectral densities of the responses of the vehicle subjected to random inputs from the road [167],

the index of wheel–ground contact derives from the integration of the power spectral density of the dynamical wheel load

$$\sigma^2_{F_{zi}} = \int_{-\infty}^{+\infty} P_{SD\,F_{zi}}(\omega)d\omega$$
$$i = F(\text{front}), R(\text{rear})$$

4, 5. *Standard deviation of relative displacement wheel–body (vertical direction z, front and rear).* Again, with reference to the computation introduced above, the index of wheel–body relative displacement derives from integration of the corresponding power spectral density

$$\sigma^2_{ws,i} = \int_{-\infty}^{+\infty} P_{SD\,ws,i}(\omega)d\omega$$
$$i = F(\text{front}), R(\text{rear})$$

6, 7. *Ramp steering-wheel input (on dry and wet roads).* The manoeuvre is defined in [245]. The vehicle runs at 100 km/h. The steering wheel is turned at a speed of 200°/s to reach a steady deviation (30°). Due to the ramp input at the steering wheel, the yaw rate $\dot{\psi}$ and sideslip angle β at the centre of gravity (Fig. 8.1) vary as indicated in Fig. 8.2. The two reference variables that have been used to build up the

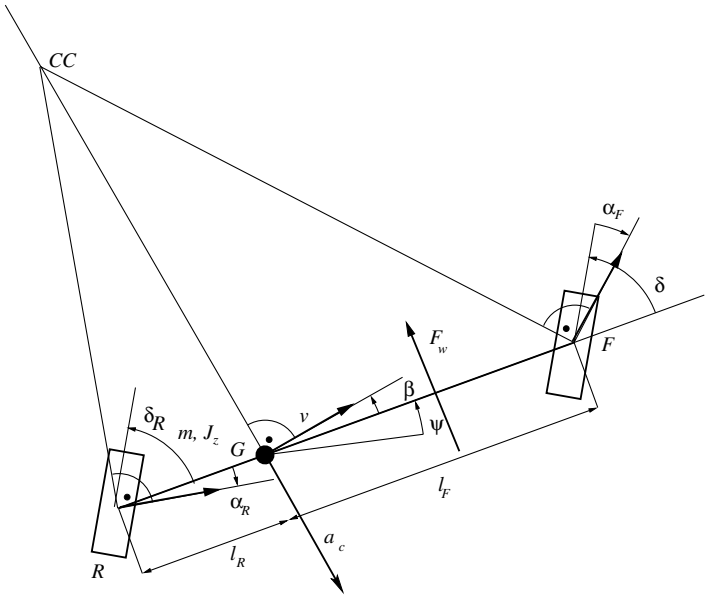


Fig. 8.1. Variables relevant for describing the horizontal motions of a vehicle, and used by active controls. Angles are shown in their respective positive orientations. F_w is the force related to wind gust

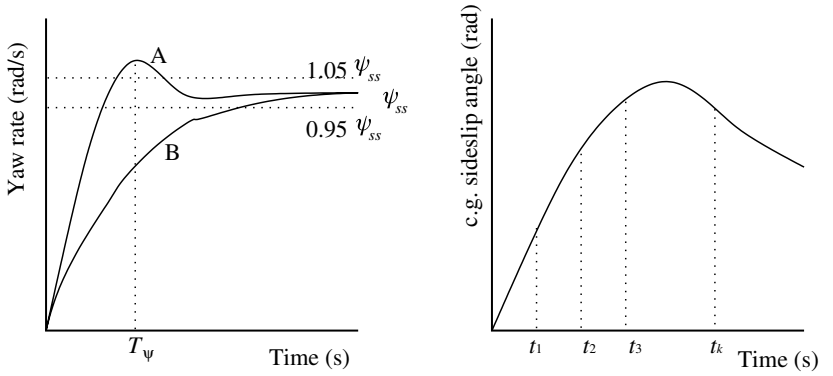


Fig. 8.2. Ramp steering-wheel input. Peak response time of yaw rate $\dot{\psi}$ and sideslip angle at centre of gravity β (symbols in Fig. 8.1). Case A: with overshoot; case B, without overshoot

objective function are the peak response time $T_{\dot{\psi}}$ and the sideslip angle β (at different times). $T_{\dot{\psi}}$ is defined in Fig. 8.2: Two cases (A and B) are possible depending on the parameters which influence the phenomenon. $\beta(t)$ is shown in Fig. 8.2. The objective function reads

$$J_i = T_{\dot{\psi}} + q_i \sum_{j=1}^{k} \beta_j^2(t_j)$$
$$i = 1\,(\text{dry}), 2\,(\text{wet})$$

where q_i is a weighting parameter.

8, 9. *Power-on while steering (on dry and wet roads).* The throttle is pushed down for 0.5 s from the position which is needed to maintain the velocity of 30 km/h on a path with radius 40 m, to a position which is next to that one corresponding to full spin of both wheels. The steering wheel is kept fixed. This manoeuvre involves particularly the wheel slip controls [227]. In this study, the objective function has been constructed on the basis of (1) the maximum distance of the c.g. from the reference steady-state path, (2) the sum of sideslip angles at different times (Fig. 8.3). The objective function reads

$$P_i = d_i + q_{P,i} \sum_{j=1}^{k} \beta_j^2(t_j)$$
$$i = 1\,(\text{dry}), 2\,(\text{wet})$$

where $q_{P,i}$ is a weighting parameter.

10. *Wind gust.* A typical wind gust has been described as follows. The vehicle is running at 100 km/h, when at the pressure centre on the side of the vehicle, a lateral force equal to 1000 N acts as a step input.

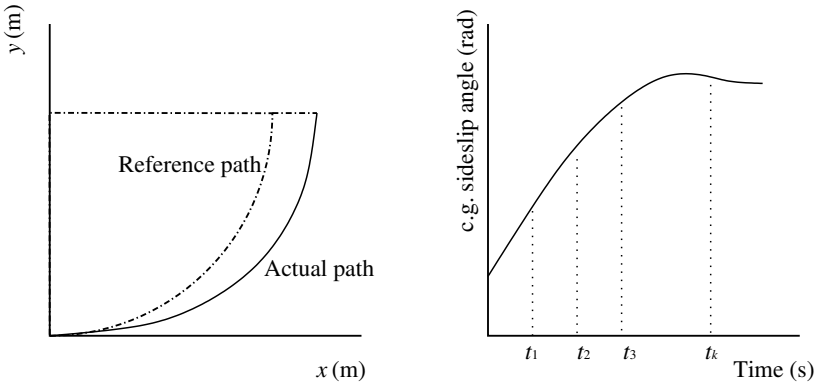


Fig. 8.3. Power-on while steering. Reference path and actual path of the c.g of the accelerating vehicle. Typical c.g. sideslip angles during this manoeuvre

The road surface is considered wet (worst case). The steering wheel is kept fixed. The objective function has been constructed on the basis of (1) the maximum distance of the c.g. from the reference path (2) the sum of sideslip angles β at different times (Fig. 8.4). The objective function reads

$$W = d + q_W \sum_{j=1}^{k} \beta_j^2(t_j)$$

where q_W is a weighting parameter.

11. *Power-on on μ-split.* The vehicle is running at 100 km/h. When the driving wheels encounter the μ-split road surface, the throttle is set to

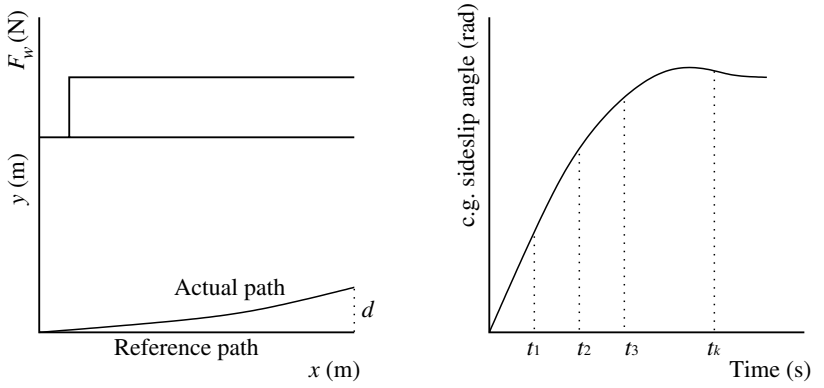


Fig. 8.4. Wind gust. Reference path and actual path of the c.g of a vehicle subjected to a wind gust F_w. Typical c.g. sideslip during this manoeuvre for a vehicle having a pressure centre position in front of the c.g.

a fixed level (rather high) after a ramp input lasting 0.2 s. The steering wheel is kept fixed. The objective function has been constructed on the basis of (1) the maximum distance of the c.g. from the reference path (2) the sum of sideslip angles β at different times (Fig. 8.5). The objective function reads

$$P_\mu = d + q_{P_\mu} \sum_{j=1}^{k} \beta_j^2(t_j)$$

where q_{P_μ} is a weighting parameter.

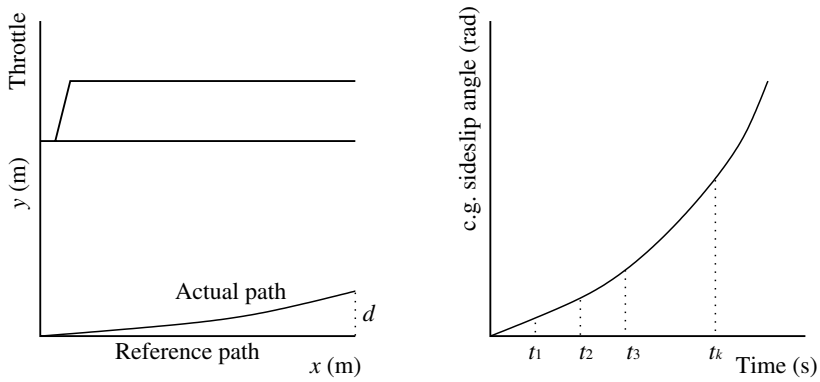


Fig. 8.5. Power-on μ–split. Reference path and actual path of the c.g of the vehicle. Throttle position t_r. Typical c.g. sideslip during this manoeuvre

12, 13. *Lane change (on dry and wet roads).* The vehicle has to follow the path indicated in Fig. 8.6. The lane-change manoeuvre requires a driver model that has been modelled as shown in Fig. 8.6. A commonly accepted driver model is still under development [102, 168]. Here, the driver is assumed to act in order to minimise the error ϵ

$$\epsilon = y_{des}(x_{prev}(t+\tau)) - y_{prev}(t+\tau)$$

and the steering wheel angle is then

$$\delta = \varkappa\epsilon$$

The stability of the vehicle–driver system depends on the values of $\varkappa$ and τ. After preliminary investigations, the following values for driving on dry and wet roads have been set

$$\varkappa = 0.012\,\mathrm{m}^{-1}, \tau = 0.73\,\mathrm{s} \quad \text{(dry)}$$
$$\varkappa = 0.006\,\mathrm{m}^{-1}, \tau = 0.97\,\mathrm{s} \quad \text{(wet)}$$

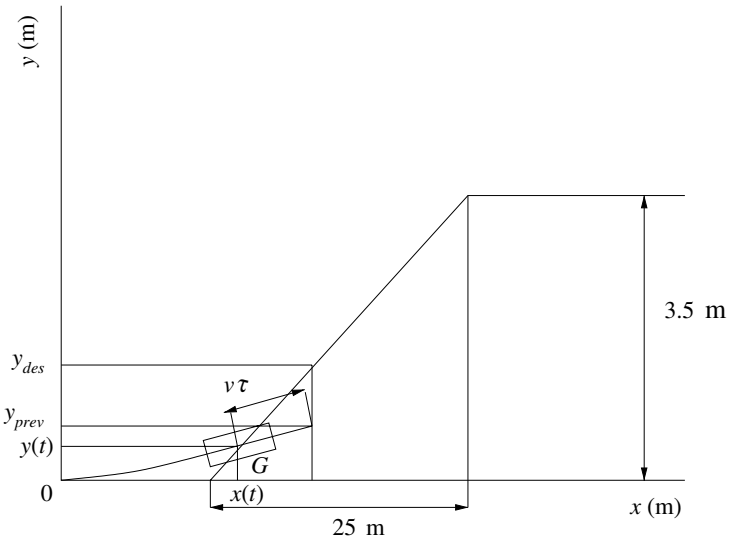


Fig. 8.6. Lane change and driver's control (*des*: desired; *prev*: preview)

The objective function has been constructed on the basis of (1) the maximum distance of the c.g. from the reference path d_i (2) the sum of sideslip angles β_j at different times (Fig. 8.7). The objective function reads

$$L_i = \sum_{j=1}^{k} d_j^2(t_j) + q_L \sum_{j=1}^{k} \beta_j^2(t_j)$$

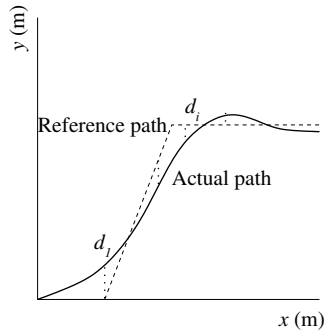


Fig. 8.7. Lane change. Distances d_i of the actual path from the reference path

8.2 Numerical Application

An application of the design method presented is shown. It has been developed with the only aim to illustrate how the designer can perform his task of developing integrated controls. Two iterations will be performed to achieve a satisfactory integration between active chassis controls.

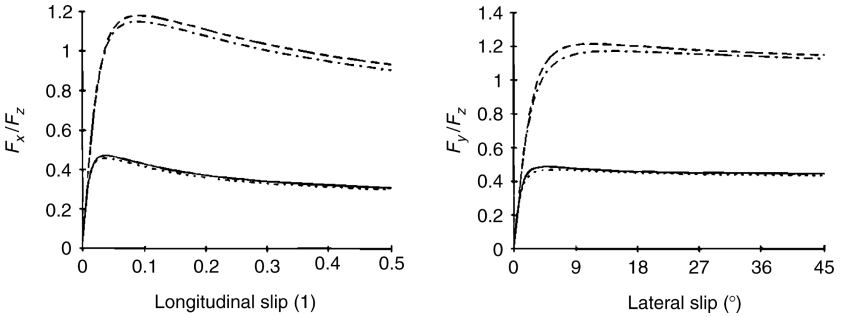


Fig. 8.8. Tyre characteristics of front and rear tyres on dry and wet roads. These characteristics have been employed for the presented numerical simulations

In Fig. 8.9, the scheme adopted for the design of an integrated active chassis control is presented. It is inspired both by the strategy presented in [144] and by the study reported in [143].

8.2.1 First Iteration

Stage 1: Innovation (First Iteration)

At this level design innovation is demanded. As a reference design, that of a common production automobile may be introduced.

$$\mathbf{f}^{(r)} = \{f_1(\mathbf{x}^{(r)}), f_2(\mathbf{x}^{(r)}), \ldots, f_{n_{of}}(\mathbf{x}^{(r)})\}$$

Alternatively, the output design of a previous optimisation can be taken into account to start a new iteration.

$$\mathbf{f}^{(i+1)} = \{f_1(\mathbf{x}^{(i+1)}), f_2(\mathbf{x}^{(i+1)}), \ldots, f_{n_{of}}(\mathbf{x}^{(i+1)})\}$$

The designer is free to propose new design issues on the basis of his own reasoning and cleverness. There is not any limitation to the kind of controller to be adopted. It can be based either on classical control schemes or on other schemes, e.g. [260, 265].

A high performance rear-wheel drive production automobile is taken into consideration (main parameters in Table 8.2). A significant improvement of

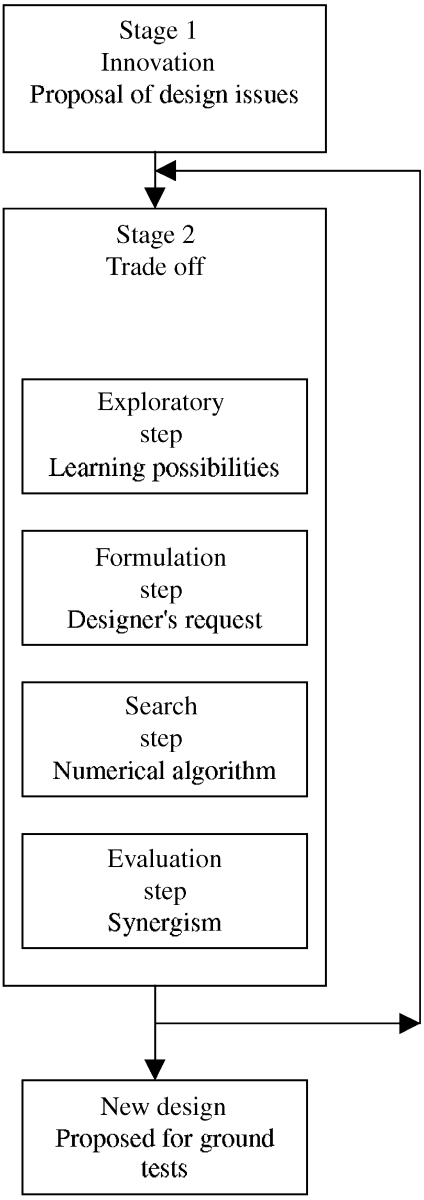


Fig. 8.9. A strategy for the design of an integrated active chassis control

Table 8.2. Main design variable values used for the presented numerical simulations

Design variable		Value
l_F	Relative distance front axle – c.g.	1.42 m
l_R	Relative distance rear axle – c.g.	1.03 m
c_F/c_R	Front/rear halftrack	0.77 m/0.78 m
d_w	Distance of c.g. from wind pressure centre	0.125 m
F_w	Wind force	1,000 N
h	c.g. height	0.4 m
$J_{wF/R}$	Wheel polar moment of inertia	2 kg m^2
$J_x/J_y/J_z$	Chassis principal moments of inertia	378, 1713, 1751 kg m^2
k_{sF}, k_{sR}	Front/rear suspension stiffness	Passive car 43,000/39,000 N/m
k_{rF}/k_{rR}	Front/rear anti-roll bar stiffness	Passive car 30,000/22,000 Nm/rad
k_{wF}/k_{wR}	Tyre vertical stiffness	330,000/350,000 N/m
r_F/r_R	Front/rear suspension damping	Passive car 7,750/14,500 Ns/m
m	Unsprung mass	1,600 kg
m_F/m_R	Front/rear wheel mass	35/45 kg
ϱ	Front/rear wheel radius	0.300 m/0.312 m

Table 8.3. Active chassis controls to be integrated

#	Active chassis control	Symbol
1	Active suspensions (sky-hook damper)	A-SUSP-SH
2	Active suspensions (active anti-roll bar, i.e. roll moment distribution control)	A-SUSP-RM
3	Active four-wheel steering	A-4WS
4	Traction control (controlled differential)	TC-CD

ride and handling is proposed to be achieved by the adoption of four active chassis controls (Table 8.3).

The integration of these four active controls is the task of the designer. The performances of the vehicle have to be assessed according to the driving situations reported in Table 8.1.

Initial Design of Active Chassis Controls (First Iteration)

The following initial design schemes of each active chassis control have been adopted. They are subjected to be changed during iterations between Stage 1 (Innovation) and Stage 2 (Trade-off) (Fig. 8.9).

– *Active suspensions (sky-hook damper).* In parallel with the spring and the damper, at each wheel an ideal actuator is fitted [25, 130] (see Sect. 5.3). The actuator characteristic is

$$F_{act} = G_1(z_w - z_b) + G_2\dot{z}_b$$

By a proper combination of the linear damping characteristic of the actuator and the non-linear damping characteristic of the conventional damper, an effective tuning of suspension design variables can be obtained in particular driving situations [49]. It seems that a limited number of state variables could be used to control an active suspension system with satisfactory performances [250].

The design variables G_i have to be tuned for integration with other controls.

– *Active suspensions (roll moment distribution control).* An active anti-roll bar is fitted at the front axle of the vehicle. Such a device (see, e.g. [130,213]) is able to vary the roll moment distribution between the front and the rear axle. The following control scheme is proposed (similar, but simpler to that introduced in [264])

$$R_m = R_{m0} - K_R\dot{\psi}(\dot{\psi}^\star - \dot{\psi})$$

where R_{m0} is a reference value for the ratio between the front roll moment R_F and the total roll moment front and rear $R_F + R_R$, and R_m is the same ratio at time t

$$R_m = \frac{R_F}{R_F + R_R}$$

A simple expression for the reference yaw velocity $\dot{\psi}^\star$ is (the symbols are reported in Fig. 8.10)

$$\dot{\psi}^\star = v\frac{\delta}{l}$$

which corresponds to the kinematical yaw velocity, i.e. the yaw velocity of a vehicle with equal lateral slips at the front and rear axles.

This kind of control is known as *model following.*

The design variable K_R has to be tuned for integration with other controls.

– *Active four-wheel steering (A-4WS).* A possibly effective control law for rear wheel steering is

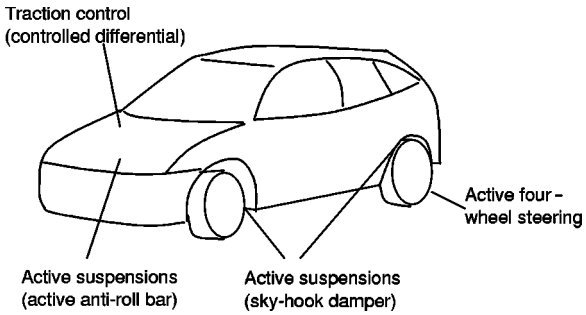


Fig. 8.10. The four active chassis controls to be integrated

$$\delta_R = -K_S(\dot{\psi}^\star - \dot{\psi})$$

The rear wheel steering angle δ_R is proportional to the difference between the reference yaw velocity $\dot{\psi}^\star$ and the actual yaw velocity $\dot{\psi}$. This is, again, a model following control scheme. The introduced control law does not guarantee a negligible c.g. sideslip (as achieved by other control schemes [75, 227, 264]), but it has the great advantage that the implementation of the control is very simple; actually, only the forward speed v, the front steering angle δ and the wheelbase l are used. On this concept the traction control scheme will be developed too (see, e.g. [50]). The design variable K_S has to be tuned for integration with other controls.

– *Traction control (controlled differential).* The influence of traction forces on handling and stability of automobiles has been dealt with in [50, 138, 142, 228, 264]. The proposed control law is

$$\Delta T = T_o - T_i = K_{dfr}(\dot{\psi}^\star - \dot{\psi})$$

where ΔT (the difference of the two torques at the inner and outer wheel) acts in order to make the vehicle achieve the requested yaw velocity $\dot{\psi}^\star$. The design variable K_{dfr} has to be tuned for integration with other controls.

The design variable to be tuned are collected in Table 8.4. The ranges have not been set too wide because the dynamic behaviour of a new vehicle has not to be very different to that of the reference vehicle. This is strongly requested in order to avoid the driver to adapt, while driving, to a completely different vehicle with respect to conventional ones.

Stage 2: Trade-off (First Iteration)

At this level, a trade-off among conflicting requirements is performed. This is the key part of the method. The design variables $\mathbf{x}^*$ satisfying the opposed requests on objectives f_i are the solutions. If the requests cannot be satisfied, new design variable ranges $\mathcal{X}$ or new design issues have to be proposed.

Table 8.4. Design variables to be tuned for the initial design process of the four integrated controls reported in Table 8.3. The allowed ranges of design variable variations are reported too. Data of the reference car in Table 8.2. (One additional design variable will be added in the final design, see Table 8.5).

Design variables to be tuned	Symbols	ranges
susp. stiffness (front)	k_{sF}	20–50 kN/m
susp. stiffness (rear)	k_{sR}	20–50 kN/m
susp. damping (front)	r_F	1–15 kNs/m
susp. damping (rear)	r_R	1–15 kNs/m
active susp. sky-hook design variables (front and rear)	G_{1F}	0.1–10 kNs/m
	G_{1R}	−1–5 kNs/m
	G_{2F}	0.1–10 kNs/m
	G_{1R}	−1–5 kNs/m
active susp. roll moment distribution control design variable	K_R	0.1–5 s^2/rad^2
four wheel steering design variable	K_S	0.1–2 s
traction control design variable (controlled differential)	K_{dfr}	0.5–2 Nms

Table 8.5. Additional design variable to be tuned for the second (and final) iteration of the presented design process of the four integrated controls reported in Table 8.3. The other design variables are reported in Table 8.4. The allowed range of variation of this design variable is reported too.

Design variable to be tuned	Symbols	ranges
second traction control design variable (controlled differential)	K_{dfr2}	0.5–2kNms

In the *exploratory step*, the designer acquires information on the possibilities that may come from the current control scheme, i.e. he learns how much can be achieved in terms of the values of the objective functions. Either a structured or a non-structured procedure may be followed to investigate the problem under consideration. An example of unstructured investigation will be given in Sect. 8.2.1. To perform a structured investigation, the following procedure may be followed.

(a) The n_{of} objective functions are optimised (minimised) separately. $\mathbf{x}^{(p)}$ is the solution that minimises the p–objective function

$$\mathbf{x}^{(p)} = \{x_1^{(p)}, x_2^{(p)}, \ldots, x_{n_{dv}}^{(p)}\} \quad p = 1, 2, \ldots, n_{of}$$

(b) The following matrix is computed which collects the values of the objective functions related to each previous minimisation (*matrix of separate optimisations*)

$$\begin{array}{lll} \min f_1 & \Rightarrow & p=1 \\ \min f_2 & \Rightarrow & p=2 \\ & \vdots & \\ \min f_{n_{of}} & \Rightarrow & p=n_{of} \end{array} \quad \begin{pmatrix} f_1(\mathbf{x}^1) & f_2(\mathbf{x}^1) & \cdots & f_{n_{of}}(\mathbf{x}^1) \\ f_1(\mathbf{x}^2) & f_2(\mathbf{x}^2) & \cdots & f_{n_{of}}(\mathbf{x}^2) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(\mathbf{x}^{n_{of}}) & f_2(\mathbf{x}^{n_{of}}) & \cdots & f_{n_{of}}(\mathbf{x}^{n_{of}}) \end{pmatrix}$$

This matrix can be analysed in terms of correlations between indexes. Spearman's rank–order correlation coefficient (see Sect. 3.3.3) can be employed. The following *matrix of correlation coefficients* can be constructed (see Sect. 3.3.3).

In the *formulation step* the multi-objective problem is converted into a single objective-constrained problem (scalarisation by *constraints method*, see Sect. 3.4.10).

The constraints ε_p have to be set within the extrema $M^{(p)}$ and $m^{(p)}$ that are extracted from the p–column of the matrix. The number of intervals n_v suitable for reconstructing the Pareto-optimal space or for getting a sufficiently refined result is set:

$$\varepsilon_p = m^{(p)} + \left(\frac{t}{n_v - 1}\right)(M^{(p)} - m^{(p)})$$
$$t = 0, 1, 2, \ldots, (n_v - 1);\ p = 1, 2, \ldots, n_{of}$$

The constrained problem is solved for a suitable combination of constraints ε_p. The constraints may be chosen according to the results of the exploration performed in the previous step.

In the *search step* numerical techniques are adopted to find efficient solutions. The numerical techniques for solving this kind of optimisation problem are dealt with in Part I. The numerical examples that are presented in the next section have been obtained by the numerical method SUMT [11] (see Sect. 3.4.7).

In the *evaluation step*, the concept of *synergism*, which is fundamental for the design of integrated controls in the automotive field [148], is introduced. Let us consider the variation of the ith objective function of a vehicle fitted with a number of active controls with respect to a reference vehicle

$$\Delta f_i = f_i - f_i^{ref}$$

As objective functions are supposed to be always minimised by new design issues (i.e. $\Delta f_i < 0$), by definition, the target of the synergism between n_{ac} active controls may be expressed as

$$\Delta f_i^{(n_{ac})} < \sum_{j=1}^{n_{ac}} \Delta f_i^{(j)} \tag{8.1}$$

$$\forall i = 1, \ldots, n_{of} \tag{8.2}$$

where n_{of}, as before, is the number of objective functions under consideration. $f_i^{(j)}$represents the ith objective function of a vehicle fitted with one active chassis control (the jth one). Alternatively, one may require, at least, to have an improvement extended over the whole set of objective functions

$$\sum_{i=1}^{n_{of}} q_i \Delta f_i^{(n_{ac})} < \sum_{i=1}^{n_{of}} q_i \sum_{j=1}^{n_{ac}} \Delta f_i^{(j)} \tag{8.3}$$

where q_i are suitable weights. This requirement cannot be satisfied when physical limitations hold (e.g. approaching tyre-ground adhesion limit). Unfortunately, this is always the case.

Let us consider the number s of subsets composed of the combination of $n_{ac} - 1$ elements, extracted from a set of n_{ac} active chassis controls

$$s = \frac{n_{ac}!}{(n_{ac}-1)!} = \frac{n_{ac}(n_{ac}-1)\cdots(1)}{(n_{ac}-1)(n_{ac}-2)\cdots(1)} = n_{ac}$$

If $n_{ac} = 4$ then $s = 4$. Let us consider this subset, and the element with the minimum value among the other elements

$$\Delta f_{i,min}^{(n_{ac}-1)} = \min\{f_1^{(n_{ac}-1)}, f_2^{(n_{ac}-1)}, \ldots, f_{n_{of}}^{(n_{ac}-1)}\}$$

This minimum variation (with respect to a reference objective function) represents the best performance (related to the ith objective function) that can be obtained by a proper combination of $n_{ac} - 1$ active chassis controls. Usually, one finds

$$\Delta f_{i,min}^{(n_{ac}+1)} \simeq \Delta f_{i,min}^{(n_{ac})} \simeq \Delta f_{i,min}^{(n_{ac}-1)} \simeq \cdots \simeq \Delta f_{i,min}^{(k)} < f_{i,min}^{(k-1)} < \cdots < f_{i,min}^{ref}$$

If this happens for a relevant number of the n_{of} objective functions, obviously, there is no need to adopt additional $n_{ac} - k$ controls, and the number k of controls is sufficient. In any case the minimum requirement on synergism must be

$$\Delta f_i^{(n_{ac})} < \Delta f_{i,min}^{(n_{ac}-1)}$$
$$\forall i = 1, \ldots, n_{of}$$

or at least

$$\sum_{i=1}^{n_{of}} \Delta f_i^{(n_{ac})} < \sum_{i=1}^{n_{of}} q_i \Delta f_{i,min}^{(n_{ac}-1)}$$
$$(\Delta f_i^{(n_{ac})} < \Delta f_{i,min}^{(n_{ac}-1)},\ \forall i = 1, \ldots, n_{of})$$

where q_i are suitable weights. These last minimum requirements are consistent with the scalarisation proposed above. At this stage an analysis of control

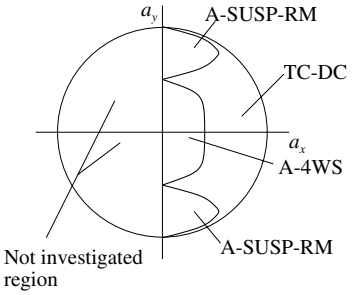


Fig. 8.11. The ranges in which each control is particularly effective are shown in terms of longitudinal and lateral accelerations of the centre of gravity of the vehicle (symbols in Table 8.3). The combinations of longitudinal and lateral accelerations that have been considered in this study are indicated

strategies from an MOP point of view can be conducted in order to have an insight to the addressed problem of integrated controls.

As an example of a *non-structured investigation*, the dynamic performances obtainable from the adoption of each active chassis control have been studied, with reference to the prescribed driving situations of Table 8.1. From this preliminary analysis the following conclusions have been drawn (Fig. 8.11). All controls – but active suspensions (sky-hook damper) – seem to have slight influence on vehicle dynamic performances on rough road. Active suspensions (roll moment distribution control) seem almost ineffective during manoeuvres at low lateral acceleration level. On the contrary, at high lateral acceleration level this control seems quite effective both on dry and wet road. Active four-wheel steering (A-4WS) seems to provide good dynamic performances on dry road. On wet road, particularly during step steering input, the proposed control may cause a spin. This is due to the implemented control scheme which, trying to reach the kinematical yaw rate, makes the vehicle rotate around its centre of gravity instead of around the centre of the bend. Traction control by a controlled differential has given good simulated results at each lateral acceleration level on dry road, yet it revealed itself quite ineffective on wet road. This is due to the fact that the requested reference yaw velocity $\dot{\psi}^{\star}$ is too high, and consequently, the torque difference ΔT between the half-axles is too high.

By following a *structured* research, the matrix of separate optimisations and the corresponding matrix of correlation coefficients (see Sect. 3.3.3) have been computed. Strong conflicts between lane-change manoeuvre and power-on on μ–split appeared. The dynamic behaviour was not satisfactory because complete spin occurred during steering-wheel ramp-input on the wet road, even when this manoeuvre was being optimised.

The reason for this bad dynamic behaviour was mainly the traction control scheme which was not adequately effective. So a new scheme for the traction control had to be studied, and a new iteration was needed and performed.

8.2.2 Second (and Final) Iteration

Stage 1. Innovation (Second Iteration)

Final Design of Active Chassis Controls (Second Iteration)

Referring to the analysis in the previous section the following traction control scheme is proposed to be adopted

$$\Delta T = T_o - T_i = K_{dfr}(\dot{\psi}^\star - \dot{\psi}) - K_{dfr2}(\Omega_o - \Omega_i)$$

where, as before, ΔT refers to the difference of the two torques at the inner and outer wheels. On dry road the second term

$$K_{dfr2}(\Omega_o - \Omega_i)$$

vanishes with respect to the first one. Actually, the difference of the angular speeds of the two wheels is small. On wet road, the two wheels may have considerably different angular speeds and this makes the second term comparable and possibly greater with respect to the first one. This implies that the torque at the inner wheel may be greater than the torque at the outer wheel (the controlled differential works as a common limited slip differential).

The design variable K_{dfr2} has to be tuned for integration with the other controls.

Stage 2: Trade-off (Second Iteration)

Exploratory Step (Second Iteration)

The matrix of separate optimisations (Sect. 3.3.3) is reported in Table 8.6. From the analysis of this matrix, considerable enhancements of the objective functions that are separately minimised can be observed. A conventional value (10^4) has been inserted whenever instability (e.g. spin) occurred. In Table 8.7 the values of the design variables which refer to the separate optimisations are reported.

The matrix of correlations (Sect. 3.3.3) is reported in Table 8.8. The Spearman's rank–order correlation coefficient has been reported for each combination of two objective functions. The *significant* relationships have been marked with a bold script letters. The *significance* of the correlations has been verified properly (see Sect. 3.3.3).

Surprisingly, the discomfort D and the standard deviations of dynamical wheel loads (i.e. road holding) $\sigma_{Fz,F}$ and $\sigma_{Fz,R}$, are correlated ($r_{s12} > 0, r_{s13} > 0$), instead of being anti-correlated ($r_{s12} < 0, r_{s13} < 0$). This is due to the fact that the stiffness and the damping rates have been set in the neighbourhood of the reference passive car, which has very stiff suspensions. Thus, in this case, by minimising discomfort, road holding is improved too.

Table 8.6. Matrix of separate optimisations

	$\min D$	$\min \sigma_{Fz,F}$	$\min \sigma_{Fz,R}$	$\min \sigma_{ws,F}$	$\min \sigma_{ws,R}$	$\min J_1$	$\min J_2$
D	2.87E+01	4.33E+01	2.87E+01	3.12E+01	5.02E+01	7.48E+01	4.83E+01
$\sigma_{Fz,F}$	1.02E+03	1.00E+03	1.02E+03	1.00E+03	1.29E+03	1.25E+03	1.38E+03
$\sigma_{Fz,R}$	1.16E+03	1.51E+03	1.16E+03	1.19E+03	1.45E+03	2.24E+03	1.17E+03
$\sigma_{ws,F}$	4.38E-03	5.90E-03	4.38E-03	3.83E-03	5.00E-03	5.14E-03	5.10E-03
$\sigma_{ws,R}$	6.70E-03	4.22E-03	6.70E-03	7.68E-03	3.73E-03	5.04E-03	5.76E-03
J_1	1.62E+01	4.37E+01	1.62E+01	6.19E+01	4.33E+01	8.13E+00	1.74E+01
J_2	5.90E+01	6.10E+01	5.90E+01	6.20E+01	5.90E+01	6.40E+01	5.30E+01
P_1	2.79E-01	4.06E-01	2.79E-01	1.25E+00	3.51E-01	3.00E-01	4.13E-01
P_2	2.46E+00	2.94E+00	2.46E+00	1.00E+04	2.21E+00	7.79E+00	2.00E+00
W	9.91E-02	1.00E-01	9.91E-02	9.01E-02	9.80E-02	1.04E-01	1 .06E-01
P_μ	4.32E-02	6.87E-03	4.32E-02	1.93E-01	3.25E-02	4.48E-02	7.55E-02
L_1	6.58E+00	6.63E+00	6.58E+00	5.64E+00	6.98E+00	6.57E+00	6.43E+00
L_2	1.62E+01	1.74E+01	1.62E+01	1.21E+01	1.70E+01	1.57E+01	1.68E+01
	$\min P_1$	$\min P_2$	$\min W$	$\min P_\mu$	$\min L_1$	$\min L_2$	
D	2.87E+01	4.33E+01	7.72E+01	5.42E+01	4.78E+01	3.12E+01	
$\sigma_{Fz,F}$	1.02E+03	1.23E+03	1.73E+03	1.58E+03	1.00E+03	1.00E+03	
$\sigma_{Fz,R}$	1.16E+03	1.24E+03	2.04E+03	1.18E+03	1.51E+03	1.19E+03	
$\sigma_{ws,F}$	4.38E-03	4.79E-03	5.38E-03	5.14E-03	9.53E-03	3.83E-03	
$\sigma_{ws,R}$	6.70E-03	4.96E-03	4.83E-03	5.41E-03	5.87E-03	7.68E-03	
J_1	1.62E+01	1.03E+01	1.00E+04	1.65E+01	8.73E+01	6.19E+01	
J_2	5.90E+01	5.70E+01	1.00E+04	7.50E+01	8.10E+01	6.20E+01	
P_1	2.79E-01	7.48E-01	1.80E+00	1.00E+04	1.00E+04	1.25E+00	
P_2	2.46E+00	1.01E+00	1.00E+04	5.94E+00	1.59E+01	1.00E+04	
W	9.91E-02	8.00E-02	5.89E-02	7.62E-02	1.19E-01	9.01 E-02	
P_μ	4.32E-02	1.67E-02	1.28E-02	4.23E-03	1.00E+04	1.93E-01	
L_1	6.58E+00	7.68E+00	8.00E+00	7.94E+00	5.38E+00	5.64E+00	
L_2	1.62E+01	1.77E+01	1.73E+01	1.79E+01	1.29E+01	1.21E+01	

Table 8.7. Design variables corresponding to the separate optimisations

	$\min D$	$\min \sigma_{Fz,F}$	$\min \sigma_{Fz,R}$	$\min \sigma_{ws,F}$	$\min \sigma_{ws,R}$	$\min J_1$	$\min J_2$
k_{sF}	4.36E+04	2.46E+04	4.46E+04	2.87E+04	2.24E+04	4.94E+04	4.24E+04
k_{sR}	3.29E+04	2.53E+04	3.34E+04	4.24E+04	3.03E+04	4.73E+04	3.92E+04
r_F	1.12E+04	5.81E+03	3.84E+03	1.47E+04	8.56E+03	8.26E+03	9.67E+03
r_R	3.62E+03	1.21E+04	4.23E+03	2.83E+03	1.46E+04	1.48E+04	5.49E+03
G_{1F}	1.68E+03	3.21E+02	3.00E+03	3.07E+03	1.29E+03	1.22E+03	4.75E+03
G_{1R}	−4.55E+02	−3.62E+02	4.93E+03	−3.04E+02	3.72E+03	3.48E+03	4.25E+03
G_{2F}	3.23E+03	5.88E+03	3.01E+03	3.00E+02	4.07E+03	1.88E+03	4.21E+03
G_{2R}	3.42E+02	1.66E+03	−3.77E+02	3.71E+01	−4.68E+02	4.09E+03	3.78E+02
K_R	3.92E+00	1.21E+00	3.2JE+00	4.71E+00	1.13E+00	5.86E-01	6.47E-01
K_S	8.61E-01	7.73E-01	1.51E-01	1.72E-01	6.88E-01	9.87E-01	1.37E-01
K_{dfr}	4.89E+02	4.96E+02	4.10E+02	3.79E+02	2.38E+02	1.91 E+02	4.92E+02
K_{dfr2}	1.50E+03	9.25E+02	1.72E+03	7.38E+02	8.63E+02	1.36E+03	1.78E+03
	$\min P_1$	$\min P_2$	$\min W$	$\min P_\mu$	$\min L_1$	$\min L_2$	
k_{sF}	4.36E+04	2.95E+04	2.08E+04	2.36E+04	3.98E+04	2.87E+04	
k_{sR}	3.29E+04	2.83E+04	4.51E+04	4.15E+04	2.59E+04	4.24E+04	
r_F	1.12E+04	9.70E+03	1.19E+04	1.18E+04	1.28E+03	1.47E+04	
r_R	3.62E+03	7.71E+03	1.39E+04	6.83E+03	4.78E+03	2.83E+03	
G_{1F}	1.68E+03	4.08E+03	4.51E+03	3.53E+03	1.23E+03	3.07E+03	
G_{1R}	−4.55E+02	3.06E+03	4.82E+03	4.61E+03	7.58E+02	−3.04E+02	
G_{2F}	3.23E+03	4.51E+03	4.29E+02	4.95E+03	3.78E+03	3.00E+02	
G_{2R}	3.42E+02	4.19E+02	3.70E+03	−3.31E+02	4.53E+03	3.71E+01	
K_R	3.92E+00	3.35E+00	8.09E-01	2.46E+00	2.37E+00	4.71E+00	
K_S	8.61E-01	7.96E-01	8.11E-01	1.45E+00	5.78E-01	1.72E-01	
K_{dfr}	4.89E+02	1.70E+02	3.54E+02	2.27E+02	4.2JE+02	3.79E+02	
K_{dfr2}	1.50E+03	8.84E+02	1.67E+03	8.81E+02	1.34E+03	7.38E+02	

Table 8.8. Matrix of Spearman's rank–order correlation coefficients between the objective functions reported in Table 8.1

	D	$\sigma_{Fz,F}$	$\sigma_{Fz,R}$	$\sigma_{ws,F}$	$\sigma_{ws,R}$	J_1	J_2	P_1	P_2	W	P_μ	L_1	L_2
D	1.00	**0.81**	**0.73**	**0.70**	**−0.69**	0.22	0.48	0.53	0.31	−0.18	−0.28	0.35	0.38
$\sigma_{Fz,F}$		1.00	0.45	0.51	−0.33	0.37	0.46	**0.69**	0.38	−0.26	0.03	0.13	0.17
$\sigma_{Fz,R}$			1.00	0.64	−0.61	0.33	0.57	0.45	0.37	0.03	−0.07	0.10	0.06
$\sigma_{ws,F}$				1.00	−0.71	0.22	0.43	0.39	0.10	0.33	−0.33	0.27	0.45
$\sigma_{ws,R}$					1.00	0.03	−0.03	−0.05	0.33	0.02	0.67	**−0.67**	**−0.74**
J_1						1.00	0.53	0.62	0.65	−0.09	0.19	−0.18	−0.19
J_2							1.00	0.65	0.76	−0.19	0.02	−0.02	−0.18
P_1								1.00	0.66	−0.32	0.06	−0.04	0.06
P_2									1.00	−0.12	0.41	−0.41	0.49
W										1.00	0.42	−0.57	0.30
P_μ											1.00	**−0.96**	**−0.91**
L_1												1.00	0.88
L_2													1.00

The bold script letters refer to statistically significant correlations

The objective function $\sigma_{Fz,F}$ is correlated with P_1 ($r_{s28} > 0$). This is due to the fact that the higher is the $\sigma_{Fz,F}$, the higher is the k_{sF}, which means higher front axle lateral slip, which finally means higher P_1.

The objective function $\sigma_{ws,F}$ is anti-correlated with L_1, L_2 ($r_{s5,12} < 0$, $r_{s5,13} < 0$). This depends on the fact that lower the $\sigma_{ws,F}$, the higher is the r_R, which means higher dynamic load transfer at the rear axle, which involves higher lateral slips and finally higher L_1 and L_2. Strong anti-correlation exists between power-on on μ–split and lane-change manoeuvres ($r_{s11,12} < 0, r_{s11,13} < 0$). This is due to the fact that in the first case the vehicle is requested to run straight ahead, in the second case it is requested to negotiate a bend as fast as possible.

Formulation Step (Second Iteration)

The adopted scalarisation (see 1.2.6) is presented in Table 8.9. The discomfort has been chosen for minimisation, while the other objective functions have been constrained to be better than the best ones of cars having three controls ($n_{ac} - 1$ controls). This aligns with the requirement on synergism.

Table 8.9. Scalarisation proposed for the minimisation of the four integrated controls reported in Table 8.3

D	Minimised
$\sigma_{Fz,F}$	
$\sigma_{Fz,R}$	
$\sigma_{ws,F}$	
$\sigma_{ws,R}$	
J_1	
J_2	Constrained
P_1	
P_2	
W	
P_μ	
L_1	
L_2	

Evaluation Step (Second Iteration)

In Table 8.10 the objective function set which comes from the previous numerical search is reported. The corresponding design variable set can be found in Table 8.11. The objective functions of the passive vehicle and the best vehicle with one less control are also reported for providing a comparison

Table 8.10. Objective functions of the car with four active chassis controls as they can be obtained from the presented design method of integrated controls

	A-SUSP-SH, A-SUSP-RM, A-4WS, TC-CD	A-SUSP-SH, A-SUSP-RM, TC-CD	Passive
D	30.6	30.6	40.6
$\sigma_{Fz,F}$	1,005 N	1,050 N	1,050 N
$\sigma_{Fz,R}$	1,160 N	1,170 N	1,430 N
$\sigma_{ws,F}$	0.0058 m	0.0067 m	0.0068 m
$\sigma_{ws,R}$	0.0057 m	0.0066 m	0.0066 m
J_1	9.16	14.0	21.2
J_2	54.0	61.0	61.5
P_1	0.31	0.36	0.89
P_2	2.3	3.3	17.1
W	0.082	0.126	0.130
P_μ	0.12	0.16	0.26
L_1	6.4	7.4	8.7
L_2	17.5	18.3	19.6

These objective functions are compared with those of two other cars (symbols in Tables 8.1 and 8.3)

Table 8.11. Optimised design variable set of the vehicles with active controls that have been considered in Table 8.10

	A-SUSP-SH, A-SUSP-RM, A-4WS,	A-SUSP-SH, A-SUSP-RM, TC-CD
k_{sF}	38,400 N/m	45,500 N/m
k_{sR}	39,300 N/m	43,900 N/m
r_F	5,900 Ns/m	4,100 Ns/m
r_R	5,400 Ns/m	4,000 Ns/m
G_{1F}	5,400 Ns/m	3,600 Ns/m
G_{1R}	5,500 Ns/m	4,200 Ns/m
G_{2F}	100 Ns/m	130 Ns/m
G_{2R}	580 Ns/m	200 Ns/m
K_R	0.64 s^2/rad^2	0.29 s^2/rad^2
K_S	0.95 s	–
K_{dfr}	2,000 Nms/rad	930 Nms/rad
K_{dfr2}	500 Nms/rad	200 Nms/rad

to the reader. The car with four active controls is superior to the other two, but, as physical limits are approached the improvement is decreasing as chassis complication increases. The evaluation of the obtained result is positive, because the search has succeeded in finding a solution consistent with the requirement on synergism.

New Design

A sample of the expected dynamic performances from the adoption on a real car of the designed active chassis integrated controls is shown in Figs. 8.12–8.15 (also see Table 8.12). Additional simulations have been performed to show how much the performances of the car under consideration (i.e. the car with four active chassis controls, see Table 8.3) are improved with respect to those ones of other less complicated cars. Table 8.11 summarises the cars that have been taken into consideration and the associated combination of active chassis controls.

Table 8.12. Combination of active chassis controls associated with five cars whose performances have been presented in Figs. 8.12–8.15

#	Active chassis controls fitted into car	Note
1	A-SUSP-SH, A-SUSP-RM, A-4WS, TC-CD	Four active chassis controls
2	A-SUSP-SH, A-SUSP-RM, TC-CD	Active suspensions and traction control
3	A-SUSP-SH, A-SUSP-RM, A-4WS	Active suspensions and four-wheel steering
4	A-SUSP-SH, A-4WS	Simple active suspensions and four-wheel steering

Acronyms (symbols) are reported in Table 8.3

From the analysis of Fig. 8.12, on dry road, all cars appear to have good response to ramp steering wheel input. As the centre of gravity sideslip is not controlled to zero, all cars have a non-zero β. On wet road, the car without traction control and active suspensions (roll-moment distribution control) is not able to apply the proper vertical and longitudinal forces in order to avoid a spin. The passive car reaches a spin after 3.5 s (not shown in the figure).

Figure 8.13 refers to power-on while steering. On dry road, the car with four active controls shows the minimum oscillations both with respect to $\dot{\psi}$ and to β. On wet road, the differences between the dynamic behaviours of the considered cars are very marked. The passive car becomes strongly oversteering. Only the cars with traction control avoid spin.

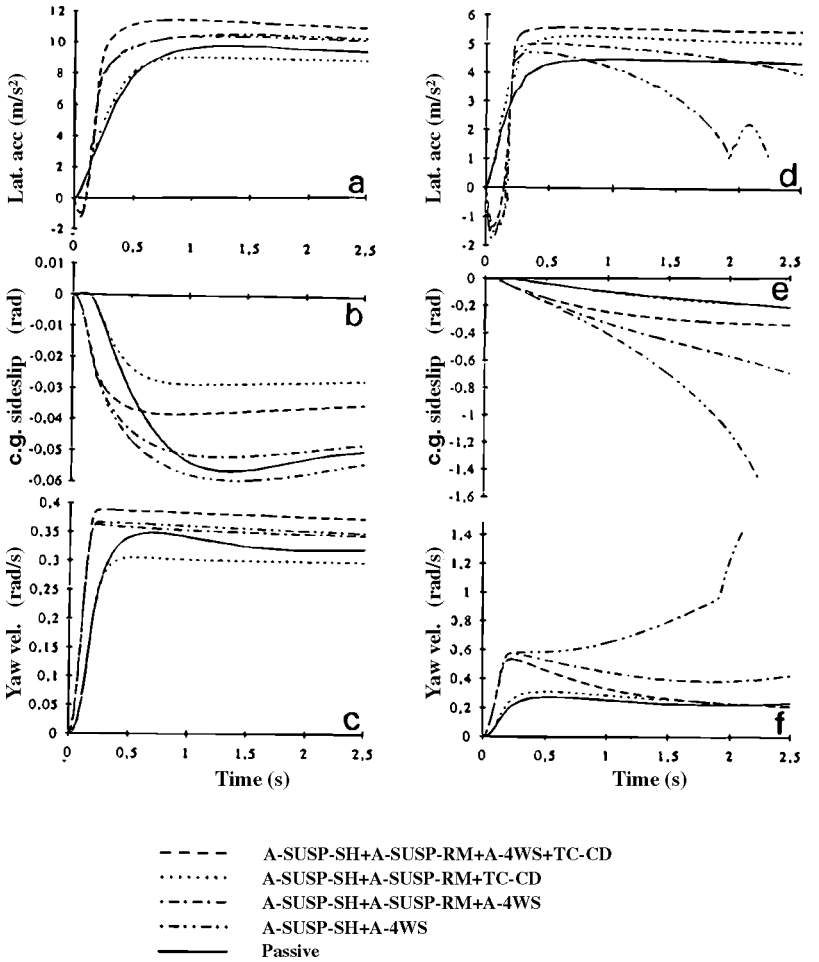


Fig. 8.12. Ramp Steering input on dry road (**a, b, c**) and on wet road (**d, e, f**). Objective functions J_1 and J_2 (Table 8.1). Vehicle seed 30 m/s. Symbols in Fig. 8.10 and in Table 8.3. Description of the manoeuvre in Sect. 8.1.2. Vehicle data in Table 8.2

Figure 8.14 refers to perturbations due to wind gust and μ–split. Traction control is relatively not very effective at low lateral accelerations; on the contrary A-4WS proves to be effective. Only the passive car deviates considerably from the original path.

In Fig. 8.15 the lane-change manoeuvre is shown both for dry and wet roads. Again traction control proves to be not very effective, by itself, to provide quick steering response.

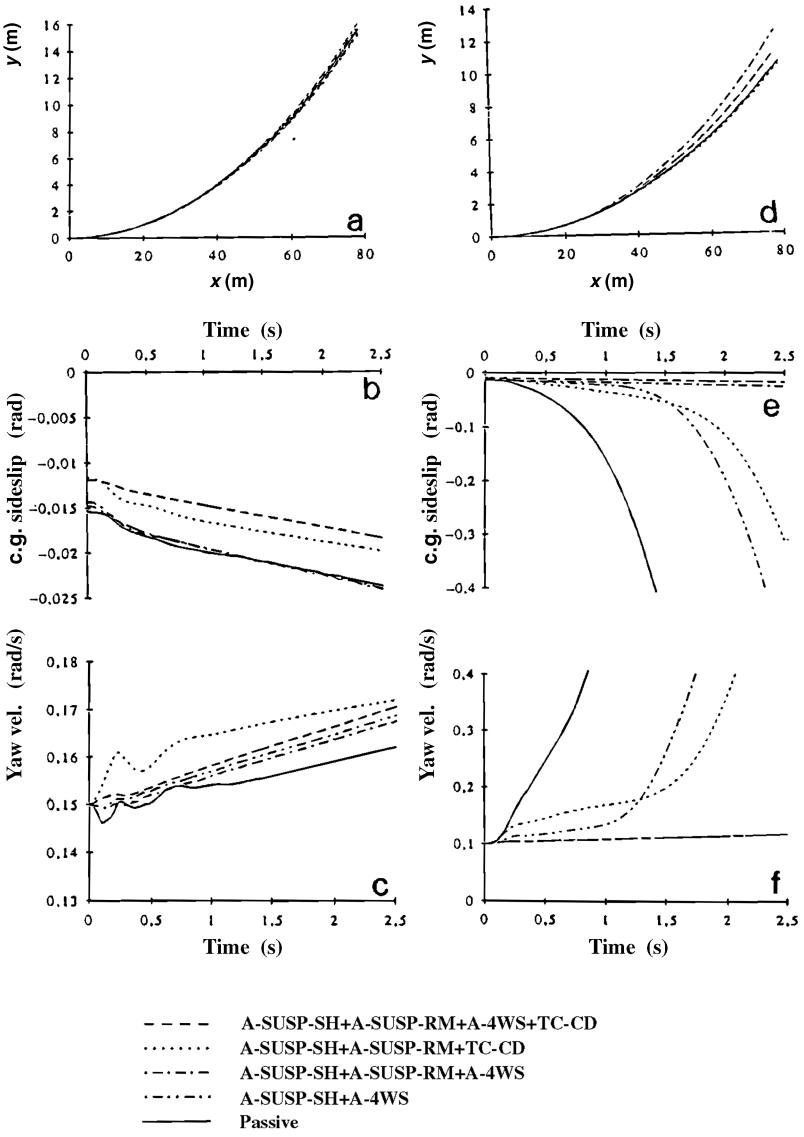


Fig. 8.13. Power-on while steering on dry road (**a, b, c**) and on wet road (**d, e, f**). Objective functions P_1 and P_2. Vehicle speed 30 m/s. Symbols in Fig. 8.10 and in Table 8.3. Description of the manoeuvre in Sect. 8.1.2. Vehicle data in Table 8.2

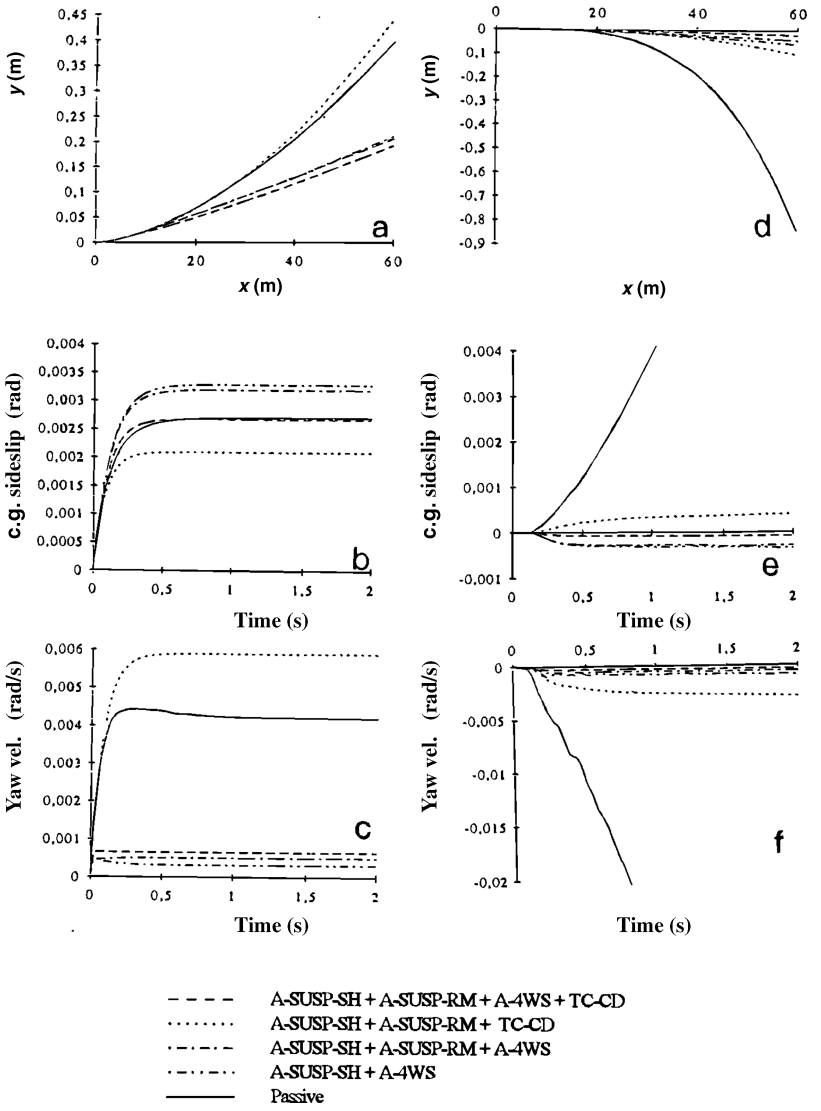


Fig. 8.14. Wind gust on wet road (**a, b, c**). Power-on μ–split (**d, e, f**). Objective functions W and P_μ (Table 8.1). Initial vehicle speed 30 m/s. Symbols in Fig. 8.10 and in Table 8.3. Description of the manoeuvre in Sect. 8.1.2. Vehicle data in Table 8.2

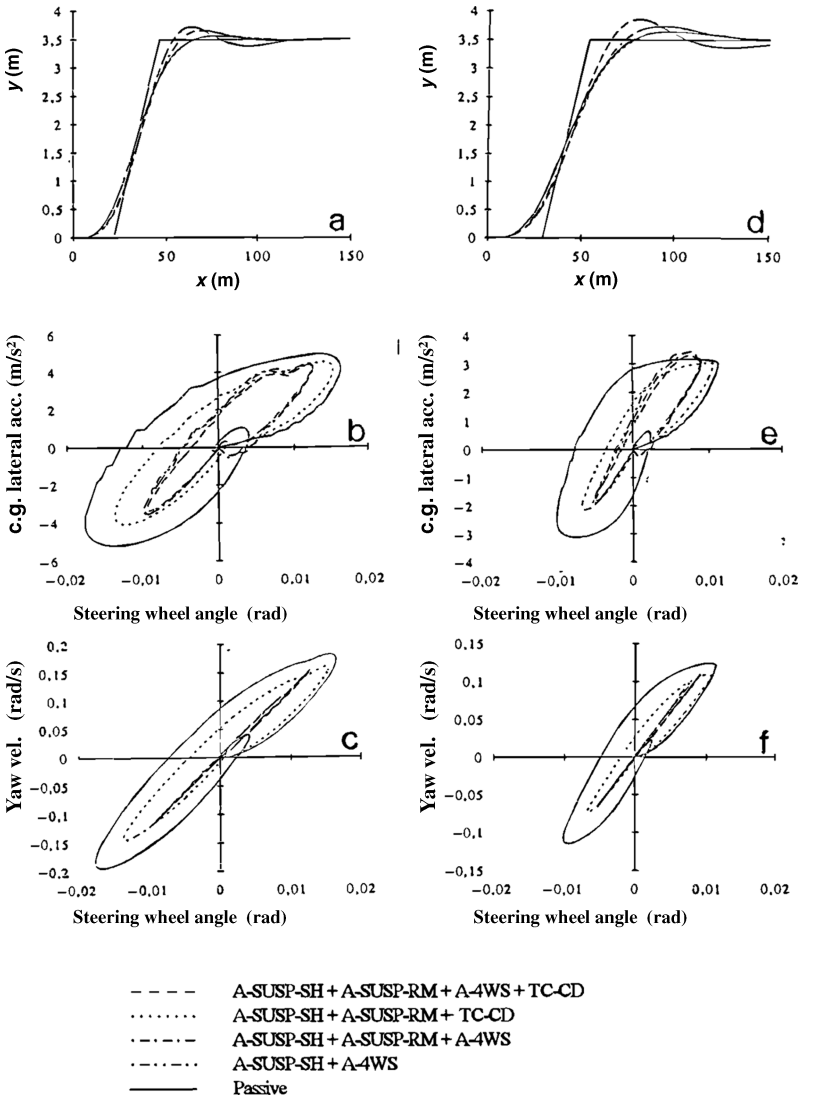


Fig. 8.15. Lane change on dry road (**a, b, c**) and on wet road (**d, e, f**). Objective functions L_1 and L_2 (Table 8.1). Initial vehicle speed 30 m/s. Symbols in Fig. 8.10 and in Table 8.3. Description of the manoeuvre in Sect. 8.1.2. Vehicle data in Table 8.2

8.3 Conclusions

A method for the design of a common control scheme for integrated active chassis systems has been proposed. The method is based on multi-objective optimisation (see Chap. 3). The designer is proposed to iterate between an innovation stage and a trade-off stage, in order to assess the effectiveness of candidate design issues for the synthesis of new control schemes. The method has been applied in order to improve the ride and the handling of an automobile fitted with four active suspensions (sky-hook dampers, active anti-roll bars), active four-wheel steering, traction control (controlled differential). By minimising the discomfort index on rough road (ride), the design variables of the active suspensions and the gains of the other controlled systems have been determined, provided that other ride and handling performances were kept within user-defined limits. A new control scheme has been proposed which is particularly effective during power-on manoeuvres on curved and wet roads. According to numerical simulations, this new control scheme provides excellent dynamic performances in almost every driving situation.

9

Optimal Design of a Double-Cone Synchroniser

The aim of the present chapter is to introduce and apply a method to design a synchroniser of a road vehicle gearbox in order to improve shiftability and driver comfort. In the literature, a few papers exist which deal with gearshift system design [2,115,116,177,185,234,267]. In [2] some performance equations are derived for multi-cone synchronisers. In [177] a number of shiftability indices are introduced and discussed in order to improve the shiftability of a manual transmission. In [185] different types of synchronisers (single cone and multi-cone) are compared in terms of shift-operating forces.

This chapter, derived from [95], is based on previous research studies in which a physical multi-body model of a synchroniser and selector mechanism [54, 115] has been developed by using the software ADAMS. An experimental validation of the physical model has been performed with satisfactory results [54]. A reference gearshift event has been defined and simulated and a number of objective functions have been defined showing a good correlation with subjective shiftability ratings [54, 177].

A multi-objective programming approach has been followed (see Sect. 2.10) to perform the synchroniser optimal design. This approach is particularly suited to deal with the optimisation (i.e. design) of complex mechanical systems, such as synchronisers. According to the multi-objective programming approach, given the (validated) physical model of the synchroniser, the designer has to define the values of many design variables (geometric dimensions, number of teeth and spring stiffness, see Figs. 9.1 and 9.3), in order to improve a number of chosen objective functions. Constraints on design variables are referred to shift events, teeth and cone geometry and pre-synchronisation mechanism. The optimal design problem and its mathematical formulation are rather involved and clearly require a special approach to find a feasible solution.

The optimisation has been performed according to a global approximation procedure (see Chap. 4); the original physical model of the synchroniser system (a complex multi-body ADAMS model) has been substituted by another purely mathematical model based on an artificial neural network (multi-layer

perceptron and radial basis function neural networks). This purely mathematical model is able to approximate the relationship between design variables and objective functions approximately as the original model based on physical laws would do. The computational time for evaluating the objective functions is extremely reduced if an artificial neural network is used instead of the original multi-body model based on physical laws.

A global sensitivity method has been applied in order to analyse the influence of each design variable on each objective function (see Chap. 3).

By means of the artificial neural network many simulations have been performed by varying the design variables. Subsequently, from the simulation set, the so-called Pareto-optimal solutions have been selected. These optimal design solutions (representing the best compromise among objective functions) have been checked for robustness by applying the minimum sensitivity method. By this way Pareto-optimal and robust design solutions have been found and their performances are discussed.

A peculiar aspect of the considered optimisation problem is the presence of many design constraints. To solve this special problem the original design variable domain has been modified according to the constraints prior to the global approximation step, to find the actual feasible design variable domain.

9.1 Synchroniser System Model

Gearbox synchroniser functions can be split into three phases: synchronising, synchroniser indexing, and gear indexing. Referring to Fig. 9.1, gearshift is accomplished by synchronising the rotational speed of the transmission output shaft with that of the idle gear to be engaged. Synchronising is performed by using the cone(s) torque. Synchroniser indexing is the phase in which the sleeve indexes the outer ring after the completion of the synchronisation (engagement of sleeve teeth and outer ring teeth).

Gear indexing is the engagement of the sleeve with the clutch gear teeth.

9.1.1 Physical Model

The multi-body physical model of the synchroniser has been implemented in ADAMS [54].
The model is composed of three subsystems:

1. Selector mechanism (internal to the gearbox)
2. Geartrain
3. Synchroniser

The selector mechanism includes input gearbox lever, command shaft and fork-shaft. The neutral detent mechanism acting on the command shaft is modeled by considering the spring stiffness and pres-load, and the sphere/cavity

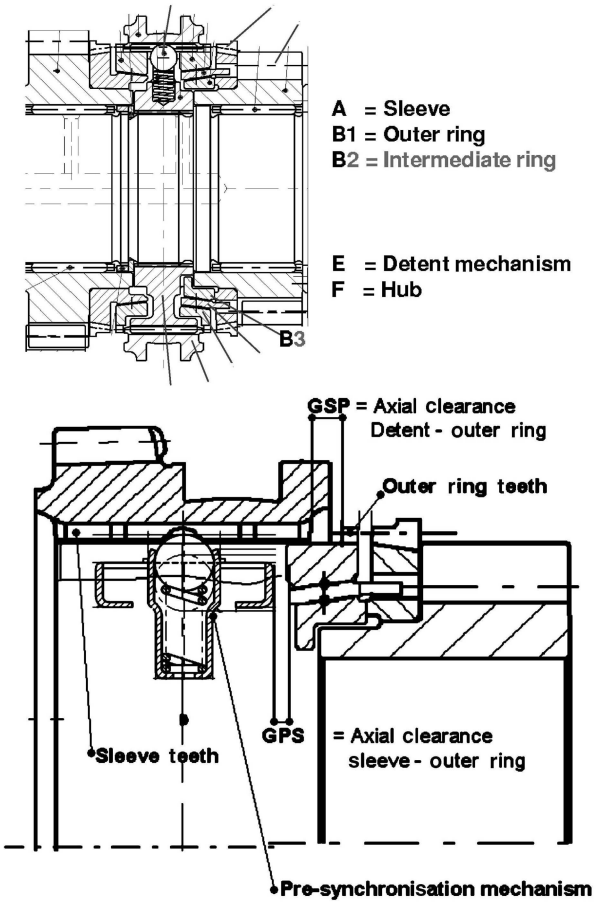


Fig. 9.1. Double-cone synchroniser (Borg–Warner)

geometry. The fork deflection and the fork-sleeve contact stiffness are taken into account.

The geartrain is composed of clutch disk, primary shaft and secondary shaft, gear pairs and differential gear. Plain bearings rolling and hydraulic losses are taken into account.

A double-cone Borg–Warner type synchroniser [181] has been considered (Fig. 9.1). It consists of the following components:

1. Hub
2. Sleeve
3. External ring
4. Intermediate ring
5. Internal ring
6. Clutch gear

The model can simulate the behaviour of the detent mechanism, the contact forces on the sleeve, outer ring and engagement teeth, as well as jamming on the synchroniser cones. Extensive experimental tests have been performed to validate the model [54]. The results of the comparison between numerical simulations and experimental data are shown in Fig. 9.2.

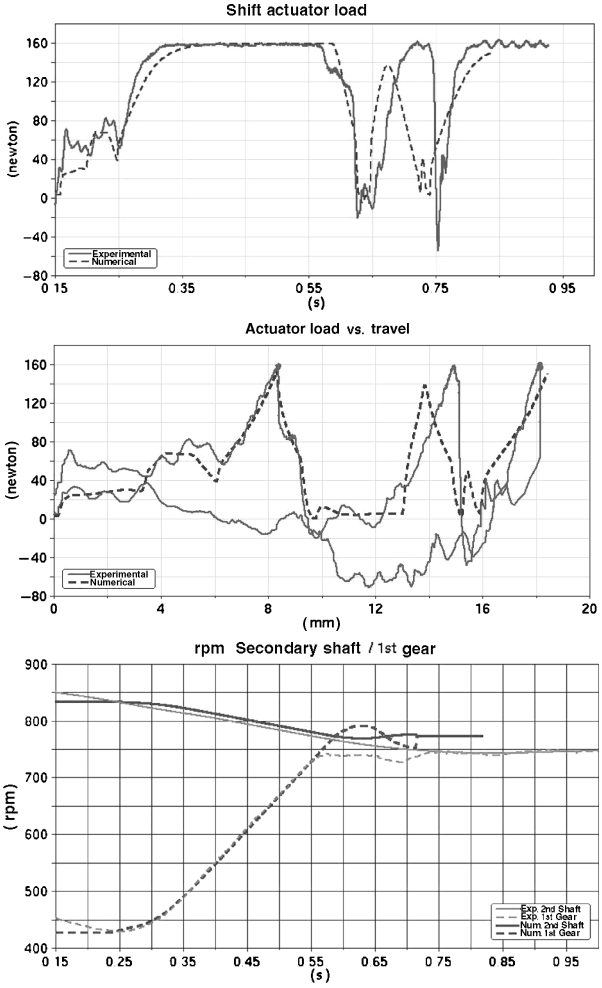


Fig. 9.2. Comparison of simulated and experimental test data. Gearshift manoeuvre from second to first gear. Initial equivalent vehicle speed 20 km/h. The maximum shift effort of actuator has been constrained to 160 N

9.2 Formulation of the Design Problem for the Optimisation of a Synchroniser

9.2.1 Design Variables

The design variables defining the synchroniser system have been divided into three subsets:

- Independent design variables (V): parameters that have to be varied (optimised).
- Parameters (P): parameters that have not to be varied in this optimisation (but could be varied for future applications).
- Dependent design variables (VP): parameters that have to be varied (optimised); these are functions of the others independent design variables.

A total of 32 design variables have been considered: 19 independent design variables, 8 parameters and 4 dependent design variables.

The 32 design variables pertaining to the synchroniser model under consideration are described in Table 9.1. In Fig. 9.3 the chamfer teeth geometry and pre-synchronisation mechanism are shown to better understand the meaning of design variables and parameters.

9.2.2 Objective Functions

A gearshift manoeuvre from second to first gear is considered. External selector mechanism is not taken into account, and a force actuator acts on the input gearbox lever. The initial equivalent vehicle speed is 20 km/h. The

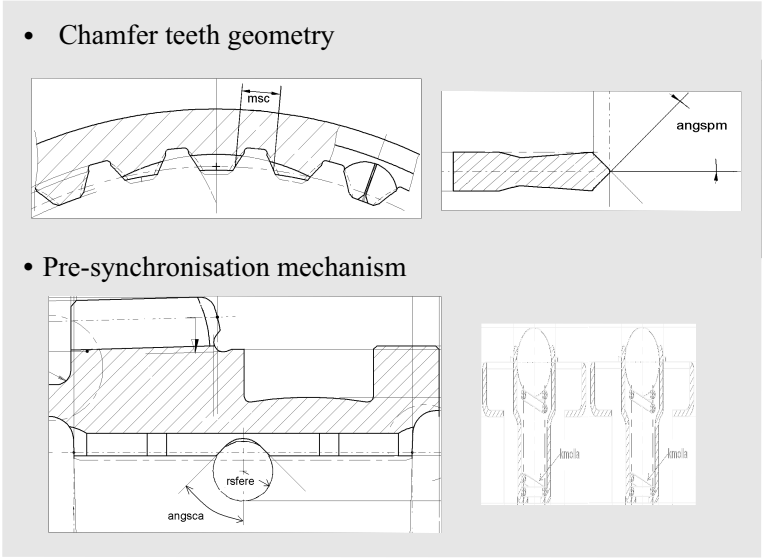


Fig. 9.3. Chamfer teeth geometry and pre-synchronisation mechanism [134]

Table 9.1. Design variables pertaining to the synchroniser under optimisation (V: independent design variable; P: parameter; VD: dependent design variable)

	Name	Type	Lower/upper bounds
Neutral detent			
Axial clearance detent – outer ring	GPS	V	−70, +140
Axial clearance outer ring – sleeve	GSP	V	−70, +30
Axial clearance outer ring – clutch teeth	$tiro$	V	−60, +200
Stroke – End	$corsam$	P	
Blocking-indexing			
Pre-synchronisation sphere radius	$rsfere$	P	
Number of pre-synchronisation spheres	$nsfere$	P	
Pre-synchronisation spring stiffness	$kmolla$	V	−50, +90
Pre-synchronisation spring pre-load	$p0molla$	V	−50,+300
Cavity slope	$angsca$	V	−50, +10
Cavity depth	$prosca$	V	−50, +140
Sleeve			
Teeth modulus	msc	VD	
Number of teeth	$ndenti$	V	−15, +15
Pitch diameter	Dp	P	
Teeth thickness	$spcdm$	V	−30, +30
Chamfer angle	$angspm$	V	−20, +20
Overlap (rotational clearance hub-outer ring)	$ricoprf$	VD	
Outer ring			
Teeth thickness	$spcds$	V	−30, +40
Outer ring chamfer angle	$angsps$	V	−15, +25
Outer ring teeth axial length	sas	P	
Outer ring cone angle	$angCONO1$	V	−15, +30
Outer ring cone mean diameter	$dANE1_c$	V	−7 +7
Outer ring cone axial length	$LANE1_c$	V	−25, +20
Equivalent dynamic friction coeff. (outer/intermediate rings)	$coedin1$	P	
Equivalent static friction coeff. (outer/intermediate rings)	$coesta1$	P	
Intermediate ring			
Intermediate ring cone angle	$angCONO2$	P	
Intermediate ring mean diameter	$dANE2_c$	V	−7, +7
Intermediate ring axial length	$LANE2_c$	V	−20, +30
Equivalent dynamic friction coeff. (Inner/intermediate rings)	$coedin2$	VD	
Equivalent static friction coeff. (Inner/intermediate rings)	$coesta2$	VD	
Clutch gear			
Clutch gear teeth thickness	$spcdi$	V	−30, +35
Clutch gear chamfer angle	$angspi$	V	−11, +30
Clutch gear teeth axiallength	$sasi$	P	

Design variables ranges are reported as percentage variation with respect to the reference synchroniser. These bounds define the design variable domain

maximum shift effort of actuator has been constrained to 160 N, with an objective maximum velocity of 225 mm/s.

The fundamental events in the process of gear-shifting [20] are

1. Neutral detent
2. Blocking-indexing (and pre-synchronisation)
3. Synchronisation
4. Blocking release
5. Teeth engagement
6. Full engagement

During phase (1), the fork moves taking off the axial fork-sleeve clearances, the sleeve slides and takes off the detent–outer ring clearance. When the detent mechanism contacts the outer ring, the pre-synchronisation starts (2). The outer ring moves into the groove of the hub, then it takes off rotational clearance (blocking-indexing). Sleeve chamfers act on outer ring chamfers and synchronisation (3) begins. Outer ring teeth 'block' sleeve until the end of synchronisation. At the end of synchronisation the sleeve teeth engage with the outer ring teeth: there is an impact, causing an increment of shift effort ('double bump') with one or more axial load peaks. Before the full engagement is completed the sleeve and clutch gear teeth impact each other.

We can put these events in relation with subjective feelings:

- One of the most important criteria affecting the evaluation of shiftability is the *shift effort* (i.e. the effort required to execute gear changes). Shift effort is determined by the maximum synchronisation load.
- The *smoothness of the shift lever operation* is another important criterion of shiftability. Two factors are mainly responsible for feeling a deterioration in shift smoothness: 'block striking' (i.e. sensation that the shift action stops momentarily at the moment of synchronisation), 'second load peak' (i.e. the feeling that a huge second load peak occurs following synchronisation). The block striking force is evident when the peak force at the beginning of synchronisation is extremely larger than the average load during synchronisation. This index shows relatively good correlation with subjective evaluations [2].

The seven objective functions which have been considered for the optimisation of a synchroniser are reported in Table 9.2.

9.2.3 Constraints

The correct operation of both the synchroniser and the selector mechanism requires 14 geometric and operating constraints to be satisfied.

Some constraints that have been considered are reported in Table 9.3. Inequality 1 guarantees the correct engagement sequences, allowing the completion of the pre-synchronisation event (defined by GPS) before the synchronisation (see also Fig. 9.1). Inequalities 5 and 6 are related to the radial gap

Table 9.2. Objective functions that have been considered for the optimisation of the synchroniser (see Fig. 9.4)

Performance Index	Definition	Target
$F_{max,s}$ (kg)	Maximum synchronisation effort	MIN
$\Delta F_{\%,s} = (F_{max,s} - F_{min,s})/F_{max,s}$ (%)	Difference synchronisation load, minimum load after synchronisation over synchronisation load (%)	MAX
$\Delta F_{,inn} = (F_{max,inn} - F_{min,s})$ (kg)	Difference 'double bump' load peak, minimum load after synchronisation	MIN
$\Delta F_{\%,inn} = (F_{max,inn} - F_{min,s})$ $/(F_{max,s} - F_{min,s})$ %	$\Delta F_{,inn}$ over the difference synchronisation load, minimum load after synchronisation (%)	MIN
$\dot{F}$ (kg/s)	$\dot{F}$ after synchronisation	MIN
$L_{max,s}$ mm	Distance synchronisation travel-stroke end	MIN
$L_{max,inn}$ mm	Distance 'section load' peak travel–Stroke end	MIN

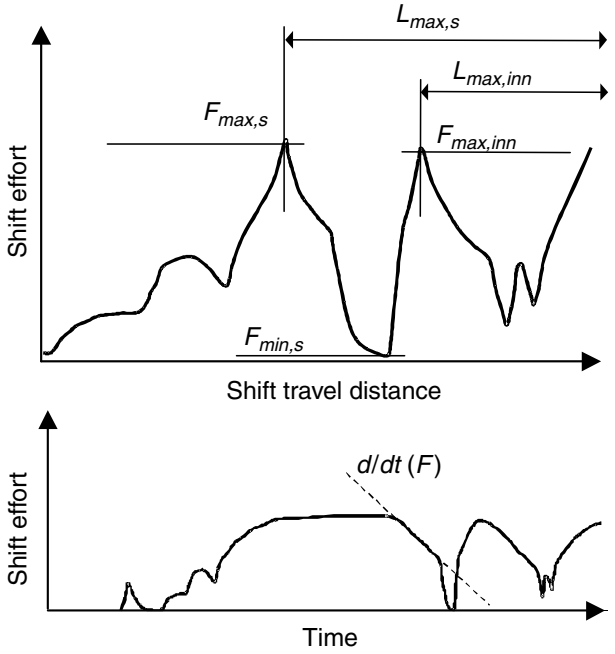


Fig. 9.4. Shift effort as function of the shift travel distance and its time-history

Table 9.3. Constraints as functions of design variables

	Constraints
1	$GPS < GSP$
5	$(spcds + spcdm) < msc \times \pi$
6	$(msc \times \pi - spcds - spcdm) < 0.5$
8	$angsps < angspm$
14	$angspi < angspm$

between sleeve and outer ring teeth. Constraint 8 allows the synchronisation torque to be governed by the chamfer angle of the outer ring. Constraint 14 allows an easier movement of the sleeve chamfers with respect to the clutch gear ones. Other constraints allow the compatibility between axial clearances, between geometric dimensions of the teeth; they also refer to the detent mechanism and to the geometry of the synchronisation cones.

9.3 Method for the Optimal Design of a Synchroniser

The procedure that has been followed to optimise a synchroniser has a sequential form. The steps are listed in Table 9.4.

9.3.1 Feasible Design Variables Domain

A feasible design variable domain has to be found by modifying the original design variable domain. This is due to the fact that, the design variable domain (see, e.g. Table 9.1) does not take into account the design constraints (see, e.g. Table 9.3). After the constraints have been rearranged, the lower and upper bounds of the variation range of the design variables have been redefined in order to be compatible with the introduced constraints. This may require a special study because the constraints might be ranked in order to allow the definition of one design variable at a time.

9.3.2 Global Sensitivity Analysis

Standard sensitivity analysis (see Chap. 3) refers to the change in system response due to small variations of the parameters of the system (partial derivatives of the system responses with respect to the system parameters). This gives the designer a very limited insight to the optimisation problem. Global sensitivity analysis has been performed here (instead of standard sensitivity analysis) because it describes the behaviour of the system when parameters are varied within broad given intervals [220] (e.g. within the whole design variable domain reported in Table 9.1). Referring to our case involving design variables, global sensitivity analysis refers to the measure on how much

Table 9.4. Sequential optimisation method based on global optimisation adopted for the gear synchroniser

Step	subproblem	Method	Result
A	Re-definition of the design variable domain taking into account the constraints	Feasible design variable domain	Modified design variable domain
B	Preliminary investigation for simplifying as much as possible the optimisation problem	Global sensitivity Analysis	Elimination of the objective functions strongly correlated
C	Substitution of the physical model of the synchroniser with a purely mathematical model based on ANNs	global approximation	ANN able to simulate the dynamic behaviour of the synchroniser (very small simulation time)
D	Computation of the Pareto-optimal set	Quasi-Monte carlo Search Multi-objective optimisation	Pareto-optimal set
E	Understanding optimal solutions	Global sensitivity analysis	Insight on optimal design solutions
F	Checking for robustness Pareto-optimal solutions	Synthesis	Robust & optimal solutions

each design variable contributes to build up the value of each performance index. This analysis can help to identify the key design variables whose variation affects mostly the performance indices. Of course, also the less influential design variables are identified, that is, those design variables for which a large tolerance in their specifications can be released.

There are many kinds of global sensitivity analysis. Variance-based methods [220] are the most effective ones in giving, quantitatively, the addressed information to the designer. Unfortunately, in our case, a complete variance analysis could not be completed because the number of design variables is very high and the model requires much time for simulations. In fact variance-based analysis requires a one-factor-at-a-time treatment.

If the monotonicity of the system model response is properly checked and confidently assessed, a Spearman rank correlation analysis (SRCA) can be employed successfully [220] (see Sect. 3.3.3). The rank transformation can cope with non-linear relationships and allows a robust estimation of the global sensitivity, but it underestimates higher order terms. Another advantage of using SRCA is that the coefficient of determination r_s can be used to check the validity of the estimated influence of the design variables. A low value of r_s indicates that the sensitivity analysis is failing (non-monotonic relationship). In

case of existence of non-monotonic relationships, the global sensitivity analysis can be performed anyway after a proper partition of design variables domain. An empirical method, which has been used in the present application, is the partition of the design variables domain into a number of sub-domains and the evaluation of the global sensitivity in each sub-domain. If the normalised importance of each design variable remains constant by varying the number and the hyper-volumes of the sub-domains, it is reasonable to consider a design variable as influencing.

By means of an Artificial Neural Network (see Chap. 4) it is possible to perform global sensitivity [109].

By analysing the interconnection weights of the trained network (w_{ij}) we can get an insight into the influence of each design variable on the performance indices.

9.3.3 Global Approximation

According to the global approximation procedure, the original physical model (e.g. the ADAMS model presented in Sect. 9.1) is used for defining the parameters of a purely mathematical model able to simulate (approximately) the dynamic behaviour of the actual system. The simulations for solving the optimal design problem, i.e. computing the Pareto-optimal set, will be performed by means of the derived purely mathematical model which is very fast in computing the performance indices as functions of design variables.

In this work two artificial neural network algorithms (ANNs) [87, 88, 109], namely, multi-layer perceptron neural networks (MLPNN) and radial basis function neural network (RBFNN) [109, 159, 229] have been used and their performances compared. The MLPNN is the most employed ANN type in the literature. The RBFNN has been developed more recently and it is based on a linear regression process, which makes the training process (definition of parameters) easier (see Chap. 4). The different approximation methods which have been employed are described and compared in Table 9.5.

Table 9.5. Global approximation methods applied for solving the optimal design problem.

Method	Training effort	Accuracy
Polynomial n-dimensional interpolation	Average	Average
Neural network 'Multi-layer perceptron'	Very high	Very good
Neural network 'Radial basis function'	Average/high	Good

Training effort and accuracy are evaluated referring to the considered optimisation problem

For artificial neural networks, the 'training' process requires the determination of all the 'weights' and 'offsets' (see [109]). The ANNs are generally trained by considering [109]the mean-squared error (M_{SE})

$$M_{SE} = \frac{1}{2} \sum_{i=1}^{N_t} \frac{(y_i - \hat{y}_i)^2}{N_t}$$

where N_t is the number of simulations performed by the original physical ADAMS model used for the training of the artificial neural network, $\hat{y}_i$ is the output of the artificial neural network, y_i is the performance index value computed by the original physical ADAMS model. This value will be called 'nominal value'.

Three different types of error metrics are considered by using the N_v simulations that constitute the validation set

Percentage error with respect to the nominal value y_i

$$E_{PN} = \frac{\hat{y}_i - y_i}{y_i}$$

(useful in presence of limited variations of $\hat{y}_i$ with respect to the nominal values y_i).

Percentage error with respect to the variation of y_i

$$E_{PI} = \frac{\hat{y}_i - y_i}{\Delta y_i}$$

(useful in presence of large variations Δy_i due to small variation of design variables).

Correlation coefficient between y_i and $\hat{y}_i$. A linear regression (ranked) between $\hat{y}_i$ and d_i is performed, and the coefficient of determination R can be used as error metric.

$$R = \frac{\sum_{i=1}^{N_v} \left(\hat{y}_i - \hat{\bar{y}}\right)(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N_v} \left(\hat{y}_i - \hat{\bar{y}}\right)^2 \sum_{i=1}^{N_v} (y_i - \bar{y})^2}}$$

where $\hat{\bar{y}} = \frac{1}{N_v} \sum_{i=1}^{N_v} \hat{y}_i$ and $\bar{y} = \frac{1}{N_v} \sum_{i=1}^{N_v} y_i$

9.3.4 Quasi-Monte Carlo Search

To find the Pareto-optimal set (i.e. the optimal design solutions) many simulations are needed. This can be performed quickly by means of a purely mathematical model (e.g. a mathematical model based on ANNs). The

strategy that can be followed is computing by simulation a huge number of performance indices as function of a huge number of design variables combinations selected from the feasible design domain. The implemented search algorithm is a Quasi-Monte Carlo algorithm [246] based on low discrepancy sequences [154] (see Chap. 3). This process generates a set of design solutions containing a sub-set composed of the Pareto-optimal solutions.

9.3.5 Multi-Objective Optimisation

The set of design solutions generated has to be processed in order to extract from this set the Pareto-optimal design solutions. To perform this selection, a multi-objective programming problem has to be solved. The multi-objective programming problem optimisation of a synchroniser can be formulated (see Sect. 2.10). Reference will be made to our particular case (synchroniser defined in Sects. 9.1 and 9.2).

The Pareto-optimal solutions pertaining to the synchroniser are defined as follows. Given the minimisation problem (10.1) with seven performance indices and 32 design variables, a solution $\mathbf{x}_i$ is Pareto-optimal if it is non-dominated by applying Definition 2.7. This definition can be used to find directly the Pareto-optimal design variables combinations x_i [87, 88, 159].

9.3.6 Robust Design and Synthesis of Optimal Design Solutions

Pareto-optimal design solutions must be checked for robustness. Robust design requires to minimise the effects on performance indices of uncontrollable variations of design variables (of any kind, the design variables can be dependent, parameters and independent). These uncontrollable variations are typically related to the loading conditions, tolerances, temperature changes, etc.

An optimal solution which minimises the performance indexes may be unacceptable if the variance (due to design variable uncontrollable variation) is large.

The minimum sensitivity method is a straightforward approach for achieving robustness [3]. No probability distributions related to the uncontrollable variation of design variables are needed and it can be proved that minor sensitivity implies greater robustness. Relying on minimum sensitivity method, a standard sensitivity analysis has been performed, referring to each optimal design solution. High sensitivity involves poor robustness.
A 'robustness index' defined as the mean of the normalised sensitivity indices will be used.

When additional statistical information are available about the variations of the design variables, it is possible to obtain a more accurate estimate of the 'level of robustness' of the optimal design solution under study.

By imposing that the uncertain design variables ($x_i \quad i = 1, \ldots, n_{dv}$) have a Gaussian distribution, the covariance matrix of the objective functions $\mathbf{C}_{FOBS}$ can be computed as

$$\mathbf{C}_{FOBS}(\mathbf{x}) = \mathbf{J}(\mathbf{x})\mathbf{C}_{\mathbf{x}}\mathbf{J}^T(\mathbf{x})$$

where $\mathbf{C}_{\mathbf{x}}$ is the covariance matrix of the uncertain design variables and $\mathbf{J}(\mathbf{x})$ is the Jacobian matrix of the partial derivatives of the performance indices with respect to the design variables vector. The reference design variables vector $\mathbf{x}$ is obviously one of the Pareto-optimal design solutions.

9.4 Optimal Design of a Synchroniser

The optimisation of the reference production synchroniser defined in Table 9.1 has been performed according to the method in Sect. 9.3 (Table 9.4).

Step A. The feasible design variable domain has been determined. A special study had to be performed to take into account the constraints (Table 9.3).
Step B. About 1,000 simulations have been performed by means of the physical ADAMS multi-body model described in Sect. 9.1. The design variable vectors have been generated by using a Sobol uniformly distributed sequence (see Sect. 3.4.2). The Global Sensitivity Analysis has been performed on the 1,000 simulation output data. The relationships between design variables and performance indices, and between performance indices and performance indices have been highlighted. In Table 9.6 the most influencing design variable on each performance index is shown. It is emphasised a strong influence on performance indices of the design variables related to the pre-synchronisation mechanism. It is also shown that $F_{max,s}$ (see Table 9.2) seems to have only a weak dependence on the selected design variables. This aspect will bring difficulties during the global approximation phase.

Important correlations between performance indices have not been found.

Table 9.6. Global sensitivity analysis

Parameter	Performance index	r_s	R
GSP	$L_{max,s}$	−0.992	0.99
tiro	$L_{max,inn}$	−0.362	0.51
p0molla	$\dot{F}$	−0.497	0.70
prosca	$\Delta F_{\%,s}$	−0.497	0.72
prosca	$\Delta F_{,inn}$	−0.635	0.81
prosca	$\Delta F_{\%,inn}$	−0.403	0.67
angsps	$F_{max,s}$	−0.615	0.46

Step C. In order to allow an effective optimisation within a reasonable computation time, the physical ADAMS model is substituted by a purely numerical mathematical model. A study is presented for assessing the accuracy of the approximation obtainable by different artificial neural networks. The results are summarised in Table 9.7. The training process of both artificial neural networks (see Chap. 4) has been performed by means of the 1,000 simulations (training set) used for the global sensitivity analysis presented in Sect. 3.3. The approximation errors have been computed on a 'validation set' constituted by 200 solutions not belonging to the training set.

By inspection of Table 9.7, it is evident that the performance index $F_{max,s}$ cannot be approximated with a sufficient accuracy. This is due to the fact that the maximum force is limited in the ADAMS model by means of a feedback controller, so it is almost independent from all the design variables as already shown in the previous sections.

Table 9.7 shows that the performances of the MLPNN artificial neural network described in Table 9.8 are evidently superior.

Table 9.7. Artifical neural networks(MLPNN and RBFNN) approximation errors on the seven performance indices

MLPNN	$F_{max,s}$	$\Delta F_{\%,s}$	$\Delta F_{,inn}$	$\Delta F_{\%,inn}$	$\dot{F}$	$L_{max,s}$	$L_{max,inn}$
Mean ($\|\|E_{PN}\|\|$) *100	0.6	1.6	2.8	2.5	9.8	0.9	6.7
Mean ($\|\|E_{PI}\|\|$)) *100	5.8	2.7	3.2	3.9	5.1	1.9	4.1
R 'ranked'	0.61	0.96	0.97	0.96	0.96	0.98	0.91
RBFNN	$F_{max,s}$	$\Delta F_{\%,s}$	$\Delta F_{,inn}$	$\Delta F_{\%,inn}$	$\dot{F}$	$L_{max,s}$	$L_{max,inn}$
Mean ($\|\|E_{PN}\|\|$)) *100	0.9	6.6	9.5	5.9	16.9	4.9	14.0
Mean ($\|\|E_{PI}\|\|$) *100	9.0	11.2	10.9	9.2	10.1	10.3	9.1
R 'ranked'	0.28	0.84	0.89	0.77	0.80	0.79	0.63

Table 9.8. Back-propagation multi-layer perceptron neural network

Training set	1° hidden layer (# of neurons)	2° hidden layer (# of neurons)	Output layer (# of neurons)
1200	70	80	7

By employing the adopted artificial neural network the simulation time was reduced by about 1,000 times with respect to the same simulation performed by the physical ADAMS model.
Steps D, E, F. A large number (nearly 10^6) of design solutions were generated by the artificial neural network in a short time. Optimal solutions could be obtained by resorting to the Pareto-optimal solution definition.

The solutions which improve all the performance indexes have been considered. The performance index $F_{max,s}$, which is independent from all the design variables in the ADAMS model, will not be taken into account in the following.

Ten Pareto-optimal solutions have been selected (Table 9.9). The solutions numbered from 1 to 6 are the ones with the maximum improvement on each performance index. The solutions numbered from 7 to 10 are the best according to the designer preferences.

Table 9.9. Variation of the performance indices of ten Pareto-optimal solution with respect to the reference configuration

	$\%\Delta F_{\%,s}$	$\Delta F_{,inn}$	$\Delta F_{\%,inn}$	$\dot{F}$	$L_{max,s}$	$L_{max,inn}$	Robustness
1	−2.5	−20	−17	−48	0	−31	2.2
2	1	−34	−26	−42	−1	−36	4.9
3	1	−31	−32	−51	−1	−32	2.3
4	1	−17	−16	−62	−1	−21	1.4
5	1	−9	−8	−17	−10	−8	1.3
6	−2	−19	−21	−16	−4	−55	1.3
7	0	−20	−21	−29	−6	−29	1.1
8	1	−29	−29	−44	−4	−27	1.4
9	1	−33	−31	−47	−6	−45	2.1
10	−2	−27	−22	−47	−4	−44	2.9

The average robustness index must be as close as possible to 1

The final selection of the preferred optimal and robust design solution depends upon the designer which acts as a decision maker. The solution indicated with number 7 is the one with the highest level of robustness, and anyway it gives very important improvements on all the performance indices (average >17%). The performance of this optimal design solution can be compared with that of the reference synchroniser in Figs. 9.5 and 9.6.

9.5 Conclusion

A new procedure for designing a double-cone synchroniser (together with selector mechanism) has been proposed and applied.

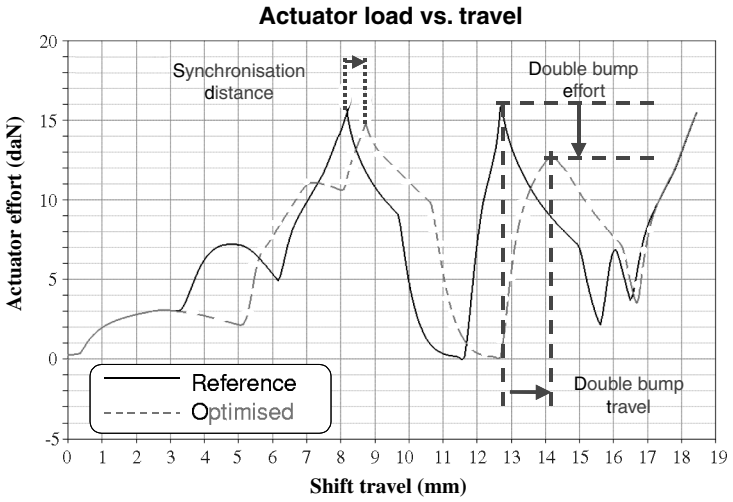


Fig. 9.5. Shift effort as function of the shift travel distance for the reference and optimised solution number 7

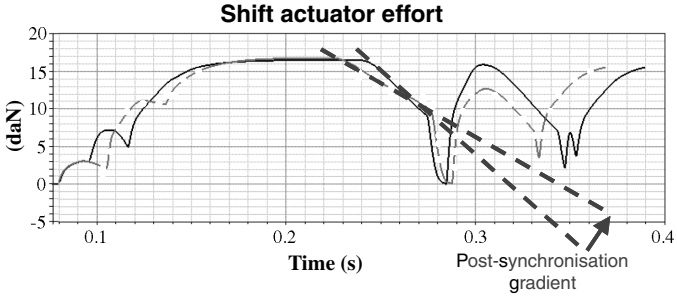


Fig. 9.6. Time-history of shift actuator effort for the reference and optimised solution number 7

A multi-objective programming approach has been followed. A global approximation procedure has been adopted to simulate quickly the dynamic behaviour of the synchroniser. According to the global approximation procedure, the physical model of the system (a complex multi-body ADAMS model) has been substituted by another purely mathematical model based on an artificial neural network, able to approximate at the desired extent the relationship between design variables and performance indices. The method is an improved version of the one described in [87].

Very effective and reliable metrics have been introduced in order to evaluate the performance of two different artificial neural network types. For the considered optimisation problem, the multi-layer perceptron neural network trained by means of the back-propagation algorithm seems clearly the best approximation model.

A peculiar aspect of the considered optimisation problem is the presence of many design constraints. This has requested a special study to define the actual feasible domain for the design variables.

A global sensitivity method has been applied in order to analyse the influence of the design variables on each performance index. This has allowed to detect a strong influence on performance indices of the design variables related to the pre-synchronisation mechanism.

A gearshift event has been simulated by varying 32 design variables and considering seven performance indices which have shown a good correlation with subjective shiftability ratings.

Pareto-optimal solutions have been computed within an acceptable time. These solutions have been checked in terms of robustness by applying the minimum sensitivity method. Finally, some optimal and robust design solution have been found.

The dynamic behaviour of a preferred optimal solution has been shown in time domain. The efficiency of the proposed procedure is clear and the adopted design method can be used for further studies and actual engineering design activity. The prosecution of this research project will involve the reformulation of the problem to a multi-objective stochastic framework.

10

Optimal Design of the Suspension System of Railway Vehicles

In the present chapter the dynamic response of rail vehicles due to random excitation is dealt with by deriving very simple *analytical formulae*. The optimisation of railway vehicle suspension system is proposed by using multi-objective programming. Basic hints are given on how to select the railway vehicle suspension design variables to obtain the best trade-off between standard deviation of vertical acceleration at the body and standard deviation of secondary suspension stroke.

In the literature the authors have not found papers dealing with the problem of deriving simple analytical formulae for the estimation of the dynamic response of railway vehicles to random excitations generated by vertical track irregularity. With reference to vehicle–bogie vibrations and to vehicle–track interaction, a number of authors have dealt with the problem of deriving basic concepts useful for rail vehicle design [60, 76, 99, 199]. They usually have resorted to numerical simulations even when dealing with simple models.

Successful applications of multi-objective programming in the field of railway vehicle design are reported in [51, 147, 151].

In the first section of the chapter analytical formulae are derived [151] to describe the dynamic behaviour of a railway vehicle. These formulae are then used in the second section of the chapter to derive the best trade-off between the standard deviations of body acceleration and secondary suspension stroke [151].

10.1 System Model

10.1.1 Equations of Motion and Responses to Stochastic Excitation

The adopted system model is shown in Fig. 10.1. The system has two degrees of freedom. The mass m_1 represents one-fourth of the mass of the bogie

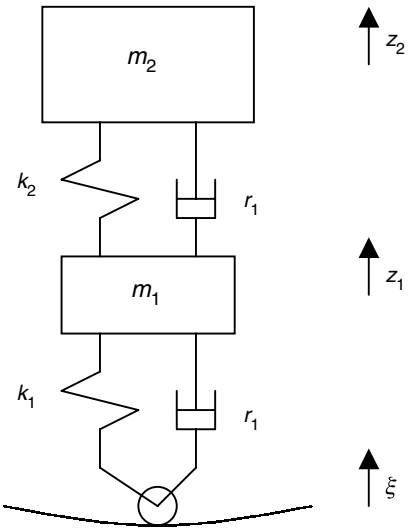


Fig. 10.1. Railway vehicle system model

and the mass m_2 represents one-eighth of the mass of the body. The excitation comes from the displacement (ξ) which represents the motion of the axle-box, the track being regarded as an uneven and infinitely stiff structure. This hypothesis had to be forcedly introduced to preserve both the analytical formulation and the analytical solution to the problem. In Sect. 9.2.3 the accuracy of the response of the model will be discussed.

The equations of motion may be written in matrix form as

$$\mathbf{M}\,\ddot{\mathbf{z}} + \mathbf{R}\,\dot{\mathbf{z}} + \mathbf{K}\,\mathbf{z} = \mathbf{F} \tag{10.1}$$

$\mathbf{M}$, $\mathbf{R}$, $\mathbf{K}$ are the mass, damping and stiffness matrixes, respectively

$$\mathbf{M} = \begin{pmatrix} m_1 & 0 \\ 0 & m_2 \end{pmatrix} \tag{10.2}$$

$$\mathbf{R} = \begin{pmatrix} r_1 + r_2 & -r_2 \\ -r_2 & r_2 \end{pmatrix} \tag{10.3}$$

$$\mathbf{K} = \begin{pmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{pmatrix} \tag{10.4}$$

$\mathbf{F}$ is the vector of external forces related to excitation from the uneven track

$$\mathbf{f} = \begin{pmatrix} r_1\dot{\xi} + k_1\xi \\ 0 \end{pmatrix} \tag{10.5}$$

$\mathbf{z}$ is the vector of the independent variables

$$\mathbf{z} = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} \tag{10.6}$$

The displacement ξ (track irregularity) is assumed to be a random variable defined by a stationary and ergodic stochastic process. This assumption is consistent with the results of the studies, performed by many authors and organisations (see, e.g. [108,188]), on the stochastic properties of track (vertical) irregularity. A number of analytical formulae have been adopted (see [76,199]) to interpolate the measured data referring to the power spectral density (PSD) of the stochastic process defining ξ. In the present chapter two of those analytical formulae have been considered

$$S_\xi(\omega) = \frac{A_b (2\pi\ v)^3}{\omega^4} \tag{10.7}$$

$$S_\xi(\omega) = \frac{A_v \omega_c^2 v}{\omega^2 (\omega^2 + \omega_c^2)} \tag{10.8}$$

In a log–log scaled plot (abscissa ω), the spectrum in Eq. (10.7) (reported in [199]) takes the shape of a line sloped at rate 4. In the following, it will be indicated as *one-slope* power spectral density (1S-PSD). The PSD in Eq. (10.8) has been reported in [76], and is widely used in railway vehicle dynamics simulations. In the log–log scaled plot shown in Fig. 10.2, Eq. (10.8) takes approximately the shape of a *two-slope* curve, thus reference to it will be made by the acronym 2S-PSD.

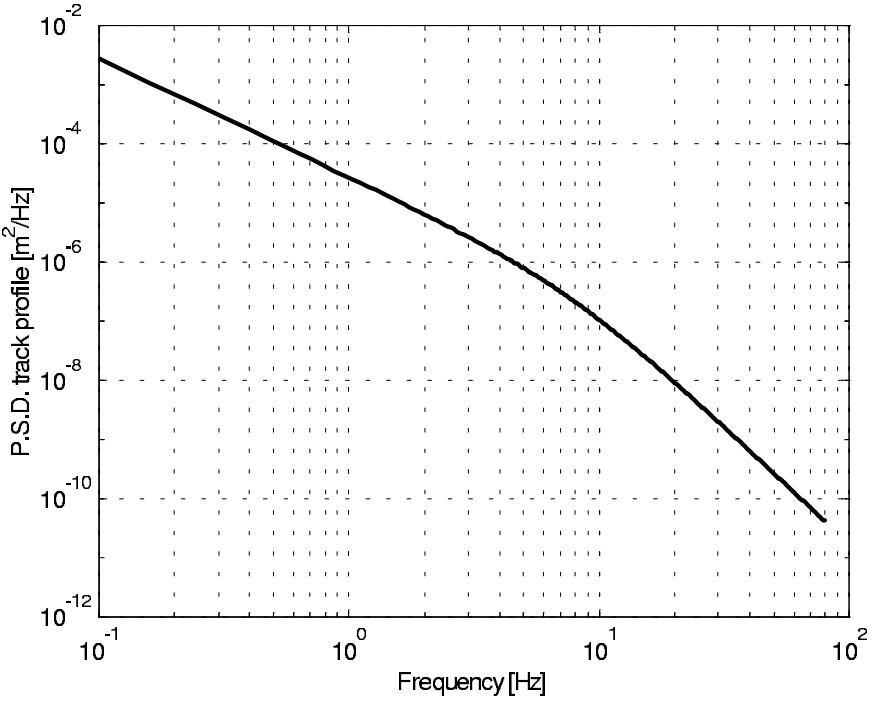


Fig. 10.2. Power spectral density (PSD) of the irregularity of the track in the vertical plane. Two slope PSD (2S-PSD, Eq. 10.8), at 177 km/h, (adapted from Ref. [76]).

The system (10.1) can be rewritten in a more convenient form

$$\begin{aligned} \dot{\underline{\mathbf{z}}} &= \mathbf{A}\,\underline{\mathbf{z}} + \mathbf{B}\,u \\ \mathbf{y} &= \mathbf{C}\,\underline{\mathbf{z}} + \mathbf{D}\,u \end{aligned} \tag{10.9}$$

$\underline{\mathbf{z}}$ is the vector of state variables

$$\underline{\mathbf{z}} = \begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \\ z_1 \\ z_2 \end{pmatrix} \tag{10.10}$$

$\mathbf{A}$ and $\mathbf{B}$ are the state matrixes

$$\mathbf{A} = \begin{pmatrix} -\frac{r_1+r_2}{m_1} & \frac{r_2}{m_1} & -\frac{k_1+k_2}{m_1} & \frac{k_2}{m_1} \\ \frac{r_2}{m_2} & -\frac{r_2}{m_2} & \frac{k_2}{m_2} & -\frac{k_2}{m_2} \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad \mathbf{B} = \begin{pmatrix} 1/m_1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{10.11}$$

u is the input variable, related with track irregularity ξ

$$u = \left(r_1 \dot{\xi} + k_1 \xi \right) \tag{10.12}$$

$\mathbf{y}$ is the vector of output variables

$$\mathbf{y} = \begin{pmatrix} F_z \\ \ddot{z}_2 \\ z_2 - z_1 \end{pmatrix} \tag{10.13}$$

The output variables represent, respectively, the force on the axle-box (F_z), the vertical acceleration of the body ($\ddot{z}_2$), the (vertical) relative displacement body–bogie (stroke of the secondary suspension) ($z_2 - z_1$).

$\mathbf{C}$ and $\mathbf{D}$ matrixes read respectively

$$\mathbf{C} = \begin{pmatrix} -r_1 & 0 & -k_1 & 0 \\ \frac{r_2}{m_2} & -\frac{r_2}{m_2} & \frac{k_2}{m_2} & -\frac{k_2}{m_2} \\ 0 & 0 & -1 & 1 \end{pmatrix} \tag{10.14}$$

$$\mathbf{D} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \tag{10.15}$$

The frequency response of the linear dynamic system (10.9) is

$$\mathbf{H}_{\mathbf{y},u}(s) = \mathbf{C}\,(s\mathbf{I} - \mathbf{A})^{-1}\,\mathbf{B} + \mathbf{D} \tag{10.16}$$

where $\mathbf{I}$ is the identity matrix. The input is represented by u, the output is represented by vector $\mathbf{y}$.

Given the PSD, P_{SDu} of the excitation input, the P_{SDj} of the jth element of vector $\mathbf{y}$, can be computed as (see, e.g. Sect. 5.1.1)

$$P_{SDj}(s) = H_j(s)\ H_j(-s)\ P_{SDu}(s) \tag{10.17}$$

$$\mathbf{H}(s) = \begin{pmatrix} H_1(s) \\ H_2(s) \\ H_3(s) \end{pmatrix} = \begin{pmatrix} H_{F_z,u}(s) \\ G_{\ddot{z}_2,u}(s) \\ G_{z_2-z_1,u}(s) \end{pmatrix} \tag{10.18}$$

$H_j(s)$, being a frequency response function of a mechanical system, is obviously a ratio of two polynomials (of variable s), i.e. $H_j\ (s)$=$N(s)/D(s)$. P_{SDu} and $P_{SD\xi}$ are linked by the following expression:

$$P_{SDu}(s) = L_{u,\xi}(s)\ L_{u,\xi}(-s)\ P_{SD\xi}(s) \tag{10.19}$$

where $L_{u,\xi}(s) = k_1 + s\, r_1$.

In order to write $P_{SDj}(s)$ (Eq. (10.17)) in a form matching with that of following Eq. (10.24) (the reason for this need will be explained in the next section), $P_{SD\xi}(s)$ has to be written as

$$P_{SD\xi}(s) = W_{qS}\ L_{qS}(s)\ L_{qS}(-s), q = 1, 2 \tag{10.20}$$

where for $q = 1$ reference is made to the 1S-PSD, and it follows from Eq. (10.7) that $s = j\omega$ ($j = \sqrt{-1}$), $L_{1S}(s) = 1/s^2$ and $W_{1S} = A_b(2\pi v)^3$; for $q = 2$ reference is made to the 2S-PSD (Eq. (10.8)), so $L_{2S}(s) = 1/\{s(s+\omega_c)\}$ and $W_{2S} = A_v\omega_c^2 v$.

The P_{SDj} of the jth element of vector y can be finally written as

$$P_{SDj}(s) = W_{qS}\ L_{qS}(s)\, L_{u,\xi}(s)\ H_j(s)\ H_j(-s)\, L_{u,\xi}(-s)\, L_{qS}(-s) \tag{10.21}$$

This form is convenient for computations that will be presented in the next section (Eqs. (10.23) and (10.24)).

The (vertical) relative displacement bogie–axle box (stroke of the primary suspension) $(z_1 - \xi)$ and the force on the axle-box are related by the following expression:

$$L_{z_1-\xi,F_z}(s) = -\frac{1}{k_1 + sr_1} = -\frac{1}{L_{u,\xi}(s)}$$

So, the PSD of $(z_1 - \xi)$ reads

$$P_{SD\,z_1-\xi}(s) = W_{qS}L_{qS}(s)\ H_{F_z,u}(s)\, H_{F_z,u}(-s)\, L_{qS}(-s) \tag{10.22}$$

10.1.2 Derivation of Standard Deviations in Analytical Form

By definition (see Sect. 5.1.1) the variance of a random variable described by a stationary and ergodic stochastic process is

$$\sigma_j^2 = \frac{1}{2} \int_{-\infty}^{+\infty} P_{SDj}(s)\ ds \tag{10.23}$$

In [180] it is shown that an analytical solution exists for σ_j^2 if P_{SDj} can be written as

$$P_{SDj} = \frac{N_{k-1}(s)\ N_{k-1}(-s)}{D_k(s)\ D_k(-s)} \tag{10.24}$$

where D_k is a polynomial of degree k, and N_{k-1} is a polynomial of maximum degree $k-1$ ($k \geqslant 1$). This is actually the case, in fact it can be verified that P_{SDj} may be written as in Eq. (10.24) both by inspection of Eqs. (10.21) and (10.22) and by analysing the expressions of H_j and $H_{F_z,u}$.

For example, considering the vertical acceleration of the vehicle body ($\ddot{z}_2$), the following expression for $N_{k-1}(s)/D_k(s)$ can be obtained:

- for the 1S-PSD ($q = 1$, Eqs. (10.20), (10.7)), $k = 4$

$$\frac{N_3(s)}{D_4(s)} = \sqrt{W_{1s}} L_{1S}(s)\ L_{u,\xi}(s)\ H_{\ddot{z}_2,u}(s)$$

where (setting $r_1 = 0$) assuming

$$W_{1S} = A_b (2\pi v)^3, \quad L_{1S}(s) = \frac{1}{s^2}, \quad L_{u,\xi}(s) = k_1,$$

$$H_{\ddot{z}_2,u}(s) = (k_2 + r_2 s)\, s^2 / \big(k_1 k_2 + k_1 r_2 s + k_2 m_1 s^2 + k_1 m_2 s^2 + k_2 m_2 s^2 + m_1 r_2 s^3 + m_2 r_2 s^3 + m_1 m_2 s^4\big)$$

we obtain

$$N_3(s) = (A_b)^{1/2} (2\pi v)^{3/2} (k_1 k_2 + k_1 r_2 s)$$

$$D_4(s) = (k_1 k_2 + k_1 r_2 s + k_2 m_1 s^2 + k_1 m_2 s^2 + k_2 m_2 s^2 + m_1 r_2 s^3 + m_2 r_2 s^3 + m_1 m_2 s^4)$$

- for the 2S-PSD ($q = 2$, Eqs. (10.20), (10.8)), $k = 5$

$$\frac{N_4(s)}{D_5(s)} = \sqrt{W_{2s}} L_{2S}(s)\ L_{u,\xi}(s)\ H_{\ddot{z}_2,u}(s)$$

where (setting $r_1 = 0$) assuming

$$W_{2S} = A_v \omega_c^2 v, \quad L_{2S}(s) = \frac{1}{s(s + \omega_c)}$$

we obtain

$$N_4(s) = (A_v \omega_c^2 v)^{1/2}(k_1 k_2 s + k_1 r_2 s^2)$$

$$D_5(s) = (k_1 k_2 + k_1 r_2 s + k_2 m_1 s^2 + k_1 m_2 s^2 + k_2 m_2 s^2 + m_1 r_2 s^3 + m_2 r_2 s^3 + m_1 m_2 s^4)(s + \omega_c)$$

The analytical formulae presented in the following subsections have been derived by means of the analytical solutions of the integral (10.22) reported in Sect. 5.6.

10.1.3 Complete Formulae Using the 1S-PSD (Eq. (10.7))

The 1S-PSD (Eq. (10.7)) has been considered because it allows a full and compact analytical solution of the problem. For the standard deviations of interest, by solving analytically Eq. (10.23) one gets

$$\sigma_{F_z} = 2\pi^2 v^{3/2} \sqrt{A_b} \sqrt{\frac{n_1}{d_1}} \tag{10.25}$$

$$\begin{aligned} n_1 = {} & r_1^2 m_1 m_2 \left(r_1 r_2^2 k_1 + r_1^2 r_2 k_2 + r_1 k_2^2 m_1 + r_2 k_1^2 m_2 + r_1 k_2^2 m_2\right) + \\ & (m_1 + m_2)\left(r_2 m_1 + r_1 m_2 + r_2 m_2\right)\left[(m_1 + m_2)\left(r_2^2 k_1^2 + r_1^2 k_2^2\right) - \right. \\ & \left. 2 k_1^2 k_2 m_1 m_2\right] + k_1 k_2 (m_1 + m_2)^2 \left[-m_1 m_2 (r_2 k_1 + r_1 k_2) + \right. \\ & \left. (r_2 m_1 + r_1 m_2 + r_2 m_2)(r_1 r_2 + k_2 m_1 + k_1 m_2 + k_2 m_2)\right] + \\ & (r_2 k_1 + r_1 k_2)\left[-2 r_1 (r_2 k_1 + r_1 k_2) m_1 m_2 (m_1 + m_2) + \right. \\ & \left. (r_1 r_2 m_1 + r_1 r_2 m_2 + k_1 m_1 m_2)^2\right] \\ d_1 = {} & \left[- m_1 m_2 (r_2 k_1 + r_1 k_2)^2 - k_1 k_2 (r_2 m_1 + r_1 m_2 + r_2 m_2)^2 + \right. \\ & \left. (r_2 k_1 + r_1 k_2)(r_2 m_1 + r_1 m_2 + r_2 m_2)(r_1 r_2 + k_2 m_1 + k_1 m_2 + k_2 m_2)\right] \end{aligned}$$

$$\sigma_{\ddot{z}_2} = 2\pi^2 v^{3/2} \sqrt{A_b} \sqrt{\frac{n_2}{d_2}} \tag{10.26}$$

$$\begin{aligned} n_2 = {} & r_1^2 r_2^2 (r_2 k_1 + r_1 k_2) + \left(r_2^2 k_1^2 + r_1^2 k_2^2\right)(r_2 m_1 + r_1 m_2 + r_2 m_2) + \\ & k_1 k_2 \left[-m_1 m_2 (r_2 k_1 + r_1 k_2) + (r_2 m_1 + r_1 m_2 + r_2 m_2)(r_1 r_2 + \right. \\ & \left. k_2 m_1 + k_1 m_2 + k_2 m_2)\right] \\ d_2 = {} & \left[-m_1 m_2 (r_2 k_1 + r_1 k_2)^2 - k_1 k_2 (r_2 m_1 + r_1 m_2 + r_2 m_2)^2 + \right. \\ & \left. (r_2 k_1 + r_1 k_2)(r_2 m_1 + r_1 m_2 + r_2 m_2)(r_1 r_2 + k_2 m_1 + k_1 m_2 + k_2 m_2)\right] \end{aligned}$$

$$\sigma_{z_2-z_1} = 2\pi^2 v^{3/2}\sqrt{A_b}\sqrt{m_2^2\frac{n_3}{d_3}} \tag{10.27}$$

$$\begin{aligned} n_3 &= r_1^2 k_2\left(r_2 m_1 + r_1 m_2 + r_2 m_2\right) + k_1\left[-m_1 m_2\left(r_2 k_1 + r_1 k_2\right) + \right. \\ &\quad \left.\left(r_2 m_1 + r_1 m_2 + r_2 m_2\right)\left(r_1 r_2 + k_2 m_1 + k_1 m_2 + k_2 m_2\right)\right] \\ d_3 &= k_2\Big[-m_1 m_2\left(r_2 k_1 + r_1 k_2\right)^2 - k_1 k_2\left(r_2 m_1 + r_1 m_2 + r_2 m_2\right)^2 + \\ &\quad \left(r_2 k_1 + r_1 k_2\right)\left(r_2 m_1 + r_1 m_2 + r_2 m_2\right)\left(r_1 r_2 + k_2 m_1 + k_1 m_2 + k_2 m_2\right)\Big] \end{aligned}$$

$$\sigma_{z_1-\xi} = 2\pi^2 v^{3/2}\sqrt{A_b}\sqrt{\frac{n_4}{d_4}} \tag{10.28}$$

$$\begin{aligned} n_4 &= k_1\left(r_2 k_1 + r_1 k_2\right) m_1^2 m_2^2 + k_1\left(m_2 + m_1\right)\left(r_2 m_1 + r_1 m_2 + \right. \\ &\quad \left. r_2 m_2\right)\left(r_2^2 m_1 + r_2^2 m_2 - 2k_2 m_1 m_2\right) + \\ &\quad + k_2\left(m_1 + m_2\right)^2\left[-\left(r_2 k_1 + r_1 k_2\right) m_1 m_2 + \right. \\ &\quad \left.\left(r_2 m_1 + r_1 m_2 + r_2 m_2\right)\left(r_1 r_2 + k_2 m_1 + k_1 m_2 + k_2 m_2\right)\right] \\ d_4 &= k_1\Big[-m_1 m_2\left(r_2 k_1 + r_1 k_2\right)^2 - k_1 k_2\left(r_2 m_1 + r_1 m_2 + r_2 m_2\right)^2 + \\ &\quad \left(r_2 k_1 + r_1 k_2\right)\left(r_2 m_1 + r_1 m_2 + r_2 m_2\right)\left(r_1 r_2 + k_2 m_1 + k_1 m_2 + k_2 m_2\right)\Big] \end{aligned}$$

The above-mentioned standard deviations ($\sigma_{\ddot{z}_2}$, $\sigma_{z_2-z_1}$, σ_{F_z}) depend on the cube of the square root of the power of vehicle speed v, on the square root of track irregularity coefficient A_b, and on analytical functions of the model's parameters. These analytical expressions are rather complex. As the square roots do not depend on v, according to this formulation, the optimal settings of suspension design variables *do not* depend on vehicle speed.

10.1.4 Formulae for Vanishing Primary Damping Using the 1S-PSD (Eq. (10.7))

Primary dampers are commonly not fitted into the bogies of urban and suburban railway vehicles. This is due to the fact that, the primary stiffness (k_1) being relatively high (due to the high ratio payload/tare mass), the pitching of the bogie is already limited by the primary stiffness itself and dampers are not necessarily needed to limit the dynamic pitching oscillations. However, primary dampers can be adopted for reducing track wear.

Setting the primary damping to zero, i.e. $r_1 = 0$, the standard deviations (10.25), (10.26), (10.27), (10.28) assume relatively simple expressions

– Force on the axle-box

$$\sigma_{F_z} = 2\pi^2 v^{3/2}\sqrt{A_b}\cdot \sqrt{\frac{k_1^2 m_1^2 m_2^2 + k_1(m_1+m_2)^2\left[r_2^2(m_1+m_2) - 2k_2 m_1 m_2\right] + k_2(m_1+m_2)^2\left[k_2(m_1+m_2)^2 + k_1 m_2^2\right]}{r_2 k_1 m_2^2}} \tag{10.29}$$

– Body acceleration

$$\sigma_{\ddot{z}_2} = 2\pi^2 v^{3/2}\sqrt{A_b}\sqrt{\frac{r_2^2 k_1 (m_1+m_2) + k_2^2 (m_1+m_2)^2 + k_1 k_2 m_2^2}{r_2 k_1 m_2^2}} \tag{10.30}$$

– Secondary stroke

$$\sigma_{z_2 - z_1} = 2\pi^2 v^{3/2}\sqrt{A_b}\sqrt{\frac{k_2 (m_1+m_2)^2 + k_1 m_2^2}{r_2 k_1 k_2}} \tag{10.31}$$

The primary stroke is obviously $\sigma_{z_1-\xi} = \sigma_{F_z}/k_1$.

10.1.5 Simplified Formulae Using the 1S-PSD (Eq. (10.7))

Equations (10.29)–(10.31) can be simplified by neglecting those terms which vanish when parameter values refer to actual railway vehicles[1]:

$$\sigma_{F_z} = B\sqrt{\frac{k_1 m_1^2}{r_2}(1 + \beta\mu^2 + \beta^2\mu^2) + m_1 r_2 (3+\mu)} \tag{10.32}$$

$$\sigma_{\ddot{z}_2} = B\sqrt{\frac{r_2}{m_2} + \frac{k_2}{r_2}} \tag{10.33}$$

$$\sigma_{z_2 - z_1} = B\sqrt{\frac{m_2^2}{r_2 k_2}} \tag{10.34}$$

A comparison has been made between the standard deviations $\sigma_{\ddot{z}_2}, \sigma_{z_2-z_1}, \sigma_{F_z}$ computed by Eqs. (10.29)–(10.31) and the corresponding standard deviations $\sigma_{\ddot{z}_2}$, $\sigma_{z_2-z_1}$, σ_{F_z} computed by Eqs. (10.32)–(10.34) (vehicle parameters in Table 10.1.). The absolute value of the error was 0.49% for the force on the axle-box (Eq. (10.32)), 1.7% for the body vertical acceleration (Eq. (10.33)), 0.37% for the secondary stroke (Eq. (10.34)).

[1]For actual railway vehicles usually one finds

$$5 < \mu < 10, \quad 0.01 < \beta < 0.2$$

with $\mu = m_2/m_1$ and $\beta = k_2/k_1$. Assuming that $(1+\mu)/\mu \approx 1$ and $\sqrt{\beta+1} \approx 1$ Eqs. (10.29)–(10.31) and Eqs. (10.39)–(10.41) can be simplified

Table 10.1. Data of the reference railway vehicle taken into consideration (data from [76])

m_{1r}	773 kg
m_{2r}	5,217 kg
k_{1r}	28,240,000 N/m
k_{2r}	162,000 N/m
r_{1r}	21,890 Ns/m
r_{2r}	14,600 Ns/m

10.1.6 Complete Formulae Using the 2S-PSD (Eq. (10.8))

The analytical expressions of the standard deviations of the force on the axle-box (F_z), of the vertical acceleration of the body ($\ddot{z}_2$), of the (vertical) relative displacement body–bogie have been derived by solving analytically Eq. (10.23). They are not reported here because of their extreme complexity. In this case it is more convenient to solve Eq. (10.23) numerically. It has to be noticed that, due to the employed spectrum, contrary to what happens in Eqs. (10.25)–(10.28), the speed v is mixed among the parameters A_v, ω_c, m_i, r_i, k_i. According to this formulation the optimal suspension design variables do depend on vehicle speed v.

$$\sigma_{F_z} = l_{1DS}\left(v, A_v, \omega_c, m_1, m_2, r_1, k_1, r_2, k_2\right) \tag{10.35}$$

$$\sigma_{\ddot{z}_2} = l_{2DS}\left(v, A_v, \omega_c, m_1, m_2, r_1, k_1, r_2, k_2\right) \tag{10.36}$$

$$\sigma_{z_2 - z_1} = l_{3DS}\left(v, A_v, \omega_c, m_1, m_2, r_1, k_1, r_2, k_2\right) \tag{10.37}$$

$$\sigma_{z_1 - \xi} = l_{4DS}\left(v, A_v, \omega_c, m_1, m_2, r_1, k_1, r_2, k_2\right) \tag{10.38}$$

10.1.7 Formulae for Vanishing Primary Damping Using the 2S-PSD (Eq. (10.8))

If the primary damping vanishes (i.e. $r_1 = 0$) the standard deviations in Eqs. (10.35)–(10.38) assume relatively simple expressions

– Force on the axle-box

$$\sigma_{F_z} = \omega_c \sqrt{\pi\ v\ A_v} \sqrt{\frac{n_{v1}}{d_{v1}}} \tag{10.39}$$

$$\begin{aligned} n_{v1} = {} & k_1^2 m_1^2 m_2^2 \left(k_2 + r_2\omega_c + m_2\omega_c^2\right) + k_1 \left(m_1 + m_2\right) \left(-2k_2 m_1 m_2 + \right. \\ & \left. m_1 r_2^2 + m_2 r_2^2\right) \left(k_2 m_1 + k_2 m_2 + m_1 r_2 \omega_c + m_2 r_2 \omega_c + m_1 m_2 \omega_c^2\right) + \\ & + k_2^2 \left(m_1 + m_2\right)^2 \left(k_2 m_1^2 + 2k_2 m_1 m_2 + k_1 m_2^2 + k_2 m_2^2 + \right. \\ & \left. m_1^2 r_2 \omega_c + 2m_1 m_2 r_2 \omega_c + m_2^2 r_2 \omega_c + m_1^2 m_2 \omega_c^2 + m_1 m_2^2 \omega_c^2\right) \\ d_{v1} = {} & 2m_2^2 r_2 \left(k_1 k_2 + k_1 r_2 \omega_c + k_2 m_1 \omega_c^2 + k_1 m_2 \omega_c^2 + k_2 m_2 \omega_c^2 + \right. \\ & \left. m_1 r_2 \omega_c^3 + m_2 r_2 \omega_c^3 + m_1 m_2 \omega_c^4\right) \end{aligned}$$

– Body acceleration

$$\sigma_{\ddot{z}_2} = \omega_c \sqrt{\pi \; v \; A_v} \sqrt{\frac{n_{v2}}{d_{v2}}} \tag{10.40}$$

$$\begin{aligned} n_{v2} = & \left[k_1 r_2^2 \left(k_2 m_1 + k_2 m_2 + m_1 r_2 \omega_c + r_2 m_2 \omega_c + \right.\right. \\ & \left. m_1 m_2 \omega_c^2\right) + k_2^2 \left(k_2 m_1^2 + 2 k_2 m_1 m_2 + k_1 m_2^2 + k_2 m_2^2 + \right. \\ & \left.\left. m_1^2 r_2 \omega_c + 2 m_1 m_2 r_2 \omega_c + m_2^2 r_2 \omega_c + m_1^2 m_2 \omega_c^2 + m_2^2 m_1 \omega_c^2\right)\right] \\ d_{v2} = & \left[2 m_2^2 r_2 \left(k_1 k_2 + k_1 r_2 \omega_c + k_2 m_1 \omega_c^2 + k_1 m_2 \omega_c^2 + k_2 m_2 \omega_c^2 + \right.\right. \\ & \left.\left. m_1 r_2 \omega_c^3 + m_2 r_2 \omega_c^3 + m_1 m_2 \omega_c^4\right)\right] \end{aligned}$$

– Secondary stroke

$$\sigma_{z_2 - z_1} = \omega_c \sqrt{\pi \; v \; A_v} \sqrt{\frac{n_{v3}}{d_{v3}}} \tag{10.41}$$

$$\begin{aligned} n_{v3} = & \; k_2 m_1^2 + 2 k_2 m_1 m_2 + k_1 m_2^2 + k_2 m_2^2 + m_1^2 r_2 \omega_c + \\ & 2 m_1 m_2 r_2 \omega_c + m_2^2 r_2 \omega_c + m_1^2 m_2 \omega_c^2 + m_1 m_2^2 \omega_c^2 \\ d_{v3} = & \; 2 r_2 \left(k_1 k_2 + k_1 r_2 \omega_c + k_2 m_1 \omega_c^2 + k_1 m_2 \omega_c^2 + k_2 m_2 \omega_c^2 + \right. \\ & \left. m_1 r_2 \omega_c^3 + m_2 r_2 \omega_c^3 + m_1 m_2 \omega_c^4\right) \end{aligned}$$

The primary stroke is obviously $\sigma_{z_1 - \xi} = \sigma_{F_z} / k_1$.

10.1.8 Simplified Formulae Using the 2S-PSD (Eq. (10.8))

Equations (10.39)–(10.41) can be simplified by neglecting those terms which vanish when parameter values refer to actual railway vehicles (see footnote 1).

– Force on the axle-box

$$\sigma_{F_z} = \omega_c \sqrt{\pi \; v \; A_v} \sqrt{\frac{n_{a1}}{d_{a1}}} \tag{10.42}$$

$$\begin{aligned} n_{a1} = & \; k_1^2 m_1^2 \left(\beta \; k_1 + r_2 \omega_c + m_1 \mu \; \omega_c^2 + \frac{1}{k_1 m_1} \left(-2 \beta \; k_1 m_1 + r_2^2\right) \cdot \right. \\ & \left. \left(\beta \; k_1 \mu + r_2 \omega_c \mu + m_1 \mu \; \omega_c^2\right)\right) \\ d_{a1} = & \; 2 r_2 \left(\beta \; k_1^2 + k_1 r_2 \omega_c + m_1 k_1 \omega_c^2 \mu + r_2 m_1 \omega_c^3 \mu + m_1^2 \mu \; \omega_c^4\right) \end{aligned}$$

– Body acceleration

$$\sigma_{\ddot{z}_2} = \omega_c \sqrt{\pi \; v \; A_v} \cdot \sqrt{\frac{\frac{k_1 r_2}{m_1 \mu} \left(\beta \; k_1 + r_2 \omega_c + m_1 \omega_c^2\right) + \beta^2 \frac{k_1^2}{r_2} \left(k_1 + r_2 \omega_c + m_1 \omega_c^2\right)}{2 \left(\beta \; k_1^2 + k_1 r_2 \omega_c + m_1 k_1 \omega_c^2 \mu + r_2 m_1 \omega_c^3 \mu + m_1^2 \mu \; \omega_c^4\right)}} \tag{10.43}$$

– Secondary stroke

$$\sigma_{z_2-z_1} = \omega_c \sqrt{\pi\ v\ A_v}\cdot \sqrt{\frac{m_1^2\mu^2\left(k_1 + r_2\omega_c + m_1\omega_c^2\right)}{2r_2\left(\beta\ k_1^2 + k_1 r_2\omega_c + m_1 k_1\omega_c^2\mu + r_2 m_1\omega_c^3\mu + m_1^2\mu\ \omega_c^4\right)}} \tag{10.44}$$

The primary stroke is obviously $\sigma_{z_1-\xi} = \sigma_{F_z}/k_1$.

A comparison has been made between the $\sigma_{\ddot{z}_2}$, $\sigma_{z_2-z_1}$, σ_{F_z} computed by Eqs. (10.39)–(10.41) and the $\sigma_{\ddot{z}_2}$, $\sigma_{z_2-z_1}$, σ_{F_z} computed by Eqs. (10.42)–(10.44) (vehicle parameters in Table 10.1.). In the vehicle speed range 20–100 m/s, the error varies from −1.5 to 0.3% for the force on the axle-box (Eq. (10.42)), from −2.2 to 1.2% for the body vertical acceleration (Eq. (10.43)), from −0.2 to −1.4 for the secondary stroke (Eq. (10.44)).

10.2 Validation

In order to validate the simple model described in Sect. 10.1, a comparison with the data presented in [76] is performed. In [76] a model for the study of the vertical dynamics of railway vehicles has been proposed. The model is rather complex as it accounts for the heave, pitch and roll of body and bogies. Vehicle body bending and torsional modes of vibration have been also included. From [76] it is argued that the output model responses were validated experimentally with satisfactory results. The data of the vehicle studied in [76] are reported in Table 10.1.

In [76] the adopted PSD of the stochastic process defining track irregularity in the vertical plane (2S-PSD, see Eq. (10.8)) is

$$P_{SD\xi}(\omega) = \frac{A_v\omega_c^2 v}{\omega^2\left(\omega^2 + \omega_c^2\right)} = \frac{A_v\Omega_c^2 v^3}{\omega^2\left(\omega^2 + v^2\Omega_c^2\right)} \tag{10.45}$$

where A_v=0.035 cm^2 rad/m is the track quality coefficient and $\Omega_C = 0.99$ rad/m is the break wave number. $P_{SD\xi}\ (\omega)$ is plotted in Fig. 10.2.

In [76] an in-depth sensitivity analysis was performed referring to the standard deviations of body acceleration and of secondary stroke. Unfortunately, no data are available on primary stroke and force on the axle-box.

The following validation has been performed by considering Eqs. (10.36) and (10.37) which refer to the model shown in Fig. 10.1 and described in Sect. 10.1. The adopted PSD of the track irregularity is given by Eq. (10.45). A steady speed v equal to 177 km/h has been considered.

10.2.1 Primary Stiffness

In [76] the primary suspension stiffness k_1 was at first increased by a factor 4, then decreased by the same factor relatively to the reference value

Table 10.2. Comparison between computed results and data referring to an actual vehicle (adapted from [76])

k_1	$\sigma_{\ddot{z}_2}(\mathrm{m/s^2})$, from [76]	$\sigma_{\ddot{z}_2}(\mathrm{m/s^2})$, Eq. (10.36)	$\sigma_{z_2-z_1}(\mathrm{mm})$, from [76]	$\sigma_{z_2-z_1}(\mathrm{mm})$, Eq. (10.37)
$k_{1r}/4$	0.56	0.62	9.6	9.8
k_{1r}	0.72	0.68	9.5	9.6
$4k_{1r}$	0.67	0.69	9.5	9.5

Variation of the primary stiffness k_1 (k_{1r} is reported in Table 10.1)

(Table 10.1). In Table 10.2 the values of standard deviations $\sigma_{\ddot{z}_2}$ and $\sigma_{z_2-z_1}$ computed by means of Eqs. (10.36) and (10.37) are compared with the corresponding values reported in [76]. Body acceleration is substantially unaffected by primary stiffness variation: The error given by the model with respect to the corresponding data in [76] is always less than 10%. For the secondary stroke the error is always less than 2%.

10.2.2 Natural Frequency

The natural frequencies of the body rigid modes (heave, pitch and roll) were varied in [76] without changing their damping ratios. This was achieved by suitable variations in the secondary suspension stiffness, k_2 and damping r_2. The effect of halving and doubling the reference natural frequencies was investigated. The results are reported in Table 10.3. The effect is relevant both on body acceleration and on secondary suspension stroke. Again the simple model is able to give the responses of the reference vehicle with a limited error.

10.2.3 Damping Ratio

The effect of the secondary damping on body acceleration and secondary stroke was studied by varying the coefficient r_2, such that body's heave damping ratio $\varsigma = r_2/\left(2\sqrt{k_2 m_2}\right)$ was increased from the reference value (0.25)

Table 10.3. Effect of the vehicle body natural frequency

Natural frequency	f_h (Hz)	k_2 (N/m)	r_2 (Ns/m)	$\sigma_{\ddot{z}_2}(\mathrm{m/s^2})$, from [76]	$\sigma_{\ddot{z}_2}(\mathrm{m/s^2})$, Eq. (10.36)	$\sigma_{z_2-z_1}(\mathrm{mm})$, from [76]	$\sigma_{z_2-z_1}(\mathrm{mm})$, Eq. (10.37)
Halved	0.445	162,000/4	14,600/2	0.39	0.41	15.1	13.8
Base	0.885	162,000	14,600	0.72	0.68	9.5	9.6
Double	1.769	162,000·4	14,600·2	1.17	1.27	5.3	6.6

Comparison between computed results and data referring to an actual vehicle (adapted from [76])

Table 10.4. Comparison between computed results and data referring to an actual vehicle (adapted from [76])

Damping ratio	ς	r_2 (Ns/m)	$\sigma_{\ddot{z}_2}$(m/s^2), from [76]	$\sigma_{\ddot{z}_2}$(m/s^2), Eq. (10.33)	$\sigma_{z_2-z_1}$(mm), from [76]	$\sigma_{z_2-z_1}$(mm), Eq. (10.34)
r_2	0.250	14,600	0.72	0.68	9.5	9.6
$r_2 \cdot 1.5$	0.375	$14{,}600 \cdot 1.5$	0.85	0.81	7.5	7.7
$r_2 \cdot 2$	0.50	$14{,}600 \cdot 2$	1.02	0.96	6.4	6.6
$r_2 \cdot 2\sqrt{2}$	0.707	$14{,}600 \cdot 2\sqrt{2}$	1.20	1.19	5.4	5.5

Variation of the secondary suspension damping r_2 (r_{2r} is reported in Table 10.1)

to 0.375, 0.50 and 0.707 (Table 10.4) (ς is defined as if k_1 were infinity). Referring to the vertical body acceleration, the error given by the model is always less than 6%. For the secondary stroke the error is always less than 3%.

10.3 Parameter Sensitivity Analysis

The dynamic response of the railway vehicle system model in Fig. 10.1 is analysed on the basis of Eqs. (10.29)–(10.31) and Eqs. (10.39)–(10.41). The same analysis (not reported here for sake of space) has been performed by means of Eqs. (10.32)–(10.34) and Eqs. (10.42)–(10.44). The discrepancies were negligible.

A typical railway passenger vehicle for intercity service is taken into consideration (Table 10.1). The results of the parameter sensitivity analysis are shown in Figs. 10.3–10.5. The parameters are varied within wide ranges. The data are presented in non-dimensional form, i.e. the standard deviation of interest σ is divided by the corresponding one (σ_r) computed by considering the parameters at their reference values (see Table 10.1)

$$\sigma_{F_z r} = \sigma_{F_z}(m_{1r}, m_{2r}, r_{1r}, r_{2r}, k_{1r}, k_{2r}) \tag{10.46}$$

$$\sigma_{\ddot{z}_2 r} = \sigma_{\ddot{z}_2}(m_{1r}, m_{2r}, r_{1r}, r_{2r}, k_{1r}, k_{2r}) \tag{10.47}$$

$$\sigma_{z_2-z_1 r} = \sigma_{z_2-z_1}(m_{1r}, m_{2r}, r_{1r}, r_{2r}, k_{1r}, k_{2r}) \tag{10.48}$$

The non-dimensional standard deviations derived from Eqs. (10.29)–(10.31) do not depend on vehicle speed. The opposite occurs for the non-dimensional standard deviations derived from Eqs. (10.39)–(10.41) (referring to 2S-PSD, Eq. (10.8)). For this reason the last non-dimensional standard deviations are analysed at two different vehicle speeds, low speed (10 m/s) and high speed (100 m/s).

10.3.1 Standard Deviation of Force on Axle-box

Figure 10.3 shows that

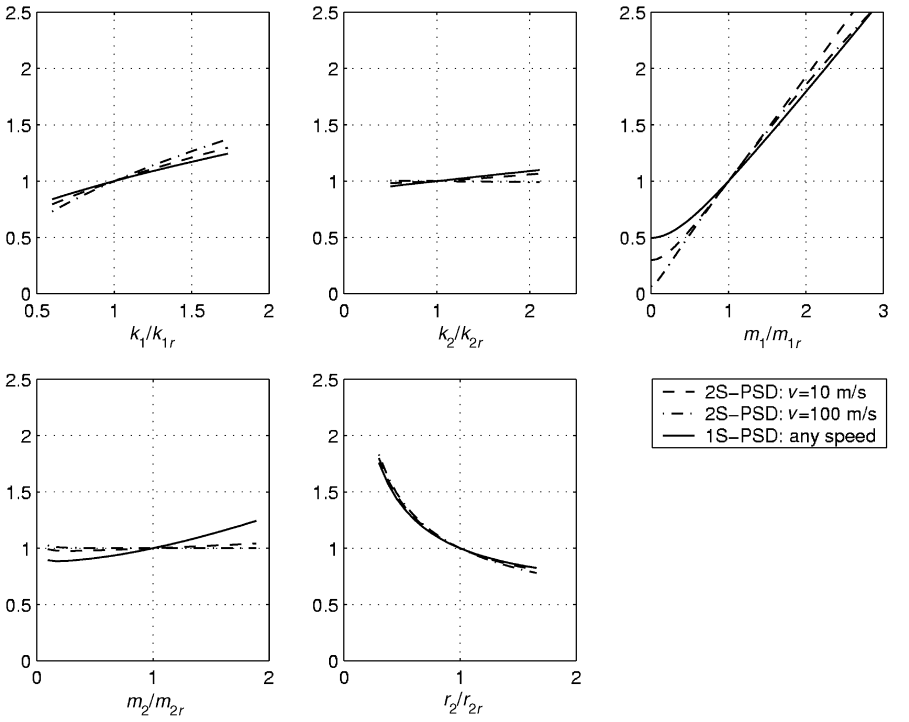


Fig. 10.3. $\sigma_{F_z}/\sigma_{F_z r}$, non-dimensional standard deviation of force on the axle-box as function of model parameters. Data of the reference vehicle in Table 10.1. Each diagram has been obtained by varying one single parameter, the other ones being constant and equal to those of the reference vehicle

- σ_{F_z} depends almost linearly on the primary suspension stiffness k_1,
- σ_{F_z} does not depend significantly on the secondary suspension stiffness k_2,
- σ_{F_z} depends almost linearly on m_1 (non-linear at start),
- σ_{F_z} does not depend significantly on m_2 if the excitation is given by Eq. (10.8) (2S-PSD),
- the secondary suspension damping r_2 has an important influence on the standard deviation σ_{F_z}.

Some of the above considerations can be derived by a simple inspection of Eqs. (10.32) and (10.42).

10.3.2 Standard Deviation of Body Acceleration

By inspection of Fig. 10.4 one may notice that

- k_1 does not influence significantly $\sigma_{\ddot{z}_2}$,
- m_1 does not influence significantly $\sigma_{\ddot{z}_2}$ if the excitation is given by Eq. (10.7) (1S-PSD),

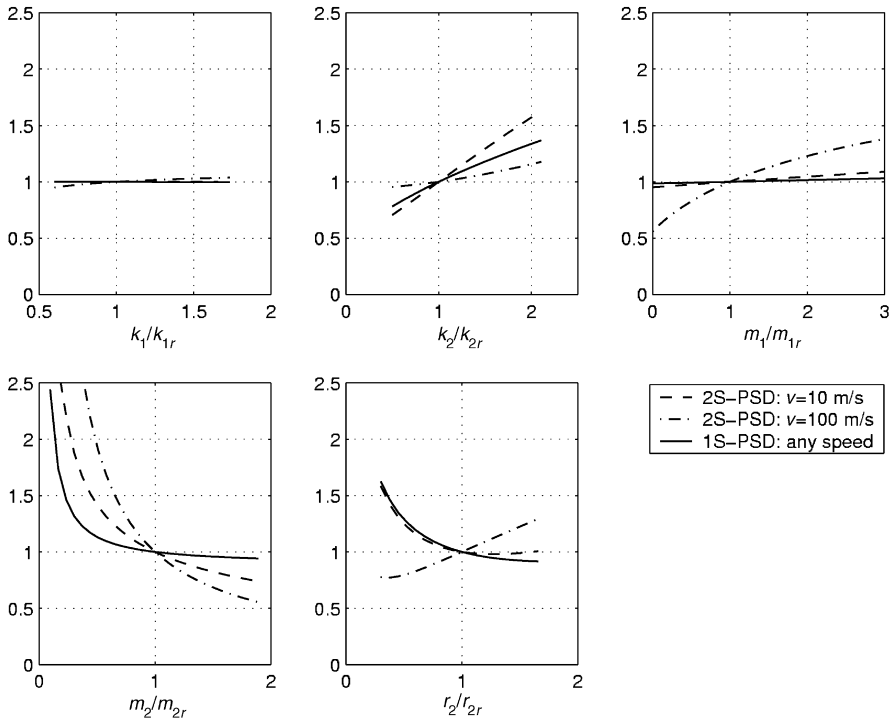


Fig. 10.4. $\sigma_{\ddot{z}_2}/\sigma_{\ddot{z}_{2r}}$, non-dimensional standard deviation of body acceleration as function of model parameters. Data of the reference vehicle in Table 10.1. Each diagram has been obtained by varying one single parameter, the other ones being constant and equal to those of the reference vehicle

– $\sigma_{\ddot{z}_2}$ may decrease strongly as m_2 increases,
– the parameters of secondary suspension (k_2, r_2) have a remarkable influence on the $\sigma_{\ddot{z}_2}$. For a 2S-PSD excitation the effect of an increase of the secondary suspension damping r_2 is positive at low speed but negative at high speed. For a 1S-PSD excitation the effect of an increase of the secondary suspension damping r_2 is positive at any speed.

Some of the above considerations can be derived by a simple inspection of Eqs. (10.33) and (10.43).

10.3.3 Standard Deviation of Secondary Stroke

By inspection of Fig. 10.5, one may notice that

– $\sigma_{z_2-z_1}$ is not influenced by primary suspension stiffness k_1,
– the stiffness of the secondary suspension k_2 has a remarkable influence on the $\sigma_{z_2-z_1}$. For a 2S-PSD excitation the effect of a variation of k_2 is less relevant increasing the vehicle speed,

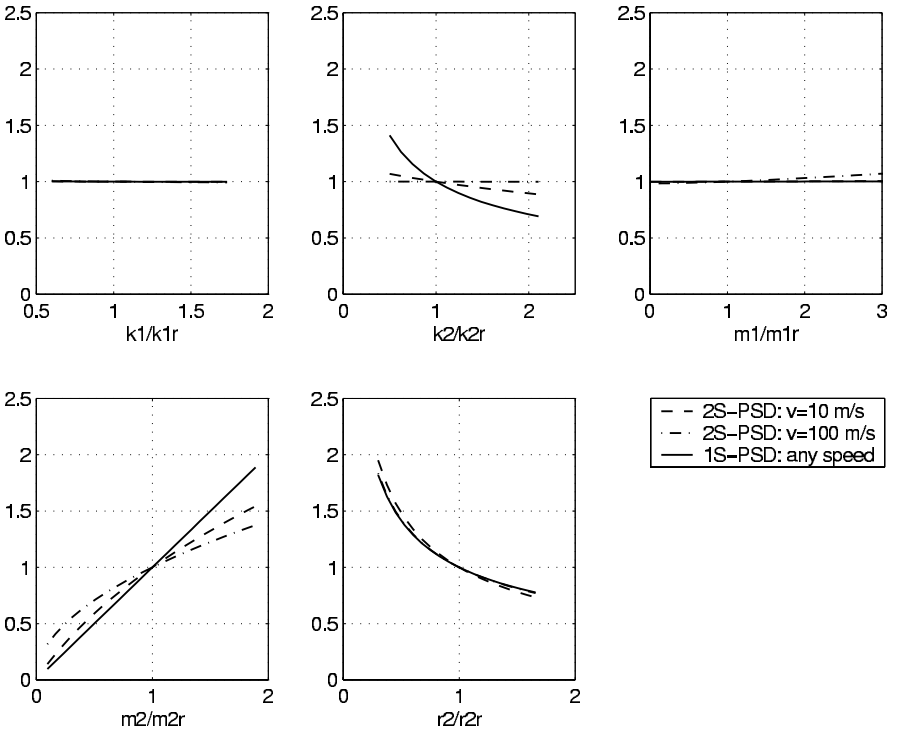


Fig. 10.5. $\sigma_{z_2-z_1}/\sigma_{z_2-z_1 r}$, non–dimensional standard deviation of secondary stroke as function of model parameters. Data of the reference vehicle in Table 10.1. Each diagram has been obtained by varying one single parameter, the other ones being constant and equal to those of the reference vehicle.

- $\sigma_{z_2-z_1}$ is not influenced significantly by m_1,
- $\sigma_{z_2-z_1}$ depends almost linearly on m_2 considering a 1S-PSD excitation. The relationship is non-linear considering the 2S-PSD excitation,
- $\sigma_{z_2-z_1}$ is influenced remarkably by secondary suspension damping r_2.

Some of the above considerations can be derived by a simple inspection of Eqs. (10.34) and (10.44).

10.3.4 Optimal Secondary Suspension Design Variables

The constraints method introduced in Sect. 3.7.1 has been used to optimise the design variables of the secondary suspension of a railway vehicle described by the simple system model in Fig. 10.1. The design variables to be optimised were the stiffness k_2 and the damping r_2 of the secondary suspension, the objective functions were $\sigma_{\ddot{z}_2}$and $\sigma_{z_2-z_1}$.

Derivation of Optimal $\sigma_{\ddot{z}_2}$, $\sigma_{z_2-z_1}$ and Optimal k_2, r_2 Using 1S-PSD

The objective functions are defined by Eqs. (10.33) and (10.34). The optimisation procedure based on constraints method and described in Sect. 3.7.1 is applied as follows:

a) from Eq. (10.34) the expression of r_2 as function of $\sigma_{z_2-z_1}$ and k_2 is derived

$$r_2 = \frac{B^2 m_2^2}{k_2 \cdot \sigma^2_{z_2-z_1}} \tag{10.49}$$

b) the expression of $\sigma_{\ddot{z}_2}$ as function of r_2 and $\sigma_{z_2-z_1}$ by substituting the expression of r_2 (Eq. (10.49)) in Eq. (10.33)

$$\sigma_{\ddot{z}_2} = B\sqrt{\frac{B^2 m_2}{k_2 \sigma^2_{z_2-z_1}} + \frac{k_2^2 \sigma^2_{z_2-z_1}}{B^2 m_2^2}} = B\sqrt{(\cdot)} \tag{10.50}$$

c) the following derivative equal to zero gives the stationary solution

$$\frac{d\sigma_{\ddot{z}_2}}{dk_2} = B\frac{d\sqrt{(\cdot)}}{dk_2} = B\frac{1}{2\sqrt{(\cdot)}}\frac{d(\cdot)}{dk_2} = \quad 0 \tag{10.51}$$

the term $\frac{1}{2\sqrt{(\cdot)}}$ is always greater than zero, so solving with respect to k_2

$$-\frac{B^2 m_2}{\sigma^2_{z_2-z_1} k_2^2} + 2\frac{k_2}{B^2 m_2^2}\sigma^2_{z_2-z_1} = 0 \tag{10.52}$$

thus

$$k_2 = \sqrt[3]{\frac{B^4 m_2^3}{2\,\sigma^4_{z_2-z_1}}} \tag{10.53}$$

d) finally, by substituting Eq. (10.53) in Eq. (10.49) and Eq. (10.33), the expression of $\sigma_{\ddot{z}_2}$ as function of $\sigma_{z_2-z_1}$ is obtained

$$\sigma_{\ddot{z}_2} = \sqrt[6]{\frac{27}{4}\frac{B^8}{\sigma^2_{z_2-z_1}}} \tag{10.54}$$

this equation defines the relationship between the standard deviation of the acceleration of the body and the standard deviation of the secondary suspension stroke when both the standard deviations are minimised.

e) The equation which defines the optimal design variable set is

$$r_2 = r_{2ott} = \sqrt{2k_2 m_2} \tag{10.55}$$

For the system composed by the mass m_2 , the damper r_2, and the spring k_2 (mass m_1 fixed), the critical damping may be defined as

$$r_{2crit} = \sqrt{4k_2 m_2} \tag{10.56}$$

it follows

$$r_{2ott} = \frac{1}{\sqrt{2}} \cdot r_{2crit} \tag{10.57}$$

By setting the stiffness and the damping of the secondary suspension as indicated above, the best compromise between the standard deviation of the body acceleration and the standard deviation of the secondary stroke is obtained.

Derivation of Optimal $\sigma_{\ddot{z}_2}$, $\sigma_{z_2-z_1}$ and Optimal k_2, r_2 Using 2S-PS

On the basis of Eqs. (10.40) and (10.41), a numerical search has been undertaken to find both the optimal set $\sigma_{\ddot{z}_2}$,$\sigma_{z_2-z_1}$ and the optimal set k_2, r_2. The corresponding plots are reported in Figs. 10.6 and 10.7, respectively.

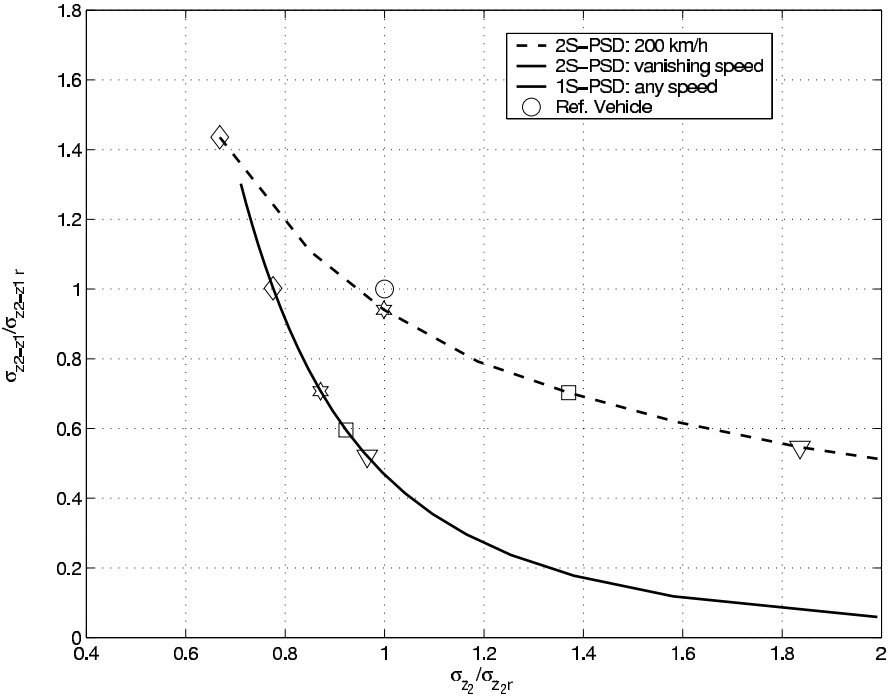


Fig. 10.6. Optimal $\sigma_{\ddot{z}_2}$ and optimal $\sigma_{z_2-z_1}$ plotted in non-dimensional form. The curves are obtained by varying k_2 and r_2, the points highlighted by using special symbols (triangle, square, ...) refer to the points in Fig. 10.7. Vehicle parameters in Table 10.1.

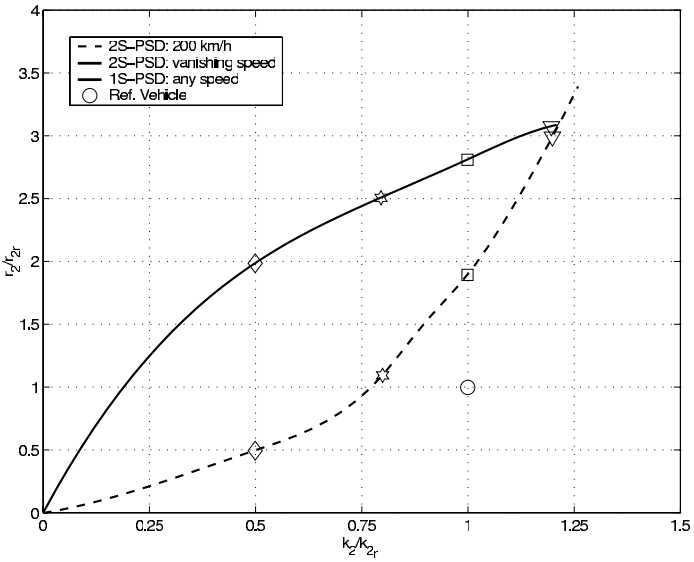


Fig. 10.7. Optimal k_2 and optimal r_2 plotted in non-dimensional form for minimising $\sigma_{\ddot{z}_2}$ and $\sigma_{z_2-z_1}$, the points highlighted by using special symbols (triangle, square, ...) refer to the points in Fig. 10.6. Vehicle parameters in Table 10.1.

Optimal $\sigma_{\ddot{z}_2}$, $\sigma_{z_2-z_1}$ and Optimal k_2, r_2

Both for the 1S-PSD (Eqs. (10.54), (10.55)) and for the 2S-PSD excitations, optimal $\sigma_{\ddot{z}_2}$, $\sigma_{z_2-z_1}$ and optimal k_2, r_2 are plotted in non-dimensional form in Figs. 10.6 and 10.7, respectively. To obtain non-dimensional values, reference is made to a railway passenger vehicle whose relevant parameters are reported in Table 10.1. $\sigma_{\ddot{z}_2}$ increases when $\sigma_{z_2-z_1}$ decreases, i.e. these two standard deviations are conflicting. The designer should choose, on the basis of given technical specifications, the desired compromise between $\sigma_{\ddot{z}_2}$ and $\sigma_{z_2-z_1}$ by selecting one point lying on the curves plotted in Fig. 10.6, e.g. one of those marked with special symbols (triangle, square, ...). Having chosen the preferred compromise between $\sigma_{\ddot{z}_2}$ and $\sigma_{z_2-z_1}$, the corresponding values of the design variables k_2, r_2 are uniquely defined. This correspondence between the points of the curves plotted in the $\sigma_{\ddot{z}_2}$, $\sigma_{z_2-z_1}$ plane (Fig. 10.6) and the points of the curves in the k_2, r_2 plane (Fig. 10.7) are highlighted by special symbols in Figs. 10.6 and 10.7. By inspection of Fig. 10.6 one may notice that for the 1S-PSD the non-dimensional $\sigma_{\ddot{z}_2}$ and $\sigma_{z_2-z_1}$ do not depend on vehicle speed. This is consequence of the fact that for this excitation spectrum, the speed parameter v is not mixed with those of the vehicle system k_2, r_2, ... (see Sect. 10.1). On the contrary, for the 2S-PSD, which is very frequently found in actual applications, the non-dimensional $\sigma_{\ddot{z}_2}$ and $\sigma_{z_2-z_1}$ do depend on vehicle speed (see Sect. 10.1). This suggests that vehicle suspension design variables should vary with vehicle speed in order to keep the optimality conditions. This is technically easily achievable and hopefully in the future adaptive suspensions could be adopted for railway vehicles.

In Fig. 10.7 the relationship between optimal stiffness k_2 and optimal damping r_2 is highlighted. To keep the best compromise between $\sigma_{\ddot{z}_2}$ and $\sigma_{z_2-z_1}$, the damping r_2 has to increase with the stiffness k_2, both for the 1S-PSD and for the 2S-PSD excitation. The rate of change of r_2 with respect to k_2 does not depend on vehicle speed for the 1S-PSD and varies considerably with the vehicle speed for the 2S-PSD excitation.

10.4 Conclusion

Analytical formulae have been derived in order to estimate the response of railway vehicles to random excitations generated by the vertical track irregularity. The accuracy of the derived formulae has been assessed by comparison with data presented in the literature. The analytical formulae should estimate with reasonable accuracy the dynamic behaviour of an actual railway vehicle running on rigid track. Referring to the performed validation, the sensitivity of body acceleration and secondary stroke to vehicle suspension design variables (primary and secondary stiffness, secondary damping) is captured satisfactorily by the analytical formulae. It has been found that analytical formulae (in complete form) predicted the standard deviations of both body acceleration and secondary stroke with an error always less than 10%, and often less than 2%. On the basis of the validated analytical formulae, a theoretical parameter sensitivity analysis has been performed with reference to the standard deviations of force on the axle-box, of body acceleration, of secondary suspension stroke. All these standard deviations are influenced by secondary suspension design variables. Particularly, the secondary damping affects significantly the body acceleration. A general result (confirmed by common experience) is the strong influence of the type of track irregularity on all the considered standard deviations. The bogie mass and the primary stiffness do not seem to influence considerably the secondary stroke.

By using the derived analytical formulae in the second part of the chapter, the constraints method has been applied to find the best trade-off between conflicting objective functions such as $\sigma_{\ddot{z}_2}$ (standard deviation of the body acceleration) and $\sigma_{z_2-z_1}$(the standard deviation of the secondary stroke). The design variables of the secondary suspension (stiffness k_2 and damping r_2) of a railway vehicle have been optimised with the aim of minimising both $\sigma_{\ddot{z}_2}$ and $\sigma_{z_2-z_1}$. Simple analytical formulae have been derived for the optimal $\sigma_{\ddot{z}_2}$,$\sigma_{z_2-z_1}$ and correspondingly optimal k_2, r_2. Optimal $\sigma_{\ddot{z}_2}$ increases when both optimal k_2 and optimal r_2 increase, the opposite occurs for optimal $\sigma_{z_2-z_1}$. If the excitation is defined by the 1S-PSD the optimal secondary suspension settings do not depend on vehicle speed. The opposite occurs for the more realistic 2S-PSD excitation, thus it seems reasonable to recommend, for future research, comprehensive studies on the application of adaptive stiffness and damping elements to railway vehicle secondary suspension systems.

11

Optimal Design of the Layout of Railway Passenger Vehicles

A basic study on definition of the optimal layout of railway passenger vehicles, derived from [150], is presented in this chapter. An attempt to answer the following questions will be performed on the basis of multi-objective programming.

- Which could be the architecture of a railway passenger vehicle that would guarantee the lowest possible life cycle cost (LCC)?
- How long should a railway passenger vehicle be?
- And how many wheelsets (or axles) should it have?
- Which could be the optimal inner layout?

These basic questions seem not easy to be answered, especially if one requires responses derived from rigourous (i.e. mathematical) reasoning. The present chapter tries to answer the addressed questions by presenting and applying a method to design optimal railway passenger vehicles.

In the last years, a number of non-conventional railway vehicles have been designed and built. Apart from Talgo [24], which dates back many decades and may be well reputed as an outstanding revolutionary realisation, the Paris metro train called BOA [215], the recent sub-urban trains of München and Copenhagen [48, 71, 235], represent new important attempts to improve the performances (LCC) of rail vehicles by adopting new layouts.

According to the knowledge of the authors, a limited number of papers have been presented in the literature on the concept design of optimal railway vehicles [69, 147, 199]. In [69], a description of future railway vehicles is made. Maybe, for the first time in the specialised technical literature the problem of reducing the mass of railway vehicles by means of reducing their length is addressed. Additionally, in [69], the importance of newly designed wheelsets/bogies is clearly stated for making possible the adoption of non-conventional vehicle layout. In [199], the design of railway passenger vehicles is reviewed by means of in-depth analyses and considerations on mass distribution among the different sub-systems which the vehicle is composed

of. In both these two fundamental papers (as well as in [51, 52]), the engineering effort for orienting the design of railway vehicles was not based on optimisation theory. In [147] and later in [174], the first attempt to optimise the layout of railway passenger vehicles was made by means of multi-objective programming (MOP). With respect to [147], a more structured and comprehensive work is presented on the topic. Genetic algorithms have been chosen and employed here as the appropriate mathematical tool to compute the optimal solutions (Pareto-optimal set) [28, 67, 190, 236, 251].

The objective functions for obtaining an optimal railway passenger vehicle are introduced. These refer to minimum LCC, i.e. maximum payload, minimum tare weight, maximum ride comfort, minimum track damaging, etc. An analysis of the proposed optimal solutions referring to urban-, sub-urban-, and IC-trains is presented.

11.1 Design Aims and Related Objective Functions

One of the main objectives in the design of a railway vehicle is minimising the *life cycle cost* (LCC) of the whole vehicle or, alternatively, minimising the LCC of the different sub-systems the vehicle is composed of. LCC is the estimated cost relative to the entire life of the vehicle, and can be expressed as

$$\begin{aligned} C_{LCC} &= C_C + C_M + C_E - V_R \\ &= C_C + \sum_{j=1}^{N} \frac{C_{M_j}}{(1+i)^j} + \sum_{j=1}^{N} \frac{C_{E_j}}{(1+i)^j} - \frac{V_R}{(1+i)^N} \end{aligned}$$

where

- C_C is the vehicle purchase cost
- C_M the maintenance costs (entire life)
- C_E the operating costs (entire life)
- V_R the residual value of the vehicle at the end of the life cycle
- N the vehicle life (years)
- i the interest rate
- C_{M_j} the maintenance cost at the jth year
- C_E the operating costs at the jth year.

The computation of LCC is not straightforward; C_M, C_E, V_R are very difficult to be estimated because they depend on i which varies considerably in the time span of vehicle life. Additionally, the number of years N a vehicle is run is not known exactly and this makes the estimation of i almost impossible at the requested accuracy.

Together with LCC, other important aspects (such as safety, comfort, visual attractiveness, environmental impact, etc.) have to be taken into account

to design an optimal vehicle. These aspects are often conflicting and this obviously involves the finding of the best compromise among them. The best compromise can be ultimately determined by the designer by means of a proper application of MOP.

The reduction of vehicle mass is strongly correlated with the reduction of LCC [147] provided that the structural properties (safety against fatigue failure and adequate response to crash) are maintained (LCC is reduced due reduction of vehicle purchase cost, reduction of energy consumption during service, expected reduction of track damaging, etc.). The known methods to reduce mass [69, 145, 147] are reducing vehicle length, eliminate or re-design bogies (i.e. reducing the number of wheelsets per car) [69, 169] and/or adopt light materials [145]. Reducing the number of wheelsets causes the increase of axle loads which might have a negative effect on track/wheel damage. Another fundamental objective is the return maximisation that is related to the payload maximisation. Obviously, it is important to increase the payload without reducing sensibly the room available for the passengers. More comfortable and wider vehicles are reputed to increase income, but train length should be minimised to save space along platforms.

From above considerations, it is clear that designing the layout of a railway vehicle is just finding a satisfactory compromise among conflicting requirements.

In the following sections a number of applications will be presented, with the *only* aim to show how the layout of a railway vehicle can be optimised. More detailed technical features and complicated railway vehicle models could have been introduced, but this would have been too much devoted to a specific application.

Table 11.1 shows the complete list of the objective functions to be minimised that have been introduced.

The first two indices aim to maximise payload and minimise the tare weight and vehicle length. A ride comfort index (vertical direction), similar to well-known Wz [60] has been introduced. An index directly proportional to track damage has been introduced to account for the vehicle/track dynamic interaction. Two different indices have been introduced to consider the room

Table 11.1. Objective functions

Indices to be optimised (minimised)	Symbol
(Unloaded vehicle mass)/(fully loaded vehicle mass)	m_r
(Vehicle length)/(payload)	l/p
Ride comfort (discomfort)	$\sigma_{\ddot{h}}$
Track damage	ϵ_{Ni}
A_{pass} index (see) [only for IC train]	A_{pass}
(Standing passengers)/(total passengers) [only for Urban & Sub-urban trains]	St_p/n_{pass}

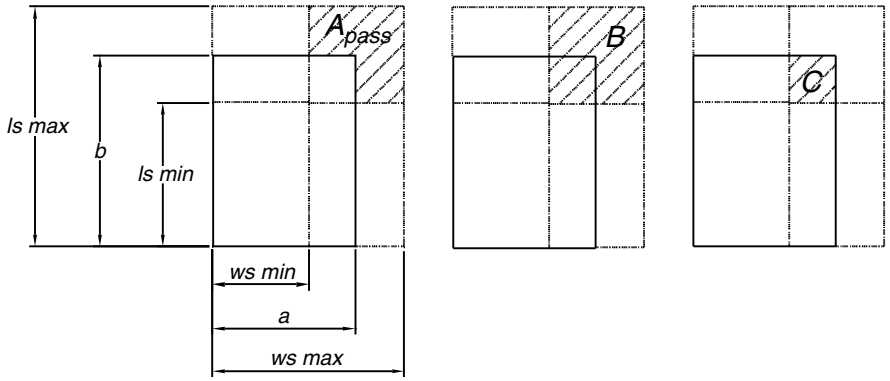


Fig. 11.1. Definition of the A_{pass} index

available for passengers. For the IC trains the index is the space reserved to a single seated passenger, for the urban and sub-urban trains the index is the ratio between standing passengers and total passengers. No objective functions referring to hunting stability and lateral vibrations/comfort [76,101,259] have been introduced. This is due to the fact that a detailed *optimal* design of the wheelset/bogie would have been involved. Additionally, the aim of the study is to indicate the best layout of railway vehicles, leaving to specialists the problem of finding the adequate tuning of primary and secondary suspensions. In not a very far future passive or actively controlled wheelsets (single wheelset bogies) will be hopefully available in order to reduce considerably the hunting/lateral vibration and wheel/wear problems [70, 205, 259].

11.2 Design Variables to Be Tuned

The system design variables that have been varied are reported in Table 11.2, together with the respective ranges of variation. Design variables have been divided into two groups, *independent* ones and *dependent* ones. Dependent design variables do depend on the values of independent ones. For example, the body cross-width, given the gauge, depends on vehicle length. The wheelset mass depends on the static axle load. The number of independent design variables is 11 (see Table 11.2). In addition to design variables, the whole set of vehicle parameters that have been considered are reported in Table 11.2.

11.3 Constraints

Three constraints have been considered during the optimisation procedure:

– *Gauge.* The international gauge (continental Europe) has been considered, the maximum cross-width being $w_s = 3,150$ mm.

Table 11.2. Design variables and parameters used in the numerical examples

Symbol	Units	Design variable/parameter	Type	IC	Sub-urban	Urban
l	m	Car length	i	10–25	10–25	10–25
n_b		Number of bogies	i	0–3	0–3	0–3
n_w		Number of wheelsets	i	1–6	1–6	1–6
l_{minc}	m	Minimum aisle/side corridor width	i	0.42	0.80–1.50	1.20–2.40
l_s	m	Seat length	i	0.80–1.10	0.72–0.88	0.64–0.80
w_s	m	Seat cross-width	i	0.50–0.65	0.50–0.55	0.45–0.50
l_{ve}	m	Entrance vestibule width	i	0.95–1.10	1.30	1.30
A_{se}	m^2	Area reserved for toilets	p	2	2	–
l_f	m	Wall thickness	d	0.04–0.07	0.04–0.07	0.04–0.07
p_{mq}		Max standing passenger per m^2	p	4	6	7
m_{pa}	kg	Mass of a passenger	p	80	75	70
m_{pv}	kg/m^2	Floor mass per m^2	d	30	30	30
m_{ba}	kg/pass	Luggage carrying structure mass	d	3.5	3.5	–
m_{se}	kg	Seat mass	p	10	10	10
m_{wi}	kg/m	Windows mass per m of vehicle len.	d	28	28	28
m_{do1}	kg	Internal door mass	p	40	40	40
m_{do2}	kg	Communicating door mass	p	50	50	50
m_{do3}	kg	Entrance door mass	p	80	80	80
n_{do1}		Number of Internal doors	p	2	–	–
n_{do2}		Number of Communicating doors	p	2	2	2
m_{ri}	kg	Toilet mass	p	480	480	–
n_{ri}		Number of toilets	p	1	1	–
m_{ic}	kg/pass	Air conditioning plant mass	p	25	25	25
m_{ie}	kg/m	Electrical installation mass	d	100	100	100
m_{ip}	kg/m	pneumatic installation mass	d	27	27	27
m_{or}	kg	Couplings mass	p	650	650	650
m_{rv}	%	Interiors mass	d	3%	3%	3%
$m_f w$	kg	Wheelset mass	d	750–1,800	750–1,800	750–1,800
k_1	N/m	Primary suspension stiffness	i	0.5×10^6–9	0.5×10^6–9×10^6	0.5×10^6–$9\ 10^6$
k_2	N/m	Secondary suspension stiffness	i	0.1×10^6–2×10^6	0.1×10^6–2×10^6	0.1×10^6–2×10^6
r_1	N/m	Primary suspension damping	i	1×10^3–25×10^3	1×10^3–25×10^3	1×10^3–25×10^3
r_2	N/m	Secondary suspension damping	i	15×10^3–80×10^3	15×10^3–80×10^3	15×10^3–80×10^3
m_b	kg	Bogie mass	d	2×10^3–10×10^3	2×10^3–10×10^3	2×10^3–10×10^3
k_0	N/m	Track, ballast stiffness	p	$40 \times 10^6 - 1 \times 10^8$	40×10^6–1×10^8	40×10^6–1×10^8
r_0	Ns/m	Track, ballast damping	p	50×10^3	50×10^3	50×10^3
m_{tr}	kg	Track mass (per wheel)	p	650	650	650
A	m	Track irregularity (see Eq. (11.13))	p	0.928×10^{-9}	0.928×10^{-9}	0.928×10^{-9}
W_s	m	Gauge cross-width	p	3.15	3.15	3.10
R_{min}	m	Minimum reference curve radius	p	250	250	150
D		Number of passengers per year	p	10^6	80×10^6	250×10^6
v	m/s	Reference vehicle speed	p	50	40	30

Design variables can be either *independent* (i) or *dependent* (d). Parameter values (p) are kept fixed during the optimisation process.

– *Maximum load per axle.* In the application examples the maximum load per axle has been limited to 17 t/axle, load corresponding to current European high speed trains (ETR 500, TGV, ICE3, AVE).
– σ_{h-y}. Given a reference speed and a reference track vertical irregularity (see Table 11.2), the relative displacement between car body and wheelset has to be limited ($\sigma_{h-y} < 10$ mm).

11.4 Objective Functions

The mathematical relationships describing the objective functions (reported in Table 11.1) as function of both the design variables and parameters (reported in Table 11.2) are now introduced.

11.4.1 Unloaded Vehicle Mass/Fully Loaded Vehicle Mass

The m_r index is defined as the ratio between the vehicle tare mass (m_t) and the total mass (m_{tot}) for the fully loaded vehicle. The total mass (m_{tot}) is the sum of the tare mass m_t and the payload p:

$$m_r = \frac{m_t}{m_t + p} \tag{11.1}$$

m_r has to be minimised and can be estimated at a sufficient level of accuracy if the main parameters defining the vehicle are known. Given the vehicle external dimensions and the layout (number of wheelsets n_w, type and number of bogies n_b, vehicle length, number of seats and room for them, number and width of toilets, number and width of entrance vestibules), both the tare weight (or tare mass) and the maximum number of passengers (seated and standing) can be estimated. The tare weight can be computed by using the empirical relations presented in [199], which give estimates of the masses of vehicle sub-systems. The payload is easily derived from the number of passengers.

To compute m_r the following steps have to be made in advance. More refined derivations could be made with respect to those presented here, e.g. accounting for rolling motions of the vehicle (see [57,199]). This has not been made here for the sake of simplicity.

– *External dimensions.* The lateral displacement of the centre line of the vehicle during turning (Fig. 11.2) is given by

$$f_1 = \frac{i^2}{8R} \tag{11.2}$$

where i is the bogie centre distance (or maximum wheelsets distance if no bogies are present), R is the minimum turning radius. The lateral displacements of the ends of the vehicle (Fig. 11.2) are (very simple geometrical model)

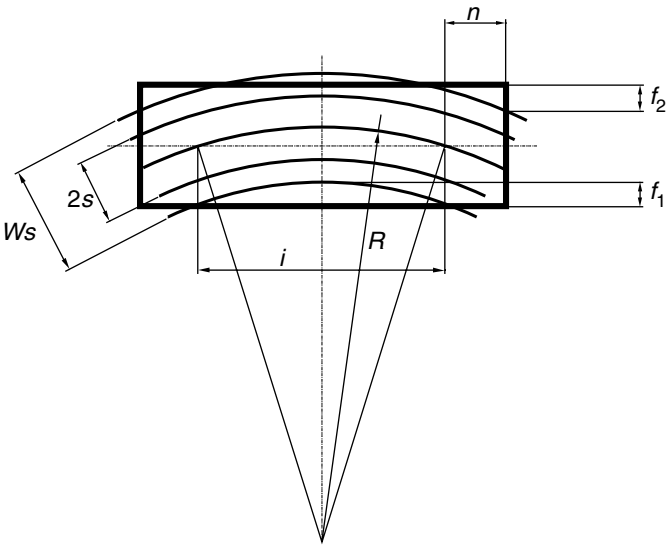


Fig. 11.2. Railway vehicle during turning (simplified scheme)

$$f_2 = \frac{n}{2\,R}(i+n) \tag{11.3}$$

where n is the distance between the centre of the bogie and the end of the car (end of the car – closer wheelset distance if no bogies are present). By imposing that $f_1 = f_2$, the value of n which does not require smoothing of the body at its ends can be computed:

$$n = l\left(\frac{1}{2} - \frac{\sqrt{2}}{4}\right) \tag{11.4}$$

More wide bodies could be conceived if smoothing is allowed and this means that the result that will be discussed later could be further improved by a proper and more refined body design. The vehicle cross-width w_v is

$$w_v = W_s - 2\,f_1 - f_s \tag{11.5}$$

where W_s is the gauge cross-width, f_s is the safety factor.

– *Auxiliary structures, entrance vestibules and aisle (or side corridor).* The interior surface is computed by considering the wall thickness l_f, the length of the entrance vestibule l_{ve}. For IC trains the aisle cross-width is computed as the difference between the inner-body cross-width and the total lateral length of the seats

$$l_c = w_v - 2 \times l_f - n_{spr} \times w_s \tag{11.6}$$

where n_{spr} is the number of seats per row. For urban and sub-urban trains the gangway can be used for standing passengers.

– *Number of seats.* The number of seats is computed by considering the inner-body room diminished by the room required for the entrance vestibule(s), for the toilet(s), for the aisle or side corridor.

Detailed Mass Computation

In this section the computation of the masses which constitute the railway vehicle is discussed in detail.

– *Body frame.* The body frame mass is a non-linear function of the vehicle length. A second-order polynomial [170] gives a good estimate of the masses of existing railway vehicles

$$m_{st}\ (\text{kg}) = a_1 + a_2\, l + a_3\, l^2 \tag{11.7}$$

where a_1 $(6,223\ \text{kg})$, a_2 $(-615.6\ \text{kg/m})$, a_3 $(25.87\ \text{kg/m}^2)$ are the coefficients obtained by interpolation of data referring to rail vehicles running in continental Europe.

– *Bogie and axles.* The bogie mass (two wheelsets) m_b (kg) depends linearly on the axle load a_l $(10^4\ \text{N})$ acting on it

$$m_b = 667(a_l - 6) + 2000. \tag{11.8}$$

For a single wheelset bogie the mass has been set to $1,200$ kg. The wheelset mass m_w (kg) depends linearly on the vertical axle load a_l $(10^4\ \text{N})$ acting on it (outside journal axle)

$$m_w = 62.5(a_l - 6) + 750 \tag{11.9}$$

– *Passengers mass.* For IC trains the mass of each passenger has been set to 80 kg (baggage included). For urban and sub-urban trains the mass has been set to 70 and 75 kg respectively.

– *Windows, seats, interior paneling, auxiliary equipments.* According to [199], the window mass can be considered as function of the vehicle length only. The mass of the structure carrying the luggage m_{ba} is function of the number of passengers. The seat mass for a railway vehicle varies from 12.5 to 20 kg, depending on the seat complexity. The lower bound for the seat mass is 7.7 kg (aircraft seat). We have considered a lower bound of 10 kg that is the mean value between the minimum railway vehicle seat mass and aircraft seat mass. The insulating material is about the 3% of the total mass of the vehicle. According to [199] a value of 30 kg/m^2 has been selected for the floor. The lavatory mass (only on IC and sub-urban vehicles) has been set to 480 kg.

– *Doors.* The mass of the doors varies according to their function, external doors 80 kg, intercommunication doors 50 kg, internal doors 40 kg. The proposed solution has two intercommunication doors, two external doors for each vestibule, and only for IC trains two internal doors for each vestibule.

- *Electrical, compressed-air installation, air conditioning plant.* The mass of the electrical, compressed-air installation and the air conditioning plant varies slightly with the rail vehicle type. A mean value for a high-speed train is about $4,000$ kg, which corresponds to 160 kg/m^2. Considering that the analysed railway vehicles do not include power cars, the value is decreased to 100 kg/m^2. The total mass for the compressed-air installation for passenger cars with automatic brake, with compressed-air control for secondary services is about 800 kg, that is 30 kg/m (mass for unit of length). We assume a mean value of 27 kg/m. The air conditioning plant is designed as a function of the number of passengers. A mean value is 27 kg/passenger.
- *Draft gear.* Due to safety requirements the coupling system has a substantial mass: buffers, 600 kg coupling hook, 240 kg. Considering automatic buffing and draw couplers the total mass can be reduced to 650 kg.
- *Structural mass, payload, maximum load per axle.* The structural mass is obtained by summation of all the masses considered above. The payload p is computed by considering the number of seats for IC vehicles and the total number of passengers (seated and standing) for urban and sub-urban vehicles. The load per axle is the ratio between the total mass (structural mass + effective load) and the number of axles n_a.

11.4.2 Vehicle Length/Payload

In the optimisation procedure, the numerical value of ratio vehicle length/ payload is derived from the value of the vehicle length (design variable) and the value of the payload (computed as indicated above).

11.4.3 Ride Comfort

The passenger vibrational comfort [146] is considered. The discomfort coefficient is related to the standard deviation of the vertical acceleration acting on the passenger. This index is function of the suspension characteristics and of the masses (wheelset/bogies, body) of the vehicle. A simplified vehicle model has been used to compute the discomfort index $\sigma_{\ddot{z}}$. The model is discussed in detail in Sect. 11.4.5.

11.4.4 Vehicle/Track Dynamic Interaction

Maintenance costs of the track are an important part of the *LCC* of a vehicle. For this reason, an objective function related to the track settlement under dynamic loading has been introduced. This index has been defined in [73] as the differential settlement of the track after a given loading cycle has been applied

$$\epsilon_{Ni} = [(c_1 + c_2 \frac{k_0}{c_3}) \frac{P_{dyn}}{P_{ref}}]^w \log N \tag{11.10}$$

where ϵ_{Ni} is the vertical settlement (mm) of the ith sleeper after N load cycles, k_0 is the track stiffness which is a function of P_{ref} (k_0 increases almost linearly with P_{ref}), P_{ref} is the static reference vertical load, P_{dyn} is the vertical dynamic load, N is the number of load cycles.

The values of c_1, c_2, c_3 and w are dependent on the type of track substructure and can vary from the following given values: $c_1 = 194, c_2 = -1.96, c_3 = 1 \cdot 34 \times 10^6, w = 0.3$

P_{dyn} is computed by using the simplified vehicle model described in Sect. (11.4.5):

$$P_{dyn} = P_{ref} + 3\sigma_{Fz} \tag{11.11}$$

where σ_{Fz} is the standard deviation of the vertical load between road and track. For the computation of the number of cycles N, the following relation can be used:

$$N = \frac{D}{n_{pass}} n_w \tag{11.12}$$

where D is the annual number of passengers, n_w is the number of wheelsets per car and n_{pass} is the number of passengers in a car.

11.4.5 Three Degrees of Freedom Model

The discomfort coefficient and the track damage are computed by resorting to the 3 d.o.f. system model represented in Fig. 11.3. The standard deviation of passenger acceleration ($\sigma_{\ddot{x}}$), the standard deviation of the vertical force between wheels and track (σ_{F_z}) and the standard deviation of the relative displacement wheel-chassis (σ_{h-y}) are computed.

The power spectral density of the track irregularity is given as [198]

$$P_{SD}(\Omega) = \frac{A(2\pi v)^3}{\Omega^4} \quad 0.1 < \Omega < 628 \,\text{rad/s} \tag{11.13}$$

where A is the track irregularity coefficient, v (m/s) is the vehicle speed, Ω (rad/s) is the circular frequency.

The equation of motion of the linear system (Fig. 11.3) is

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{R}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{q}Z_0 e^{i\Omega t} \tag{11.14}$$

where

$$\mathbf{q} = [h \quad u \quad y]^T \quad \mathbf{F} = [0 \quad 0 \quad -\Omega^2 m_{tr} + i\Omega r_0 + k_0]^T$$

$$\mathbf{M} = \begin{pmatrix} m_c & 0 & 0 \\ 0 & m_b & 0 \\ 0 & 0 & m_w + m_{tr} \end{pmatrix}$$

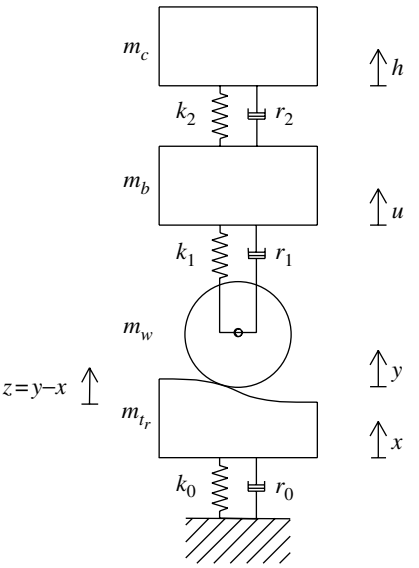


Fig. 11.3. 3 d.o.f. vehicle model. Symbols in Table 11.3

Table 11.3. Symbols referring to the simplified vehicle model in Fig. 11.2

Car body mass relative to the single wheel	m_c
Bogie mass relative to half wheelset	m_b
Half wheelset Mass	m_w
Mass of track, ballast vibrating with the wheelset	m_{tr}
Secondary suspension stiffness	k_2
Secondary suspension damping	r_2
Primary suspension stiffness	k_1
Primary suspension damping	r_1
Ballast stiffness	k_0
Ballast damping	r_0
Imposed relative displacement	$z = y - x$
Vertical displacement of the car body	h
Vertical displacement of the bogie	u
Vertical displacement of the wheelset	y
Vertical displacement of the track	x

$$\mathbf{R} = \begin{pmatrix} r_2 & -r_2 & 0 \\ -r_2 & r_1 + r_2 & -r_1 \\ 0 & -r_1 & r_1 + r_2 \end{pmatrix}$$

$$\mathbf{K} = \begin{pmatrix} k_2 & -k_2 & 0 \\ -k_2 & k_1 + k_2 & -k_1 \\ 0 & -k_1 & k_1 + k_2 \end{pmatrix}$$

We can write

$$[H_1 \quad H_2 \quad H_3]^T = (-\Omega^2\mathbf{M} + \mathbf{R}i\Omega + \mathbf{K})^{-1}\mathbf{F} \tag{11.15}$$

from the previous expression

$$\begin{gathered} H_{h-y} = (H_1 - H_3) \\ H_{\ddot{h}} = -\Omega^2 H_1 \\ H_{F_z} = -\Omega^2(m_c H_1 + m_b H_2 + m_w H_3) \end{gathered} \tag{11.16}$$

The variance of the generic output $(i = (h-y), (\ddot{h}), (F_z))$ is

$$\sigma^2_{x_i} = \int_{-\infty}^{+\infty} |H_i|^2 P_{SDu}(\Omega) d\Omega \tag{11.17}$$

11.4.6 Other Indices

Two additional objective functions have been introduced in order to take into account the passenger comfort. The room available for each seated passenger need not be too limited (IC trains) and the number of seated passengers need not be too small with respect to the number of standing passengers (urban and sub-urban trains).

- *Urban and sub-urban trains:* S_{tp}/n_{pass} is the ratio between standing passengers and the total number of passengers.
- *IC trains:* the area A_{pass} indicated in Fig. 11.1 defined as

  $$A_{pass} = (ws_{max} - ws_{min})(ls_{max} - ls_{min}) - (a - ws_{min})(b - ls_{min}) = B - C$$

  where a is the width and b is the length of the room available for a seated passenger ($ws_{min} \leq a \leq ws_{max}$ and $ls_{min} \leq b \leq ls_{max}$)

The indices S_{tp}/n_{pass} and A_{pass} are inversely correlated to l/p and m_r. The designer will have to select the desired trade-off with respect to the type of train to be designed.

11.5 Analysis and Choice of Preferred Optimal Solutions

The numerical computation of the non-dominated solutions (i.e. optimal or Pareto-optimal solutions) has been performed by resorting to genetic algorithms (GAs) (see Sect. 3.4.4).

Three Pareto-optimal sets have been obtained referring respectively to IC, sub-urban and urban railway vehicles. In each of these sets the designer may find vehicles with the minimum tare weight, with the maximum payload, with the highest comfort. The characteristics belonging to these vehicles represent the *best* compromises among the objective functions listed in Table 11.1, with the constraints introduced in Sect. 11.3. By analysing all of the optimal

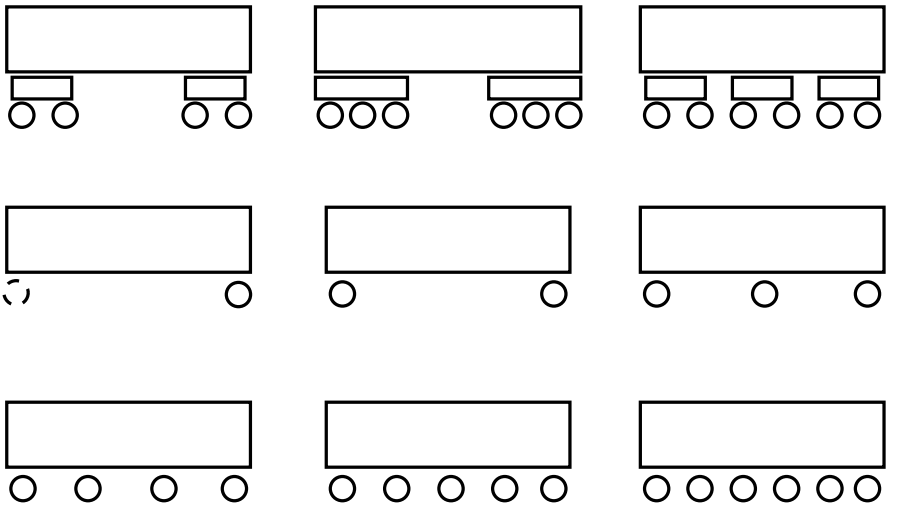


Fig. 11.4. Passenger car layouts that have been considered

solutions belonging to the Pareto-optimal set, the following considerations arise.

It is not possible to improve considerably all of the objective functions reported in Table 11.1, if a traditional vehicle layout is maintained (i.e. vehicles with two bogies and four wheelsets (Fig. 11.4, double-decker coaches have not been considered, for sake of space).

The objective function m_r *unloaded vehicle mass/fully loaded vehicle mass* is optimum if the bogies are eliminated, i.e. single-wheelset bogies are adopted. Actually, bogies increase the tare weight of the vehicle without augmenting the payload. For optimal IC vehicles the seat cross-width is directly correlated to m_r. The reasons for this is obvious: more space left to seats involves slightly higher structural mass and less passengers (i.e. lower payload). The smoothening of the carbody (at the fore and aft ends) may be crucial to fill the vehicle with seats.

The objective function *vehicle length/payload* is obviously optimum when the length is reduced. But, as it may be expected, the length should not be excessively low. The recommended length (for all kinds of vehicles that have been considered) could be about 15 m (the lower bound that was considered was 10 m, see Table 11.2). This is due to the vestibules that, e.g. for IC vehicles, decrease the number of seated passengers per unit car length.

With respect to the track damaging ϵ_{Ni}, the best cars have a relatively high payload and a reduced number of wheelsets. This conclusion is due to the adopted model of track damage which takes into account both the static/dynamic wheel loads and the *number of loading cycles* that are applied to the track. If the transportation demand is fixed (i.e. the number of passenger per year D is given), with a relatively high axle load and with a

reduced number of wheelsets, it is possible to decrease the number of load cycles N and thus to reduce track damage.

The passenger comfort is related to a proper design of vehicle suspension system and tuning of stiffness and damping. The constraint on the standard deviation of the relative displacement between car body and wheelset defines the lower bounds on stiffness and damping.

Three optimal solutions belonging to the Pareto-optimal set are described in detail in the following subsections.

11.5.1 Inter-city Cars

The layout of one of the optimal (Pareto-optimal) passenger cars for IC service is shown in Fig. 11.5. The numerical data are reported in Table 11.4. The optimised vehicle is shorter than the reference one. It has three wheelsets – no bogies, a very low axle load (one half with respect to the reference car) and a wide carbody cross-section which allows to place five seats per row. This leads to exceptionally high loading capacity (m_r improves from 0.9 to 0.7) and a favourable vehicle/track interaction. The distance between the first and the last wheelset is almost equal to the wheelbase of two axle freight wagons. It is assumed by the authors that the hunting instability at high speeds could be avoided by a proper tuning of longitudinal suspension stiffness.

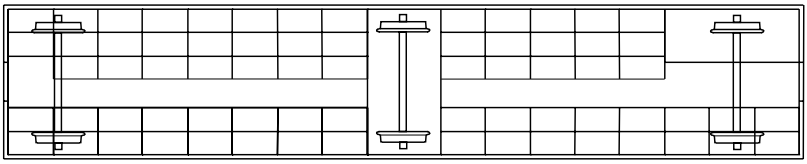


Fig. 11.5. Layout of one optimised car for inter-city (IC) service

11.5.2 Sub-urban Cars

The layout of one of the optimal (Pareto-optimal) sub-urban passenger cars is shown in Fig. 11.6. The numerical data are reported in Table 11.5. The proposed car has two single wheelsets without bogies. Nevertheless, the number of passengers per unit train length is higher than that of the reference car, the track damage is reduced because the number of loading cycles is reduced; additionally, the dynamic wheel load is reduced due to the reduced unsprung masses (there are no bogies). The proposed car is significantly shorter than the actual ones. The wheelbase is almost the same as of freight wagons with two axles. A smooth running at the maximum speed (e.g. 140–160 km/h) could be achieved with minor design effort of the running gear.

Table 11.4. Description of one optimal car for IC service and comparison with a reference passenger car running in continental Europe (Eurofima UIC-Z, II class)

	Optimal car	Reference car
Car length l (m)	14.67	25.48
Number of bogies n_b	–	2
Number of wheelsets n_w	3	4
Aisle/side corridor width l_{minc} (m)	0.42	0.77
Seat length l_s (m)	0.823	0.940
Seat cross-width w_s (m)	0.500	0.630
Wall thickness l_f (m)	0.040	0.070
Primary suspension stiffness k_1 (N/m)	1,000,000	3,240,000
Secondary suspension stiffness k_2 (N/m)	–	950,000
Primary suspension damping r_1 (Ns/m)	42,000	15,000
Secondary suspension damping r_2 (Ns/m)	–	60,000
Car cross-width W_v (m)	3.000	2.825
Car tare mass m_t (kg)	13,000	42,800
Payload p (kg)	5,600	5,280
Number of seated passengers Si_p	70	66
Number of standing pass. (overload) S_{tp}	45	102
Maximum axle load Q (daN)	7,400	14,070
σ_{F_z} (N)	1,410	4,300
σ_{h-y} (m)	0.0028	0.0043
$\sigma_{\ddot{h}}$ (m/s)^2	1.34	1.39
m_r	0.70	0.89
Vehicle length/payload l/p (m/kg)	0.00262	0.00482
A_{pass} (m^2)	0.045	0.027
Track damage ϵ_{Ni} (mm)	37.0	47.7
Number of cars per 100 m	6.8	3.9
Number of seated passengers per 100 m	477	259
Car tare mass/number of seated passengers m_t/Si_p (kg)	186	648

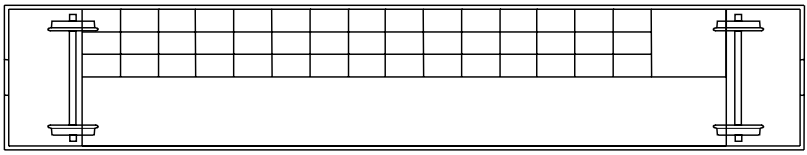


Fig. 11.6. Layout of one optimised car for sub-urban service

11.5.3 Urban Cars

Figure 11.7 shows an optimised (Pareto-optimal) urban train. Data are reported in Table 11.6. Similar considerations to those presented for IC and sub-urban cars hold in this case.

Table 11.5. Description of one optimal car for sub-urban service and comparison with an actual reference car running in continental Europe (MD 1980 of FS, Italian State Railways)

	Optimal car	Reference car
Car length l (m)	15.51	25.48
Number of bogies n_b	–	2
Number of wheelsets n_w	2	4
Aisle cross-width l_{minc} (m)	1.32	0.48
Seat length l_s (m)	0.72	0.88
Seat cross-width w_s (m)	0.50	0.55
Wall thickness l_f (m)	0.040	0.050
Primary suspension stiffness k_1 (10^6N/m)	1.1	4.5
Secondary suspension stiffness k_2 (10^6N/m)	1.7	1.2
Primary suspension damping r_1 (Ns/m)	–	15,000
Secondary suspension damping k_2 (Ns/m)	–	60,000
Car cross-width W_v (m)	2.900	2,825
Car tare mass m_t (kg)	12,540	36,500
Payload p (kg)	15,000	13,650
Number of seated passengers Si_p	45	82
Number of standing pass. (overload) S_{tp}	155	100
Maximum load per axle Q (daN)	13,770	12,500
σ_{F_z} (N)	2,800	3,290
σ_{h-y} (m)	0.0028	0.0029
$\sigma_{\ddot{h}}$ (m/s^2)	0.86	1.03
m_r	0.45	0.73
Vehicle length/payload l/p (m/kg)	0.00097	0.00190
S_{tp}/n_{pass}	0.77	0.55
Track damage ϵ_{Ni} (mm)	40.0	43.8
Number of cars per 100 m	6.5	3.9
Number of seated passengers per 100 m	290	321
Number of passengers per 100 m	1,290	714
Car tare mass/number of seated passengers m_t/Si_p (kg)	279	445
Car tare mass/number of passengers $m_t/(Si_p + S_{tp})$ (kg)	62.7	200

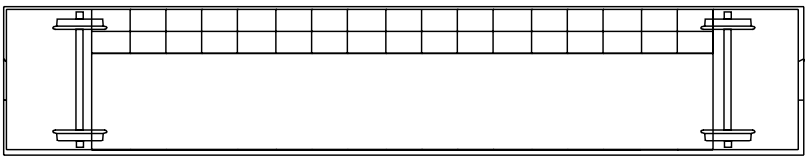


Fig. 11.7. Layout of one optimised car for urban service

Table 11.6. Description of one optimised urban car and comparison with a reference underground passenger car [52]

	Optimal car	Reference car
Car length l (m)	14.58	16.88
Number of bogies n_b	–	2
Number of wheelsets n_w	2	4
Seat cross-width w_s (m)	0.45	
Min. aisle cross-width l_{minc} (m)	1.56	
Primary suspension stiffness k_1 (N/m)	1,353,000	2,000,000
Secondary suspension stiffness k_2 (N/m)	–	2,000,000
Primary suspension damping r_1 (Ns/m)	60,000	15,000
Secondary suspension damping r_2 (Ns/m)	–	20,000
Car cross-width W_v (m)	2.72	2.85
Car tare mass m_t (kg)	10,780	16,900
Payload p (kg)	17,010	16,100
Number of seated passengers Si_p	34	40
Number of standing passengers S_{tp}	209	190
Maximum axle load Q (daN)	13,440	8,100
σ_{F_z} (N)	1,400	1,057
σ_{h-y} (m)	0.0019	0.0019
$\sigma_{\ddot{h}}$ (m/s^2)	0.55	1.24
m_r	0.387	0.512
Vehicle length/payload l/p (m/kg)	0.00086	0.0010
S_{tp}/n_{pass}	0.86	0.83
Track damage ϵ_{Ni} (mm)	31.4	34.2
Number of cars per 100 m	6.9	5.9
Number of passengers per 100 m	1,667	1,362
Car tare mass/number of passengers $m_t/(Si_p + S_{tp})$ (kg)	44	73

11.6 Conclusion

A basic study for the concept design of railway vehicles has been presented. The study is based on the application of multi-objective programming. Five relevant objective functions (Table 11.1) related to the life cycle cost of vehicles have been mathematically defined as functions of 11 independent vehicle design variables (Table 11.2). Genetic algorithms have been used as an appropriate tool to solve the numerical problem of finding the best compromise between conflicting requirements on objective functions.

The method has been applied to optimise the layout of inter-city, urban and sub-urban railway vehicles, with reference to the gauge of continental Europe. It turned out that very efficient vehicles could be designed if non-conventional layouts were adopted. In particular, it has been found that short cars (about 15 m) without bogies can carry a high payload, are characterised

by relatively low axle load and tare weight, should not damage remarkably the track and may provide acceptable comfort to passengers.

Designing short cars makes it possible to better exploit the gauge, making more room available for passengers. Moreover, the mass of the body structures can be reduced, because both the bending moment on the body and the deflection of the body are reduced. Shorter cars could reduce the platforms operating costs.

The elimination of bogies can increase the ratio between total mass and payload and reduce the maximum load per axle. In fact, the bogie mass increases the vehicle tare mass, with no improvement in the payload but increasing the load per axle. The solution without bogies does not seem to alter negatively the vibrational comfort of the vehicle if proper values for the suspension damping and stiffness are selected. The standard deviation of the car body vertical acceleration is minimised considering relatively low values of stiffness of the suspension system, provided that constraints are respected on suspension working space.

Some of the above conclusions have been already reported in the literature, mostly in qualitative manner. Here a quantitative method to derive optimal design solutions has been presented.

12

Optimal Design of Helical Spring

Helical springs are usually made from a metal wire of circular cross-section. The use of both a hollow circular section and composite material could reduce the spring mass substantially[1]. The design of a composite material tubular helical spring is not straightforward. The definition of many parameters (design variables) is required in order to get both the desired technical specifications and the best compromise among spring objective functions. In other words, given the technical specifications, (e.g. stiffness, maximum deflection, maximum acting load, etc.), the designer is charged to find the spring geometrical and mechanical parameters (diameter of the helical coil, number of coils, number of laminae of the tubular element, orientation of fibres, etc.) in order to gain the best compromise among the many different spring objective functions (minimum mass, minimum production cost, etc.).

A successful result can be obtained by applying a proper theory allowing the designer to avoid a clumsy and time-consuming 'trial and error' approach. Multi-objective programming can be exploited to solve the addressed problem in which many design variables have to be tuned to obtain the desired performances of the system (see Sect. 2.10). To apply the addressed optimisation theory, a mathematical description of the relationships between spring design variables and system objective functions is needed. Moreover, to perform effectively the optimisation of a complex system, adequate algorithms are to be used to find the optimal solutions in a reasonable amount of time.

In the literature few papers have been published on composite material helical springs. In [112] carbon fibres helical springs were tested with the aim to replace conventional road vehicle metal helical springs. Much effort was devoted to manufacturing concepts, and the conclusion was that new springs would have been three times more expensive than the conventional ones. In [6, 207] composite material tubular helical springs were adopted to

[1] In fact a hollow circular section is particularly suited for torsional loads and a tubular shell element can be easily made from composite material

solve special problems related, respectively, to passive seismic isolation and electrical isolation.

In [207] an optimisation of the performances of the spring was performed on the basis of a constrained minimisation procedure. In [266], multi-objective programming was exploited (in conjunction with genetic algorithms) to design conventional metal helical springs with circular section, using a rather simple mechanical model.

In the present chapter, derived from [89], a number of mathematical models are developed referring to a tubular helical spring whose structure is made from multiple laminae of composite material. The models are obtained by arranging in an original way existing analytical formulations concerning stress and strain, global and local buckling and structure vibrations. Unluckily, analytical formulations, although sometimes very complex, can be applied within limited design variable ranges. This restriction could have been removed by adopting a finite element model of the spring but this would have prevented to discuss the optimisation results on the basis of analytical formulae, and general conclusions would have been missed.

The proposed optimisation can be applied to the design of conventional helical springs, by adopting the inherent simplification to the mathematical spring models developed and presented in the chapter.

12.1 Fundamentals of Optimal Design of Helical Springs and Tubular Helical Springs

In this section, the basic theory referring to the mechanical behaviour of helical springs [55], [253] is originally used to derive new formulae focusing on the advantages (or disadvantages) coming from the adoption of tubular coil springs. According to the knowledge of the authors, formulae from (12.5)–(12.30) have never been presented in the literature.

Let us assume that a conventional helical spring has to be designed. The stiffness k, the maximum deflection f_s (or the maximum load P_s), the material properties (τ_l, G, ρ) are given. Let us assume that the spring mass m and the mean helical diameter $D = 2R$, the bar diameter $d_o = 2r_o$, the number of coils i_c and the helix angle α are to be found (α will be assumed to be small). Let us assume that there are generally no constraints on the volume engaged by the spring.

Let

$$k = \frac{G\, d_o^4}{8\, D^3\, i_c} \tag{12.1}$$

$$\tau_l = \frac{8\, P_s\, D}{\pi\, d_o^3} \tag{12.2}$$

$$f_s = \pi\, D\, i_c\, \alpha - d_o\, i_c \tag{12.3}$$

and, assuming the mass of the spring

$$m = \frac{\pi^2 d_o^2 D i_c \rho}{4} \tag{12.4}$$

after some proper substitutions, setting $P_s = k f_s$, the following equation relating m, k, τ_l, f_s can be derived

$$m = \frac{2 f_s^2 G \rho}{\tau_l^2} k \tag{12.5}$$

The spring mass is defined if the stiffness k, the maximum deflection f_s and the material properties G, τ_l, ρ are given and it does not depend directly on spring parameters (D, d_o, α, i_c). This relationship is general and is valid for every helical spring with circular section (bar), provided that k, τ_l, f_s, m can be expressed in a simple form as above.

Solving Eqs. (12.1)–(12.3) with respect to D, i_c, α, one obtains

$$D = \frac{{d_o}^3 \pi \tau_l}{8 f_s k} \tag{12.6}$$

$$i_c = \frac{64 f_s^3 G k^2}{{d_o}^5 \pi^3 {\tau_l}^3} \tag{12.7}$$

$$\alpha = \frac{8 f_s k}{{d_o}^2 \pi^2 \tau_l} + \frac{{d_o}^2 \pi {\tau_l}^2}{8 f_s G k} \tag{12.8}$$

These equations depend obviously on d_o. Actually, three equations (12.1)–(12.3) are available to find four spring parameters (D, i_c, α, d_o). In order to provide one additional equation to solve the problem and thus setting d_o, the minimum of the free spring length $l = \pi D i_c \alpha$ can be searched for. It turns out that the absolute minimum of $l(d_o)$ is reached for $d_o \to \infty$. Thus, to keep l to the minimum, d_o has to be set at the maximum value.

If l is given, remembering that the maximum deflection can be expressed as

$$f_s = l - i_c d_o \tag{12.9}$$

the following expressions for D, d_o, i_c and α can be found by proper substitutions

$$D = \frac{2\sqrt{2} {f_s}^{\frac{5}{4}} G^{\frac{3}{4}} \sqrt{k}}{(-f_s + l)^{\frac{3}{4}} \pi^{\frac{5}{4}} {\tau_l}^{\frac{5}{4}}} \tag{12.10}$$

$$i_c = \frac{(-f_s + l)^{\frac{5}{4}} \pi^{\frac{3}{4}} {\tau_l}^{\frac{3}{4}}}{2\sqrt{2} {f_s}^{\frac{3}{4}} G^{\frac{1}{4}} \sqrt{k}} \tag{12.11}$$

$$\alpha = \frac{l}{\pi D i_c} = l\sqrt{\frac{\tau_l}{\pi f_s (l - f_s) G}} \tag{12.12}$$

and

$$d_o = \frac{2\sqrt{2} {f_s}^{\frac{3}{4}} G^{\frac{1}{4}} \sqrt{k}}{(-f_s + l)^{\frac{1}{4}} \pi^{\frac{3}{4}} \tau^{\frac{3}{4}}} \tag{12.13}$$

D and d_o increase with the square root of the stiffness k. One may verify that d_o/D does not actually depend on k

$$\frac{d_o}{D} = \sqrt{\frac{(-f_s + l)\,\pi\,\tau_l}{f_s\,G}} \tag{12.14}$$

For tubular helical springs, given k, f_s, τ_l (and G, ρ), the derivation of five spring parameters D, d_o, d_i, i_c, α is to be performed on the basis of only three equations, namely (12.15)–(12.17). The problem could be solved by imposing that optimal tubular helical springs should always have minimum free length and minimum mass. This would give two additional equations for finding D, i_c, α, d_o, d_i. This, in general, should lead to multiple solutions, actually the minimisation of two functions is a multi-objective programming problem which, as a rule, has multiple solutions. For the tubular spring

$$k = \frac{G\,(d_o^4 - d_i^4)}{8\,D^3\,i_c} \tag{12.15}$$

$$\tau_l = \frac{8\,P_s\,D\,d_o}{\pi\,(d_o^4 - d_i^4)} \tag{12.16}$$

$$f_s = \pi\,D\,i\,\alpha - d_o\,i_c \tag{12.17}$$

solving the above equations with respect to D, i_c, α, one obtains

$$D = \frac{\left({d_o}^4 - {d_i}^4\right)\,\pi\,\tau_l}{8\,d_o\,f_s\,k} \tag{12.18}$$

$$i_c = \frac{64\,{d_o}^3\,f_s^3\,G\,k^2}{\left({d_o}^4 - {d_i}^4\right)^2\,\pi^3\,{\tau_l}^3} \tag{12.19}$$

$$\alpha = \frac{64\,{d_o}^4\,f_s^2\,G\,k^2 + \pi^3\,\tau_l^3\,({d_o}^4 - {d_i}^4)^2}{8\,{d_o}^2\,({d_o}^4 - {d_i}^4)\,f_s\,G\,k\,\pi^2\,\tau_l} \tag{12.20}$$

The mass of the tubular spring is

$$m_t = \frac{\pi^2\,(d_o^2 - d_i^2)\,D\,i_c\,\rho}{4} \tag{12.21}$$

Substituting Eqs. (12.18), (12.19) into (12.21)

$$m_t = \frac{2\,{d_o}^2\,f_s^2\,G\,\rho\,k}{\left({d_o}^2 + {d_i}^2\right)\,{\tau_l}^2} \tag{12.22}$$

If d_i, d_o are finite and $d_i \to d_o$, it follows that $m_t \to f_s^2\,G\,\rho\,k/\,\tau_l^2$, keeping f_s, τ_l and k constant. In this case the ratio between the mass of the tubular spring m_t and the mass of the conventional spring m becomes

$$\lim_{d_i \to d_o} \frac{m_t}{m} = \frac{1}{2} \tag{12.23}$$

This ratio represents a theoretical limit for m_t/m.

If l, τ_l, k and f_s are imposed,

$$m_t = \rho\, f_s^{1/2}\, G^{1/2}\, \frac{-\sqrt{a} + \sqrt{b} + \sqrt{a+b}}{4\tau_l^2} \tag{12.24}$$

where $a = d_i^4\,(-f_s + l)\pi^3\tau_l^3$, $b = 16\, f_s^3\, G\, k^2$. Function $m_t = m_t(d_i)$ does not have a minimum as it continuously decreases as $d_i \to \infty$. The expressions of $D(d_i)$, $d_o(d_i)$, $i_c(d_i)$, α are

$$D = \frac{\sqrt{4\, {f_s}^{\frac{5}{2}}\, G^{\frac{3}{2}}\, k \;+ \sqrt{{f_s}^2\, G^2\, \left(16\, {f_s}^3\, G\, k^2 + {d_i}^4\,(-f_s + l)\,\pi^3\,{\tau_l}^3\right)}}}{(-f_s + l)^{\frac{3}{4}}\, \pi^{\frac{5}{4}}\, {\tau_l}^{\frac{5}{4}}} \tag{12.25}$$

$$d_o = \frac{\sqrt{4\, {f_s}^{\frac{5}{2}}\, G^{\frac{3}{2}}\, k \;+ \sqrt{{f_s}^2\, G^2\, \left(16\, {f_s}^3\, G\, k^2 + {d_i}^4\,(-f_s + l)\,\pi^3\,{\tau_l}^3\right)}}}{\sqrt{f_s}\,\sqrt{G}\,(-f_s + l)^{\frac{1}{4}}\, \pi^{\frac{3}{4}}\, {\tau_l}^{\frac{3}{4}}} \tag{12.26}$$

$$i_c = \frac{\sqrt{f_s}\,\sqrt{G}\;(-f_s + l)^{\frac{5}{4}}\;\pi^{\frac{3}{4}}\,{\tau_l}^{\frac{3}{4}}}{\sqrt{4\, {f_s}^{\frac{5}{2}}\, G^{\frac{3}{2}}\, k \;+ \sqrt{{f_s}^2\, G^2\, \left(16\, {f_s}^3\, G\, k^2 + {d_i}^4\,(-f_s + l)\,\pi^3\,{\tau_l}^3\right)}}} \tag{12.27}$$

$$\alpha = \frac{l}{\pi\, D\, i_c} = l\sqrt{\frac{\tau_l}{\pi f_s (l - f_s) G}} \tag{12.28}$$

Equation (12.28) is equal to (12.12) and does not depend on d_i.

As indicated above, the inner diameter d_i should always be as large as possible to make vanishing the mass of the tubular spring m_t (if l is given). In this case, the minimum mass is obtained for infinite values of d_i, d_o ($d_i < d_0$), i.e. $d_i \to d_o \to \infty$. Additionally, $D \to \infty$ (see Eq. (12.25)) and $i_c \to 0$ (see Eq. (12.27)).

By inspection of (12.26) one may verify that $d_o(d_i) > d_i$ and, moreover

$$\lim_{d_i \to \infty} \frac{d_o}{d_i} = 1 \tag{12.29}$$

Surprisingly, for the tubular spring

$$\frac{d_o}{D} = \sqrt{\frac{(-f_s + l)\,\pi\,\tau_l}{f_s\, G}} \tag{12.30}$$

i.e. the ratio d_o/D is the same as computed for the conventional spring (see Eq. (12.14)).

Optimal tubular springs (minimum mass) are thin-walled. The thickness of the tubular element should be as low as possible, provided that local stability is attained. The limitation depending on local stability is not dealt with in this section.

12.2 Composite Tubular Spring Models

The mechanical and mathematical models of a composite helical spring with hollow (tubular) circular section are introduced. The spring structure is made from multiple laminae of composite material, as shown in Fig. 12.1. Obviously, conventional springs (made from homogeneous and isotropic material and with non-hollow sections) can be easily modeled by adopting the proper simplifications to the following theory. The adopted nomenclature accomplishes with SAE recommended terminology (see [122]).

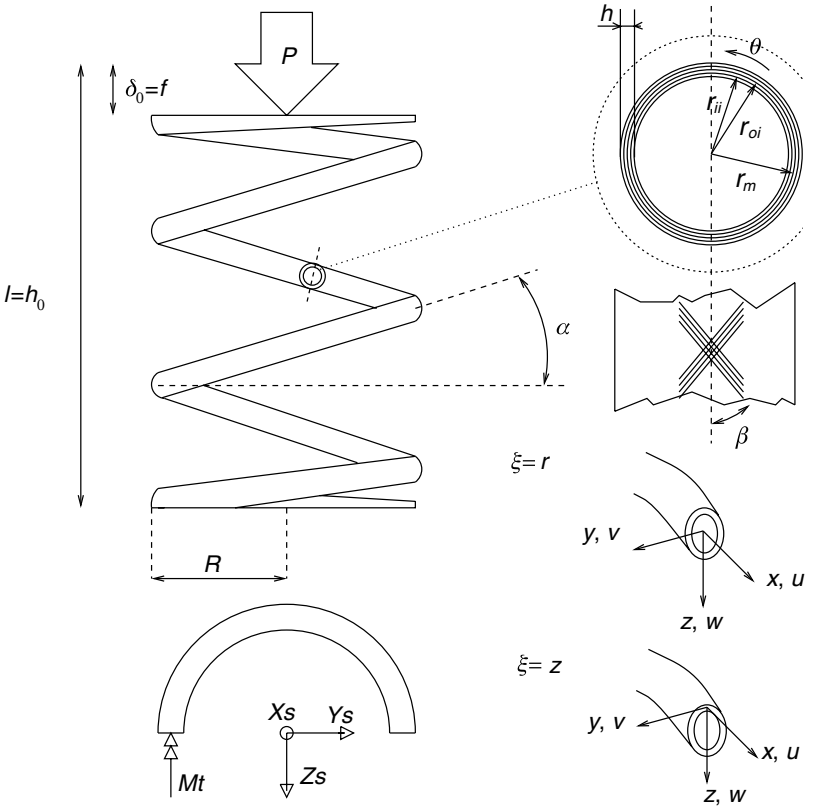


Fig. 12.1. Helical spring made from a tubular element composed of many laminae

The strains may be written in modified Donnel form [53, 135, 263] as

$$\begin{cases} \epsilon_x = \epsilon_x^0 + \xi\,\phi_x \\ \epsilon_y = \epsilon_y^0 + \xi\,\phi_y \\ \gamma \;\; = \gamma^0 + \xi\,\phi \end{cases} \tag{12.31}$$

where

$$\begin{array}{ll} \epsilon_x^0 = \frac{\partial u}{\partial x} & \phi_x = -\frac{\partial^2 w}{\partial x^2} \\ \epsilon_y^0 = \frac{\partial v}{\partial y} - \frac{w}{r} & \phi_y = -\frac{\partial^2 w}{\partial y^2} \\ \gamma^0 = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} & \phi = -2\frac{\partial w^2}{\partial x \partial y} \end{array} \tag{12.32}$$

$\xi = r$ when the external loads cause deformations of the spring for which the tubular element keeps the symmetry with respect to its axis, and $\xi = z$ in other cases (see, e.g. the section on local stability).

More complex expressions of strains (complete form) may be found in [26, 135, 263]. Here simplified expressions have been adopted [15], as they are considered sufficiently accurate for the present engineering application (see Sect. 12.3 on model validation).

12.2.1 Stress, Strain and Spring Stiffness

For the computation of the stress, strain and stiffness of the spring the expressions (12.32) can be further simplified. Remembering that $\frac{\partial v}{\partial x} = \frac{\partial \theta r}{\partial x}$ ($dv = r\, d\theta$, Fig. 12.1) and $v = v(x)$

$$\begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma \end{Bmatrix} = \begin{pmatrix} 1 & 0 & 0 & -r \\ 0 & -1/r & 0 & 0 \\ 0 & 0 & r & 0 \end{pmatrix} \begin{Bmatrix} \frac{\partial u}{\partial x} \\ w \\ \frac{\partial \theta}{\partial x} \\ \frac{\partial^2 w}{\partial x^2} \end{Bmatrix} \tag{12.33}$$

In matrix form the above equations read $\boldsymbol{\epsilon} = \mathbf{T}\mathbf{u}^*$ where $\mathbf{u}^*$ is the vector of generalised local displacements.

For plane stress $\boldsymbol{\sigma} = \mathbf{Q}\boldsymbol{\epsilon}$ where

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau \end{Bmatrix} = \begin{pmatrix} q_{11} & q_{12} & q_{16} \\ q_{21} & q_{22} & q_{26} \\ q_{61} & q_{62} & q_{66} \end{pmatrix} \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma \end{Bmatrix} \tag{12.34}$$

Matrix $\mathbf{Q}$ is full because in general the orthotropic axes are not oriented as the local reference axes.

Referring to Fig. 12.1 the relationship between the stresses along the orthotropy planes ($\boldsymbol{\sigma}_O$) and the stresses related to the local axes is $\boldsymbol{\sigma} = \mathbf{T}_m \boldsymbol{\sigma}_O$. with $\boldsymbol{\sigma}_O = \mathbf{Q}_O \boldsymbol{\epsilon}_O$, $\boldsymbol{\sigma} = \mathbf{Q}\boldsymbol{\epsilon}$, the stiffness matrix $\mathbf{Q}$ can be written as

$$\mathbf{Q} = \mathbf{T}_m \mathbf{Q}_O \mathbf{T}_m^T \tag{12.35}$$

where

$$\mathbf{Q}_O = \begin{pmatrix} E_x/\Delta & \nu_{yx}E_x/\Delta & 0 \\ \nu_{xy}E_y/\Delta & E_y/\Delta & 0 \\ 0 & 0 & G_{xy} \end{pmatrix} \tag{12.36}$$

where $\Delta = 1 - \nu_{xy}\nu_{yx}$ and $\nu_{xy}\, E_y = \nu_{yx}\, E_x$.

$$\mathbf{T}_m = \begin{pmatrix} n^2 & m^2 & 2nm \\ m^2 & n^2 & -2nm \\ -nm & nm & n^2 - m^2 \end{pmatrix} \tag{12.37}$$

where $n = \cos\beta$, $m = \sin\beta$

Combining (12.34) and (12.33) the relationship between stresses and generalised local displacements can be obtained

$$\boldsymbol{\sigma} = \mathbf{QTu}^* \tag{12.38}$$

In a single generic lamina i of the spring (inner radius r_{ii}, outer radius r_{oi}, see Fig. 12.1), the axial load in x-direction N_{xi}, the 'Mariotte' load per unit length of the tubular bar Q_{mi}, the torsion moment M_{ti}, the bending moment M_{byi} read respectively[2]

$$\begin{aligned} N_{xi} &= \int_{r_{ii}}^{r_{oi}} \int_0^{2\pi} \sigma_x r \,\mathrm{d}\theta \mathrm{d}r \\ Q_{mi} &= \int_{r_{ii}}^{r_{oi}} \sigma_t(\theta_0)\mathrm{d}r + \int_{r_i}^{r_{oi}} \sigma_t(\theta_0 + 2\pi)\mathrm{d}r \\ M_{ti} &= \int_{r_{ii}}^{r_{oi}} \int_0^{2\pi} \tau\, r r \mathrm{d}\theta \mathrm{d}r \\ M_{byi} &= \int_{r_{ii}}^{r_{oi}} \int_0^{2\pi} \sigma_x\, r r \mathrm{d}\theta \mathrm{d}r \end{aligned} \tag{12.39}$$

where $\sigma_t = \sigma_y$ (Fig. 12.1, case $\xi = z$). The shear loading can be computed approximately by introducing a correction factor [55]. Q_{mi} has been introduced for a possible further development (see [207])[3]. Combining (12.38) and (12.39) and solving the integrals in (12.39), the following system of equations is obtained:

$$\mathbf{n}_i^* = \mathbf{H}_i \mathbf{u}^* \tag{12.40}$$

where $\mathbf{n}_i^{*T} = \{N_{xi}\, \pi Q_{mi}\, M_{ti}\, M_{byi}\}$ and $\mathbf{H}_i$ reads

$$\pi \begin{pmatrix} q_{11i}(r_{oi}^2 - r_{ii}^2) & 2q_{12i}(r_{oi} - r_{ii}) & \frac{2}{3}q_{16i}(r_{oi}^3 - r_{ii}^3) & 0 \\ 2q_{21i}(r_{oi} - r_{ii}) & 2q_{22i}\log(\frac{r_{oi}}{r_{ii}}) & q_{26i}(r_{oi}^2 - r_{ii}^2) & 0 \\ \frac{2}{3}q_{61i}(r_{oi}^3 - r_{ii}^3) & q_{62i}(r_{oi}^2 - r_{ii}^2) & \frac{1}{2}q_{66i}(r_{oi}^4 - r_{ii}^4) & 0 \\ 0 & 0 & 0 & \frac{1}{4}q_{11i}(r_{oi}^4 - r_{ii}^4) \end{pmatrix} \tag{12.41}$$

The above matrix is symmetrical as $q_{ij} = q_{ji}$. The loads into the spring thickness are

$$\begin{Bmatrix} N_x \\ \pi Q_m \\ M_t \\ M_{by} \end{Bmatrix} = \begin{Bmatrix} \sum_{i=1}^{n_l} N_{xi} \\ \sum_{i=1}^{n_l} \pi Q_{mi} \\ \sum_{i=1}^{n_l} M_{ti} \\ \sum_{i=1}^{n_l} M_{byi} \end{Bmatrix} \tag{12.42}$$

[2]Should the computation of the shear be performed by using Eq. (12.67)

[3]Future study: Springs composed of two or more tubular elements revolving independently around their axes and subject to inner pressure for controlling damping due to shear resistance forces among the tubular elements

where n_l is the number of laminae the structure of the spring is composed of. By inspection of matrix $\mathbf{H}_i$ one may notice that M_{ti} is coupled with N_{xi}, Q_{mi}. The coupling depends on the fact that matrix $\mathbf{Q}$ is full. M_{byi} is decoupled from M_{ti}, N_{xi}, Q_{mi}, due to model formulation (see Eqs. (12.31)–(12.33)). In matrix form the above equations read

$$\mathbf{n}^* = \mathbf{K}\mathbf{u}^* \tag{12.43}$$

where $\mathbf{K} = \sum_{i=1}^{n_l} \mathbf{H}_i$.

If the external loads acting on the spring are known, by inverting Eq. (12.43), the generalised displacements can be computed. The deformations $\boldsymbol{\epsilon}$ (see Eq. (12.33)) and the stresses $\boldsymbol{\sigma}$ (see (12.34)) can also be derived.

If the tubular element is composed of two laminae whose fibres are oriented symmetrically with respect to x-axis (see Fig. 12.1) the elements of matrix $\mathbf{Q}$ read

$$\begin{aligned} q_{11+} &= q_{11-} \\ q_{22+} &= q_{22-} \\ q_{12+} &= q_{12-} \\ q_{16+} &= -q_{16-} \\ q_{26+} &= -q_{26-} \\ q_{66+} &= q_{66-} \end{aligned} \tag{12.44}$$

where $q_{ij\pm}$ refer to the laminae with the fibres oriented as $\pm\beta$, respectively. As a consequence, matrix $\mathbf{K}$ (12.43) reads

$$\pi \begin{pmatrix} q_{11}(r_o^2 - r_i^2) & 2q_{12}(r_o - r_i) & \frac{2}{3}q_{16}(r_{m\,cub}) & 0 \\ 2q_{21}(r_o - r_i) & 2q_{22}\log(\frac{r_o}{r_i}) & q_{26}(r_{m\,quad}) & 0 \\ \frac{2}{3}q_{61}(r_{m\,cub}) & q_{62}(r_{m\,quad}) & \frac{1}{2}q_{66}(r_o^4 - r_i^4) & 0 \\ 0 & 0 & 0 & \frac{1}{4}q_{11}(r_o^4 - r_i^4) \end{pmatrix} \tag{12.45}$$

where

$$r_{m\,quad} = r_o^2 - 2r_m^2 + r_i^2$$
$$r_{m\,cub} = r_o^3 - 2r_m^3 + r_i^3$$

Since $r_o^2 - 2r_m^2 + r_i^2 \cong 0$ for small thickness helical springs (see Fig. 12.1), the above matrix can be simplified and it follows that M_{ti} decouples from N_{xi}, Q_{mi} and M_{byi}. Only N_{xi} and Q_{mi} are coupled.

The curvature effect of the coil cannot be neglected for computing the maximum torsional stress τ_{max}. According to [55, 253] (after some proper arrangements)

$$\tau_{max} = C_f \times \frac{16M_t}{\pi d_o^3(1 - a^4)} \tag{12.46}$$

where $a = r_i/r_o$, $d_o = 2r_o$ and C_f is a factor which accounts for the curvature effect

$$C_f = \frac{4c - (1 + a^2)}{4(c - 1)} \tag{12.47}$$

where $c = R/r_o$.

The stiffness of the spring can be computed by equating the external work and the internal work. P is the external load acting along the axis of the helical spring, f the corresponding deflection, L the length of the bar, i the ith lamina

$$\begin{aligned} P\,f &= \textstyle\sum_{i=1}^{n_l} \int_0^L \int_{r_{ii}}^{r_{oi}} \int_0^{2\pi} \boldsymbol{\epsilon}^T \boldsymbol{\sigma} r \mathrm{d}\theta \mathrm{d}r \mathrm{d}x = \\ &= \textstyle\sum_{i=1}^{n_l} \int_0^L \int_{r_{ii}}^{r_{oi}} \int_0^{2\pi} \mathbf{u^*}^T \mathbf{T}^T \mathbf{Q} \mathbf{T} \mathbf{u}^* r \mathrm{d}\theta \mathrm{d}r \mathrm{d}x \end{aligned} \tag{12.48}$$

$\mathbf{u}^*$ being independent of x (away from the end coils), and observing that $\mathbf{H} = \int_{r_{ii}}^{r_{oi}} \int_0^{2\pi} \mathbf{T}^T \mathbf{Q} \mathbf{T}\, r \mathrm{d}\theta\, \mathrm{d}r$ the above equation reads

$$P\,f = \textstyle\sum_{i=1}^{n_l} \mathbf{u^*}^T \int_0^L \mathbf{H}_i \mathrm{d}x\, \mathbf{u}^* \tag{12.49}$$

Solving the integral, we get

$$P\,f = \sum_{i=1}^{n_l} \mathbf{u^*}^T \mathbf{H}_i \mathbf{u}^* L = \mathbf{u^*}^T \mathbf{K} \mathbf{u}^* L = \mathbf{u^*}^T \mathbf{n}^* L \tag{12.50}$$

If N, Q_m, M_{by} are neglected, only $M_t \cong PR$ acts (R, radius of the helical coil), i.e.

$$\mathbf{n}^*_{M_t}{}^T = \{0\,0\,M_t\,0\}^T \tag{12.51}$$

Thus inverting Eq. (12.43) and setting $\mathbf{J} = \mathbf{K}^{-1}$, vector $\mathbf{u}^*$ can be computed

$$\mathbf{u}^*_{M_t}{}^T = \left\{ \frac{\partial u}{\partial x}, w, \frac{\partial \theta}{\partial x}, \frac{\partial^2 w}{\partial x^2} \right\}^T_{M_t} = \{ j_{13} PR, j_{23} PR, j_{33} PR, j_{43} PR \}^T \tag{12.52}$$

Inserting $\mathbf{u}^*_{M_t}{}^T$ into (12.50), the spring deflection results $f = j_{33} PR^2 L$, and correspondingly, the stiffness is (remembering that $L \cong 2\pi R\, i_c$, i_c = number of coils)

$$k = \frac{1}{2\pi j_{33} R^3 i_c} \tag{12.53}$$

The spring stiffness is proportional to $1/j_{33}$. After some computations

$$\frac{1}{j_{33}} = k_{33} + \frac{-k_{13}^2\, k_{22} + 2k_{12}\, k_{13}\, k_{23} - k_{11}\, k_{23}^2}{-k_{12}^2 + k_{11}\, k_{22}} \cong k_{33} \tag{12.54}$$

k_{33} is directly related to h_{33} which depends directly on q_{66}. By varying the orientation of the orthotropy planes q_{66} varies, so different spring stiffness k can be obtained.

For conventional helical springs with circular bar section $q_{16} = q_{26} = k_{13} = k_{23} = 0$, $q_{66} = G$, $r_i = 0$, so $\frac{1}{j_{33}} = k_{33}$ and $k = k_{33}/(2\pi R^3 i_c) = Gr^4/(4R^3 i_c)$ which is the well-known formula for computing the stiffness of torsion springs. If the tubular element is composed of two laminae oriented symmetrically with respect to the x direction (see Fig. 12.1), the spring stiffness becomes $k = (q_{66}(r_o^4 - r_i^4))/(4R^3 i_c)$.

12.2.2 Global Stability

The global stability of the studied spring can be described mathematically according to [15].

The following expression for the critical load can be used:

$$\frac{P_c}{\alpha_0} = \frac{\delta_c}{l} = \frac{1 - \sqrt{1 - 4\pi^2 \frac{W_0^*}{(\chi\, l)^2 \alpha_0}(1 - \frac{\alpha_0}{T_0^*})}}{2(1 - \frac{\alpha_0}{T_0^*})} \tag{12.55}$$

$$\alpha_0 = \frac{G_Q\, J_p\, l}{2\pi\, R^3\, i_c}$$

where δ_c is the deflection at $P = P_c$, J_p is the polar moment of inertia of the tubular section, G_Q is the shear modulus which reads $G_Q = 1/n_l \sum_i^{n_l} q_{66i}$ and

$$W_0^* = \frac{E_Q J_n}{\frac{1}{2} \times (1 + \frac{E_Q J_n}{G_Q J_p})} \frac{l}{2\pi R i_c}$$

$$T_0^* = \frac{E_Q J_b l}{i\pi R^3}$$

where $J_n\,(= J_b)$ is the bending moment of inertia of the tubular section ($J_n = J_b = 0.5\, J_p$ for circular sections), E_Q is the Young modulus which reads $E_Q = 1/n_l \sum_i^{n_l} q_{11i}$ and χ is a parameter which takes into account the constraints at the ends of the spring (e.g. $\chi = 1/2$ for the considered fixed-slider constraints).

12.2.3 Local Stability

The local stability of tubular helical springs refers to the buckling of the tubular element. The problem of buckling of tubular shells under compression and bending loads has been dealt with in [13, 14, 263]. A solution to the problem of the buckling behaviour of laminated cylindrical shell under torsion based on higher order shear deformation theory has been reported in [242]. Similar research works are reported in [240, 243, 254] considering combined loading conditions.

The simplified theory presented in [263] has been extended to the problem of buckling under torsion.

Let us now consider the case in which only the load P is acting, producing a torque $M_t = PR\cos\alpha$. We will consider that the spring is loaded by M_t only (the bending, shear and compression caused by P will be neglected). This assumption has been made for two reasons. The first is that experimentally (see Sect. 12.3) the buckling seems to be well predicted under this hypothesis.

The second is that, according to the knowledge of the authors, there are not well-established criteria pre-defining the deformed shapes after buckling when the tubular element is contemporarily bent, compressed and subjected to

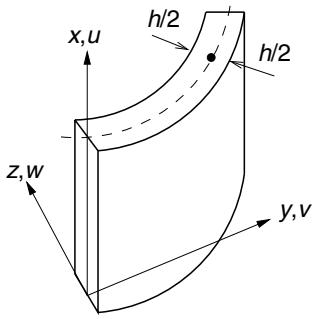


Fig. 12.2. Cylindrical shell

torque. For the post-buckling deformation (see Fig. 12.2) we consider solutions of the form [14]

$$\begin{cases} u_0 = A\left(\cos\left(m\frac{\pi\, x}{L}\right)\cos\left(n\frac{\pi\, y}{l_r}\right) + \sin\left(m\frac{\pi\, x}{L}\right)\sin\left(n\frac{\pi\, y}{l_r}\right)\right) \\ v_0 = B\left(\cos\left(m\frac{\pi\, x}{L}\right)\cos\left(n\frac{\pi\, y}{l_r}\right) + \sin\left(m\frac{\pi\, x}{L}\right)\sin\left(n\frac{\pi\, y}{l_r}\right)\right) \\ w = C\left(\cos\left(m\frac{\pi\, x}{L}\right)\sin\left(n\frac{\pi\, y}{l_r}\right) - \sin\left(m\frac{\pi\, x}{L}\right)\cos\left(n\frac{\pi\, y}{l_r}\right)\right) \end{cases} \tag{12.56}$$

where $l_r = 2\pi\, r$.

The assumed deflection does not satisfy any simple boundary conditions at the tubular element end. For this reason we assume that the deformation mode pertains to the central portion of a long cylinder that is little affected by the end conditions.

The governing equations for the buckling analysis of a laminated cylindrical shell under torsion may be expressed as (according to Donnell formulation)

$$\begin{array}{l} \frac{\partial N_{x1}}{\partial x} + \frac{\partial N_{xy1}}{\partial y} = 0 \\ \frac{\partial N_{xy1}}{\partial x} + \frac{\partial N_{y1}}{\partial y} = 0 \\ \frac{\partial^2 M_{x1}}{\partial x^2} + 2\frac{\partial^2 M_{xy}}{\partial x \partial y} + \frac{\partial^2 M_{y1}}{\partial y^2} + \frac{N_{y1}}{r} - 2N_{xy0}\frac{\partial^2 w_1}{\partial x \partial y} = 0 \end{array} \tag{12.57}$$

where the subscript 0 refers to values before buckling and 1 to changes during buckling and

$$N_x = \int_{-h/2}^{h/2} \sigma_x\, dz \quad N_y = \int_{-h/2}^{h/2} \sigma_y\, dz \quad N_{xy} = \int_{-h/2}^{h/2} \tau_{xy}\, dz$$

$$M_x = \int_{-h/2}^{h/2} \sigma_x\, z dz \quad M_y = \int_{-h/2}^{h/2} \sigma_y\, z dz \quad M_{xy} = \int_{-h/2}^{h/2} \tau_{xy} z\, dz$$

The governing buckling equation can be rewritten in terms of middle surface displacement components as

$$\begin{pmatrix} \mathcal{L}f_{11} & \mathcal{L}f_{12} & \mathcal{L}f_{13} \\ & \mathcal{L}f_{22} & \mathcal{L}f_{23} \\ \text{symm.} & & \mathcal{L}f_{33} \end{pmatrix} \begin{Bmatrix} u_0 \\ v_0 \\ w_0 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix} \tag{12.58}$$

where $\mathcal{L}f$ are functional operators.

The set of differential equations (12.58) have to be zero at each position of the cylinder, we apply Galerkin's procedure to Donnel-type equation as follows:

$$\int_V \delta\, \mathbf{u}^\star\, \mathbf{L}\, \mathbf{u}^\star\, dV = 0 \tag{12.59}$$

where

$$\delta \mathbf{u}^{\star T} = \{\delta u_0, \delta v_0, \delta w_0\} \tag{12.60}$$

(12.59) can be written explicitly as

$$\begin{aligned} \int_0^L \int_0^{2\pi r} \{ &(\delta A) \left(\cos\left(m\tfrac{\pi\, x}{L}\right) \cos\left(n\tfrac{\pi\, y}{l_r}\right) + \sin\left(m\tfrac{\pi\, x}{L}\right) \sin\left(n\tfrac{\pi\, y}{l_r}\right)\right) \\ &[\mathcal{L}f_{11}(u_0) + \mathcal{L}f_{12}(v_0) + \mathcal{L}f_{13}(w_0)] \\ +&(\delta B) \left(\cos\left(m\tfrac{\pi\, x}{L}\right) \cos\left(n\tfrac{\pi\, y}{l_r}\right) + \sin\left(m\tfrac{\pi\, x}{L}\right) \sin\left(n\tfrac{\pi\, y}{l_r}\right)\right) \\ &[\mathcal{L}f_{21}(u_0) + Lf_{22}(v_0) + \mathcal{L}f_{23}(w_0)] \\ +&(\delta C) \left(\cos\left(m\tfrac{\pi\, x}{L}\right) \sin\left(n\tfrac{\pi\, y}{l_r}\right) - \sin\left(m\tfrac{\pi\, x}{L}\right) \cos\left(n\tfrac{\pi\, y}{l_r}\right)\right) \\ &[\mathcal{L}f_{31}(u_0) + \mathcal{L}f_{32}(v_0) + \mathcal{L}f_{33}(w_0)]\} \mathrm{d}x\, \mathrm{d}y = 0 \end{aligned} \tag{12.61}$$

(12.61) should be valid for any variations δA, δB, δC which yields the following set of equations

$$\begin{aligned} \int_0^L \int_0^{2\pi r} &\left(\cos\left(m\tfrac{\pi\, x}{L}\right) \cos\left(n\tfrac{\pi\, y}{l_r}\right) + \sin\left(m\tfrac{\pi\, x}{L}\right) \sin\left(n\tfrac{\pi\, y}{l_r}\right)\right) \\ &[\mathcal{L}f_{11}(u_0) + \mathcal{L}f_{12}(v_0) + \mathcal{L}f_{13}(w_0)]\, \mathrm{d}x\, \mathrm{d}y = 0 \end{aligned} \tag{12.62}$$

$$\begin{aligned} \int_0^L \int_0^{2\pi r} &\left(\cos\left(m\tfrac{\pi\, x}{L}\right) \cos\left(n\tfrac{\pi\, y}{l_r}\right) + \sin\left(m\tfrac{\pi\, x}{L}\right) \sin\left(n\tfrac{\pi\, y}{l_r}\right)\right) \\ &[\mathcal{L}f_{21}(u_0) + \mathcal{L}f_{22}(v_0) + \mathcal{L}f_{23}(w_0)]\, \mathrm{d}x\, \mathrm{d}y = 0 \end{aligned} \tag{12.63}$$

$$\begin{aligned} \int_0^L \int_0^{2\pi r} &\left(\cos\left(m\tfrac{\pi\, x}{L}\right) \sin\left(n\tfrac{\pi\, y}{l_r}\right) - \sin\left(m\tfrac{\pi\, x}{L}\right) \cos\left(n\tfrac{\pi\, y}{l_r}\right)\right) \\ &[\mathcal{L}f_{31}(u_0) + \mathcal{L}f_{32}(v_0) + \mathcal{L}f_{33}(w_0)]\, \mathrm{d}x\, \mathrm{d}y = 0 \end{aligned} \tag{12.64}$$

After integration, (12.62)–(12.64) lead to these three linear homogeneous algebraic equations

$$\begin{pmatrix} t_{11} & t_{12} & t_{13} \\ & t_{22} & t_{23} \\ \text{symm.} & & t_{33} \end{pmatrix} \begin{Bmatrix} A \\ B \\ C \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix} \tag{12.65}$$

The existence of a non-trivial solution of (12.65) implies the vanishing of the determinant of the matrix $\mathbf{T}$. Buckling occurs when, increasing N_{xy}, $\det(\mathbf{T}(N_{xy})) = 0$, the corresponding load is the critical one (N_{xycr}). The

expression of $N_{xy\,cr}$ has been derived symbolically as function of n, m, a_{kj}, b_{kj}, d_{kj}, where

$$\begin{array}{l} a_{kj} = \sum_i^{n_l} q_{kji}(\zeta_{i+1} - \zeta_i) \\ b_{kj} = \sum_i^{n_l} \frac{1}{2} q_{kji}(\zeta_{i+1}^2 - \zeta_i^2) \\ d_{kj} = \sum_i^{n_l} \frac{1}{3} q_{kji}(\zeta_{i+1}^3 - \zeta_i^3) \end{array} \tag{12.66}$$

where $\zeta_i = r_{ii} - r$, $\zeta_{i+1} = r_{oi} - r$ (see Fig. 12.1) and $\mathbf{n}' = \mathbf{A}\,\boldsymbol{\epsilon}'$

$$\mathbf{n}' = \begin{Bmatrix} N_x \\ N_y \\ N_{xy} \\ M_x \\ M_y \\ M_{xy} \end{Bmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{16} & b_{11} & b_{12} & b_{16} \\ & a_{22} & a_{26} & b_{21} & b_{22} & b_{26} \\ & & a_{66} & b_{61} & b_{62} & b_{66} \\ & & & d_{11} & d_{12} & d_{16} \\ & \text{symm.} & & & d_{22} & d_{26} \\ & & & & & d_{66} \end{pmatrix} \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma \\ \phi_x \\ \phi_y \\ \phi \end{Bmatrix} \tag{12.67}$$

The symbolic expression of $N_{xy,cr}$ is reported in Appendix 12.6.

12.2.4 Vibrations

In Fig. 12.1 the variables defining the vibrations of a helical spring are presented, according to [204, 261]. A more refined model could have been adopted [175] but the analytical formulation of the problem would have been lost. Two motions are accounted for, namely the torsion and the bending of the tubular element. Given the shear modulus G_Q, the Young's modulus E_Q, the bending moment of inertia of the cross-section of the hollow circular section J_b, the polar moment of inertia of the cross-section of the hollow circular section J_p[4], and given the free spring length l, the length of the tubular element $L = l/\sin\alpha$, the helix angle α, the following equations may be written

$$\begin{array}{l} k = k_a = \frac{G_Q J_p \cos^2\alpha + E_Q J_b \sin^2\alpha}{LR^2} \\ k_r = \frac{G_Q J_p \sin^2\alpha + E_Q J_b \cos^2\alpha}{L} \\ k_c = \frac{(G_Q J_p - E_Q J_d)\sin\alpha\cos\alpha}{LR} \end{array} \tag{12.68}$$

which represent respectively the axial spring stiffness related to the axial deflection δ_a (k_a), the bending spring stiffness related to the rotation of the hollow circular section φ (k_r) and the coupling stiffness (k_c).

The equations of motion of the spring are then

$$\begin{cases} \frac{m}{l^2}\frac{\partial^2\delta_a}{\partial t^2} & = k_a\frac{\partial^2\delta_a}{\partial x^2} + k_c\frac{\partial^2\varphi}{\partial x^2} \\ \frac{mR^2}{l^2}\frac{\partial^2\varphi}{\partial t^2} & = k_r\frac{\partial^2\varphi}{\partial x^2} + k_c\frac{\partial^2\delta_a}{\partial x^2} \end{cases} \tag{12.69}$$

where m is the spring mass. The above equations can be rewritten in a decoupled form as

[4] G_Q, E_Q, J_b, J_p have already been defined and used in the previous section on global stability

$$\begin{cases} \frac{\partial^2 d_1}{\partial t^2} - V_1^2 \frac{\partial^2 d_1}{\partial s^2} = 0 \\ \frac{\partial^2 d_2}{\partial t^2} - V_2^2 \frac{\partial^2 d_2}{\partial s^2} = 0 \end{cases} \tag{12.70}$$

where $d_1 = \delta_a \cos\alpha + R\varphi \sin\alpha$, $d_2 = -\delta_a \sin\alpha + R\varphi \cos\alpha$, and

$$\begin{aligned} V_1^2 &= \frac{G_Q J_p L}{R^2 m} \\ V_2^2 &= \frac{E_Q J_b L}{R^2 m} \end{aligned} \tag{12.71}$$

V_1 and V_2 represent the velocity of propagation of torsional mode and flexural mode strain waves, respectively. The n_rth resonances for torsional mode and flexural mode are, respectively

$$\begin{aligned} \omega_{1n_r} &= n_r \, 2\pi V_1/(\eta L) \\ \omega_{2n_r} &= n_r \, 2\pi V_2/(\eta L) \end{aligned} \tag{12.72}$$

where η is an integer which accounts for the way the spring ends are constrained (e.g. if one end is fixed and the other one is free $\eta = 4$). If $\eta = 2$, $\omega_{1n_r}/\omega_{2n_r} = \sqrt{G_Q J_p/(E_Q J_b)}$ (for steel springs with circular bar cross-section $\omega_{1n_r}/\omega_{2n_r} = 1/1.142$). For composite material springs this resonance frequency ratio may be either <1 or >1 depending on the orientation of the orthotropy axes. Unfortunately, in the general case, the coupling of M_t and M_b (see (12.40)) makes the computation of ω_{1n}/ω_{2n} impossible by means of the above model.

For a tubular element composed of an even number n_l of laminae with orthotropy axes symmetrically oriented with respect to the x-axis, M_t and M_b are decoupled; thus, from (12.45) and (12.44)

$$\begin{aligned} V_1^2 &= \frac{\sum_i^{n_l} (q_{66+_i} + q_{66-_i}) \, J_p \, L}{R^2 \, m} \\ V_2^2 &= \frac{\sum_i^{n_l} (q_{11+_i} + q_{11-_i}) \, J_b \, L}{R^2 \, m} \end{aligned} \tag{12.73}$$

Finally, for a tubular element composed of two such laminae, i.e. $n_l = 2$, remembering (12.44), it follows

$$\frac{\omega_{1n_r}}{\omega_{2n_r}} = \sqrt{\frac{2q_{66} J_p}{2q_{11} J_b}} = \sqrt{\frac{2q_{66}}{q_{11}}} \tag{12.74}$$

12.2.5 Spring Material Strength

The static strength of the helical spring is checked by means of the Tsai-Hill criterion [15, 252]:

$$\frac{\sigma_x^2}{X_t^2} - \frac{\sigma_x \sigma_y}{X_t^2} + \frac{\sigma_y^2}{Y_t^2} + \frac{\tau_{xy}^2}{S^2} \leq 1 \tag{12.75}$$

where X_t, Y_t, S are the limit stresses for σ_x (tension), σ_y (tension) and τ_{xy}, respectively. If one of the stress components (σ_x, σ_y) is compressive, then we

replace the corresponding strength quantity, X_t or Y_t with its compressive counterpart X_c or Y_c.

Tsai-Hill criterion may be reputed sufficient for a preliminary and comparative design analysis.

Unfortunately, the fatigue strength is mostly studied with reference to particular cases as it is generally recognised that a comprehensive theory cannot be applied. In the chapter, Eq. (12.75) will be used for comparative assessments of material strength.

12.3 Model Validation

A number of springs (namely eight) of different shapes have been built in order to check the responses of the theoretical models presented in the preceding sections. In Table 12.1 the mechanical characteristics of the springs are presented.

The helical springs are made from a cylindrical glass fibre woven fabric. This implies that (for such woven fabric composites) the constitutive relationships matrixes (**A**, **B**, **D**) are different from the ones given by (12.66) which were derived by the classical laminate theory.

To account for this, an improved version of the classical laminate theory had to be employed. Being the warp and the fill fibres of the same material, the formulae reported in [211] can be written as

$$\begin{aligned} a_{kj} &= 1/a^2\{\{2h_h a_u^2 + 2(2h_h - h_t)a_0 a_u + (2h_h - 2h_t)a_0^2\}Q_{kj}^M \\ &\quad + 2h_t a_0^2 Q_{kj}^W(0) + 2h_t a_0 \textstyle\int_{-a_u/2}^{a_u/2} Q_{kj}^W(\theta)\} \\ b_{kj} &= 0 \\ d_{kj} &= 1/a^2\{\{h_h^3 a_u^2 + 2h_h^3 a_0 a_u + (2h_h - 2h_t)a_0^2\}(2/3)Q_{kj}^M \\ &\quad + 2/3h_t^3 a_0^2 Q_{kj}^W(0) + 2/3a_0(2h_t^3) \textstyle\int_{-a_u/2}^{a_u/2} Q_{kj}^W(\theta)\} \end{aligned} \tag{12.76}$$

with

$$a_u = a\,(1 - h_h/h_t\; v_f/p_d)$$

where v_f is the fibres volume fraction, p_d the packing density of the fibres, a is the projected dimension of warp and fill tows on the midplane, a_0 is the non-undulated length of a ($a_0 = a - a_u$), a_u is the undulation length, h_h is the thickness of each ply, h_t is the thickness of warp and fill tows and θ is the inclination angle of the fill (warp) tows with the x-axis (y-axis) (see Fig. 12.3).

Using (12.76) the effective Young's moduli, shear modulus and Poisson's ratios can be readily calculated.

The springs have been impregnated with epoxy resin manually, after having inserted into the tubular element a thin-walled silicon tube filled with sand in order to keep the cylindrical shape during impregnation. Polymerisation has taken place on a grooved drum, machined properly to wind the

Table 12.1. Characteristics of the eight composite springs built for tests

Spring identification number	$1, 2, 3, 4$	$5, 6$	$7, 8$
Shape	Helical	Helical	Cylinder
Spring end constraints	Fixed-slider	Fixed-slider	Fixed-pin
Spring length l(m)	0.320	0.320	0.280
Helix mean diameter $2R$(m)	0.100	0.100	–
Number of coils i_c	6	6	–
Helix angle $\alpha(^\circ)$	10	10	–
Tubular element mean diameter $2r_o$(m)	0.010	0.010	0.030
Tubular element thickness h(m)	0.00025	0.001	0.0005
Type of composite		Woven fabric	
Fabric angle ($^\circ$)		± 45	
Material		Glass fibres & epoxy	
Fabric specific mass (Kg/m)	0.019	0.040	0.056
Glass fibre Young's modulus (MPa)		72,000	
Glass fibre Poisson's modulus		0.2	
Epoxy specific mass (Kg/m^3)		1,100	
Spring specific mass (Kg/m^3)		2,600	
Max tensile stress (spring material) (MPa)		50–80	
max compression stress (spring material) (MPa)		80–100	
Effective Young's modulus (E_x) (12.76) (MPa)		28,000	
Effective Young's modulus (E_y) (12.76) (MPa)		28,000	
Effective shear modulus (G) (12.76) (MPa)		9,500	
Effective Poisson's modulus (ν_{xy}) (12.76)		0.28	

helical spring. The silicon tube could not be removed, however its very low stiffness does not influence the mechanical characteristic of the spring.

The straight tubular springs have been built in a similar manner as the coiled ones.

In Fig. 12.4 the loading of the springs is shown. The results of tests that have been carried out are summarised in Tables 12.2 and 12.3.

By inspection of Tables 12.2 and 12.3 one may notice that the measured values are within a $\pm 7\%$ range with respect to the computed ones. The (small) difference between measured and computed data depends on many facts. The mechanical characteristic of the composite material are only estimated on the basis of the ratio of epoxy and glass fibre volumes. Furthermore, both the epoxy and the glass fibre mechanical parameters could not be measured directly and were assumed from the manufacturer technical specifications. The spring ends are not modeled as they are actually shaped (tangent tail). The critical load seems to be dramatically influenced by local defects from where

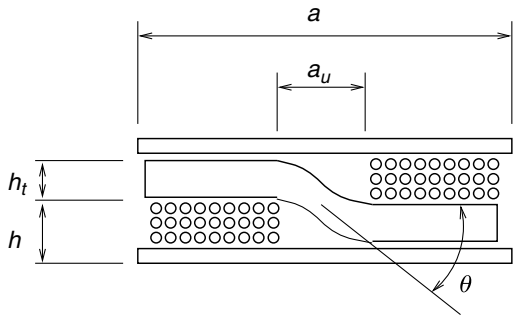


Fig. 12.3. Plain weave tow undulation definitions

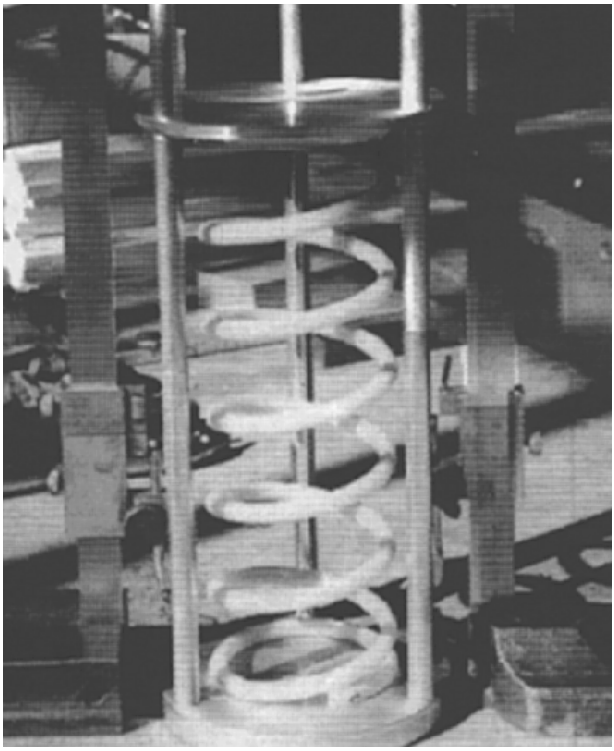


Fig. 12.4. Spring loading

unstable deformations initiate. Due to all of these reasons the obtained results seem satisfactory and the theoretical models are assumed to be adequate to perform the following comparative analyses reported in Sect. 12.4.

Table 12.2. Comparison of computed and measured data referring to tested helical springs (n.a.: not applicable, as the critical load can be measured if it is lower to the load at spring solid condition)

	Measured (mean value)	Computed	Measured (mean value)	Computed
Spring identification number	$1, 2, 3, 4$	$1, 2, 3, 4$	$5, 6$	$5, 6$
Stiffness (N/m)	698	708	1,270	1,310
Load at spring solid condition (N)	210	195	380	360
Critical load (N)	75	80	n.a.	611

Table 12.3. Comparison of computed and measured data referring to tested straight springs (torsion bars with tubular section)

	Measured (mean value)	Computed
Spring identification number	$7, 8$	$7, 8$
Stiffness (Nm/rad)	313	299
Critical load (Nm)	14	16

12.4 Numerical Application

The following application example refers to the Optimisation of a tubular section helical compression spring by using MOP.

The problem has been formulated in an academical perspective, i.e. some technological problems as well as some very peculiar technical specifications have been neglected. In other words, the number of possible combinations of different technical specifications may be so large that an ultimate Optimisation of tubular helical springs cannot be attempted. The presented application, though rather complex, has to be considered as a plain example on how optimal design can be performed with reference to the addressed type of springs.

12.4.1 Design Aims and Related Objective Functions

The following general design aims have been considered

$$\begin{gathered} \min \text{ mass} \\ \max \text{ stiffness} \end{gathered}$$

The related objective functions are defined by the following mathematical relationships. The spring mass m is

$$m = 2\pi R i_c \pi (d_o^2 - d_i^2)\rho/4 \tag{12.77}$$

where ρ is the material density.

The spring stiffness is given by (12.53) or (12.68).

The stiffness is usually given as a technical specification, and should not be maximised nor minimised. However, here it has been maximised with the only aim to build a Pareto-set which could be representative of the general technical features and mechanical properties of such type of springs.

12.4.2 Design Variables to Be Tuned

The design variables and their respective variation ranges are reported in Table 12.4. Eight design variables have been varied independently. Seven design variables were related to the spring geometry, the eighth one referred to the material (carbon–epoxy, glass–epoxy, or steel). More precisely the three considered materials (see Table 12.4) are defined by many parameters (Young's modulus, density, etc.). Changing the material involves changing at the same time many parameters (e.g. Young's modulus).

Table 12.4. Design variables and their variation ranges for the considered Optimisation problem

	Lower bound	Upper bound	# of levels
Spring free length l	$i_c\, d_o$	0.300 m	5
Helix mean diameter $2R$ (m)	0.060 m	0.120 m	8
Helix angle $\alpha(^\circ)$	5°	25°	5
Tubular element mean diameter $2r_m$ (m)	0.002 m	0.016 m	10
Tubular element thickness h (m)	0.0002 m	0.005 m	10
Number of laminae	1	6	5
Fibre angle β	0°	±45°	5
Material	Steel	Glass–epoxy	Carbon–epoxy
Young's modulus (MPa)	210,000 MPa	40,000 MPa	140,000 MPa
Poisson's modulus	0.30	0.29	0.29
Spring-specific mass (kg/m^3)	7,800	2,600	2,700
Limit stress (Tsai-Hill) X_t (MPa)	–	800	1,300
Limit stress (Tsai-Hill) X_c (MPa)	–	500	1,300
Limit stress (Tsai-Hill) Y_t (MPa)	–	70	100
Limit stress (Tsai-Hill) Y_c (MPa)	–	120	150
Limit stress (Tsai-Hill) S (MPa)	–	50	60
Limit shear stress τ_l (steel) (MPa)	660	–	–

The number of independent design variables is eight(seven referring to the spring geometry and one to the material). The lower bound for the spring free length is the solid length which is given by i_c (number of coils), times d_o (the wire or tubular element outer diameter)

12.4.3 Constraints

The constraints have to satisfy the limitations established by the following six criteria

Tsai-Hill criterion	(12.75)	
Global stability	(12.55),	$P_s \leq P_c$
Local stability	(12.78),	$P_s \leq N_{xy,cr} 2\pi r_m$
Resonance frequency	(12.72),	$\omega_{1n_r}, \omega_{2n_r} \geq \omega_{max}$
Maximum deflection	(12.53), (12.68),	$f_s \geq f_{s_0}$

where, in addition to known symbols, P_s is the load that causes the spring to reach its solid length, and f_{s_0} is a reference maximum deflection.

The first constraint establishes that the maximum allowable stress has to be reached, at most, at spring solid condition. For steel springs the first constraint becomes $\sigma_{max_V} \leq \sigma_{ref_V}$ where the subscript means that both the reference and the maximum stresses are computed according to Von Mises theory. The second and the third constraints refer to global and local stability, respectively. The fourth constraint limits the lower resonance frequency (in the example $\omega_{max} = 2\pi 50\,\text{rad/s}$). The fifth constraint establishes that the maximum deflection stroke (f_s) has to be greater than that of a reference value ($f_{s_0} = f_{s_1}$=0.2 m in Fig. 12.5). The spring to which reference was made in the application example is spring number 1 in Table 12.1.

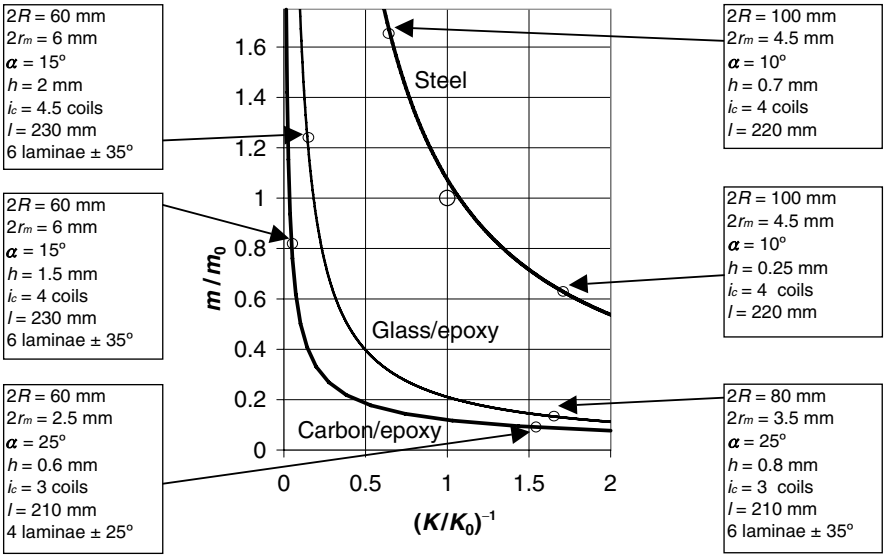


Fig. 12.5. Tubular helical compression springs: Pareto-optimal sets. Non-dimensional spring compliance ($1/k$) on the x-axis and non-dimensional spring mass on the y-axis. Values on x- and y-axes are normalised with respect to those ($1/k_0$, m_0) referring to spring no.1 in Table 12.1. Symbols and design variables' ranges in Table 12.4

12.4.4 Finding Optimal Solutions

The computation of the Pareto-set has been performed by using an exhaustive search [30,59,149,166](see Sect. 3.4.2), in conjunction with the application of the Pareto-optimality definition (2.10). No special search algorithms [149] had to be employed because the computation of the two objective functions and of the six constraints is rather simple and not time-consuming ('only' eight design variables were varied independently, see Table 12.4).

12.4.5 Analysis of the Optimal Solutions

Figure 12.5 shows the Pareto-optimal sets for the considered types of tubular section helical compression springs. Relevant improvements with respect to optimal standard steel springs can be obtained by adopting optimised composite springs. It is important to notice that the whole set of optimal steel springs do have a tubular section. A comparison between optimal steel springs with tubular and circular (i.e. bar) sections is reported in Fig. 12.6. Relevant weight reduction would be obtained for steel springs by using tubular sections. The shape of the two curves in Fig. 12.6 can be interpreted on the basis of the formulae reported in Sect. 12.1. For conventional (bar) steel springs the mass m (function of stiffness k) is well predicted by Eq. (12.5). For tubular section

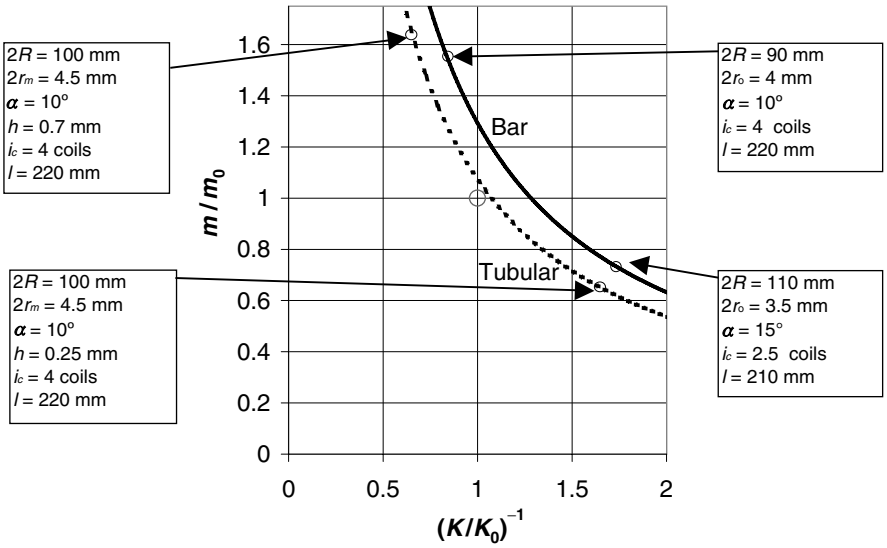


Fig. 12.6. Pareto-optimal sets: comparison between steel tubular springs and circular section (bar) steel springs. Non-dimensional spring compliance $(1/k)$ on the x-axis and non-dimensional spring mass on the y-axis. Values on x- and y-axes are normalised with respect to those $(1/k_0, m_0)$ referring to spring no.1 in Table 12.1. Symbols and design variables' ranges in Table 12.4

steel springs, as the free spring length l is constant along the Pareto-set, it turns out that, as indicated in Eq. (12.30), $r_o/R = d_o/D$ does not depend on $1/k$. Additionally, the respective values of $D = 2R$, d_o and i are also (approximately) given by Eqs. (12.25–12.27). As predicted by the theory (see Sect. 12.1), tubular steel springs are thin-walled. Their mass is between 10% and 30% lower than that of the corresponding bar springs. This mass reduction agrees with the theoretical limit of 50% reduction established by Eq. (12.23) that indeed does not account for local and/or global spring stability and thus predicts high reduction. A generalised adoption of steel springs with tubular section depends on the development of a proper technology for manufacturing them.

Coming back to the analysis of Fig. 12.5, one may notice that, for glass–epoxy springs, $D = 2R$ increases with $1/k$ and i_c decreases with $1/k$, as predicted respectively by Eqs. (12.18) and (12.19). This is also true for carbon–epoxy springs if reference is made to i_c. Figure 12.5 shows that the mass of optimal composite springs is about one order of magnitude lower than that of the corresponding steel springs. This improvement can be easily verified by evaluating the mass of the tubular spring given by (12.22)[5].

All the optimal composite springs have many laminae (4–6), generally with a fibre angle lower than 45°, and precisely about 25–35°. The number of laminae increases with the stiffness (see Table 12.6). The angle of fibre β is evidently less than the optimal for torsion (45°), in order to cope with the global stability constraint.

The helix angle is about 10° for optimal steel tubular springs and increased for optimal composite springs (15–25°). This depends on the fact that steel springs are made from a stiffer material (higher G) than are composite ones. From Eq. (12.20) one may verify that, increasing G, the helix angle α decreases.

The mass of the reference test spring (spring no. 1 in Table 12.1, represented in Fig. 12.5) could be significantly reduced by applying the presented optimisation process.

By inspection of Fig. 12.5 and Tables 12.5, 12.6 the following considerations may be drawn.

For composite Pareto-optimal springs, the stiffness, the critical compression load (global stability objective function) and the mass are strongly correlated. The same occurs substantially for Pareto-optimal steel springs. These facts can be understood a posteriori (i.e. after the Optimisation has been made and the Pareto-set has been derived!), by inspecting, respectively, Eqs. (12.53), (12.55), (12.77) and (12.5). A higher moment of inertia of the cross-section of the tubular element involves higher stiffness, critical load and mass.

[5]From Fig. 12.5, at $(k/k_0)^{-1} = 1.6$, for the carbon–epoxy spring $d_{o,ce} = 3.1\,\text{mm}$, $d_{i,ce}$=1.9 mm, for the steel tubular spring $d_{o,st} = 4.8\,\text{mm}$, $d_{i,st}$ =4.2 mm. Additionally, $G_{ce} = 20\,\text{GPa}$, $\rho_{ce} = 2,700\,\text{kg/m}^3$, τ_{ce}=510 MPa $G_{st} = 80\,\text{GPa}$, $\rho_{st} = 7,800\,\text{kg/m}^3$, $\tau_{st} = 660$ MPa; thus, from (12.22), $m_{t,st}/m_{t,ce} = 6$

Table 12.5. Significant correlations ($R_s > 0.85$) between objective functions pertaining to Pareto-optimal springs made from different materials

	Objective function	Objective function	r_s
Carb.–epoxy	Stiffness (k)	Mass (m)	1
	Stiffness (k)	Global stability (P_c)	0.98
	Global stability (P_c)	Mass (m)	0.98
	Global stability (P_c)	Res. frequency ($\omega_{1n_r}, \omega_{2n_r}$)	0.93
	Global stability (P_c)	Loc. stability ($2\pi r_m N_{xy,cr}$)	0.91
	Loc. stability ($2\pi r_m N_{xy,cr}$)	Res. frequency ($\omega_{1n_r}, \omega_{2n_r}$)	0.92
	Stiffness (k)	Loc. stability ($2\pi r_m N_{xy,cr}$)	0.87
	Stiffness (k)	Res. frequency ($\omega_{1n_r}, \omega_{2n_r}$)	0.85
Glass–epoxy	Stiffness (k)	Mass (m)	1
	Stiffness (k)	Global stability (P_c)	0.98
	Global stability (P_c)	Mass (m)	0.98
	Loc. stability ($2\pi r_m N_{xy,cr}$)	Res. frequency ($\omega_{1n_r}, \omega_{2n_r}$)	0.87
	Stiffness (k)	Loc. stability ($2\pi r_m N_{xy,cr}$)	0.86
	Mass (m)	Loc. stability ($2\pi r_m N_{xy,cr}$)	0.86
	Stiffness (k)	Res. frequency ($\omega_{1n_r}, \omega_{2n_r}$)	0.85
Steel	Stiffness (k)	Mass (m)	1
	Loc. stability ($2\pi r_m N_{xy,cr}$)	Stiffness (k)	0.90
	Loc. stability ($2\pi r_m N_{xy,cr}$)	Mass (m)	0.90
	Loc. stability ($2\pi r_m N_{xy,cr}$)	Global stability (P_c)	0.89
	Global stability (P_c)	Mass (m)	0.85
	Global stability (P_c)	Stiffness (k)	0.85

r_s refers to the Spearman rank–order correlation coefficient (see Chap. 3) ($prob_d < 1.0e-4$)

For composite Pareto-optimal springs a correlation exists among stiffness, local stability and resonance frequency. The a posteriori proof for this (in the sense indicated above) can be obtained by considering Eqs. (12.53), (12.55), (12.72). An extra correlation exists among global stability, mass and resonance frequency for carbon–epoxy springs only. This is possibly due to fact that the strength of carbon–epoxy is stronger than that of glass–epoxy, stability limits are reached more easily.

The other correlations listed in Table 12.5 are not commented here. Formulae in Sects. 12.1 and 12.3 may provide an easy interpretation of the presented results.

We now examine the computed non-linear correlations (see Chap. 3) between Pareto-optimal spring design variables and Pareto-optimal spring objective functions (Table 12.6). It should be considered that, according to the definitions of both the Pareto-optimal set (see Sect. 2.10) and the Spearman rank–order coefficient (see Chap. 3), the above-mentioned correlations have not to be seen as the result of a parameter sensitivity analysis.

Table 12.6. Significative correlations ($r_s > 0.80$) between objective functions and design variables pertaining to Pareto-optimal springs made from different materials

	Design variable	Objective functions	r_s
Carb.–epoxy	Tubular element diameter ($2\,r_m$)	Global stability (P_c)	0.99
	Tubular element diameter ($2\,r_m$)	Stiffness (k)	0.98
	Tubular element diameter ($2\,r_m$)	Mass (m)	0.98
	Tubular element thickness (h)	Global stability (P_c)	0.98
	Tubular element thickness (h)	Stiffness (k)	0.97
	Tubular element thickness (h)	Mass (m)	0.97
	Tubular element diameter ($2\,r_m$)	Res. frequency ($\omega_{1n_r}, \omega_{2n_r}$)	0.95
	Number of laminae (n_l)	Res. frequency ($\omega_{1n_r}, \omega_{2n_r}$)	0.93
	Tubular element diameter ($2\,r_m$)	Loc. stability ($2\pi r_m N_{xy,cr}$)	0.93
	Number of laminae (n_l)	Global stability (P_c)	0.92
	Tubular element thickness (h)	Res. frequency ($\omega_{1n_r}, \omega_{2n_r}$)	0.91
	Number of laminae (n_l)	Mass (m)	0.91
	Number of laminae (n_l)	Stiffness (k)	0.91
	Number of laminae (n_l)	Loc. stability ($2\pi r_m N_{xy,cr}$)	0.90
Glass-epoxy	Tubular element diameter ($2\,r_m$)	Global stability (P_c)	0.97
	Tubular element diameter ($2\,r_m$)	Stiffness (k)	0.96
	Tubular element diameter ($2\,r_m$)	Mass (m)	0.96
	Tubular element thickness (h)	Stiffness (k)	0.93
	Tubular element thickness (h)	Mass (m)	0.93
	Tubular element thickness (h)	Global stability (P_c)	0.92
	Number of laminae (n_l)	Mass (m)	0.91
	Number of laminae (n_l)	Stiffness (k)	0.91
	Number of laminae (n_l)	Loc. stability ($2\pi r_m N_{xy,cr}$)	0.91
	Tubular element thickness (h)	Loc. stability ($2\pi r_m N_{xy,cr}$)	0.91
	Number of laminae (n_l)	Global stability (P_c)	0.89
Steel	Tubular element thickness (h)	Global stability (P_c)	0.91
	Tubular element thickness (h)	Loc. stability ($2\pi r_m N_{xy,cr}$)	0.90
	Helix angle (α)	Limit shear stress (τ_l)	0.89
	Tubular element thickness (h)	Stiffness (k)	0.83
	Tubular element thickness (h)	Mass (m)	0.83
	Tubular element diameter ($2\,r_m$)	Stiffness (k)	0.80
	Tubular element diameter ($2\,r_m$)	Mass (m)	0.80

r_s refers to the Spearman rank–order correlation coefficient (see Chap. 3) ($prob_d < 1.0^{-4}$).

In parameter sensitivity analysis only one parameter is varied while the other ones are kept fixed. Here every parameter is varying provided that the variation is contained into the Pareto-optimal set, i.e. provided that the minimum mass and the maximum stiffness are attained (together with the constraints in Sect. 12.4.3).

For the two kinds of the considered Pareto-optimal springs the tubular element diameter ($2\,r_m$), the tubular element thickness (h) and the number of laminae (n_l) are related with stiffness Eq. (12.53), mass Eq. (12.77) and global stability Eq. (12.55). The tubular element diameter is related to resonance frequency (12.72).

Other occurrences referenced in Table 12.6 and not commented above are either obvious (e.g. the correlation between the tubular element thickness and the local stability for glass–epoxy springs) or of rather difficult interpretation (e.g. the correlation between the tubular element diameter and the local stability for carbon–epoxy springs). One has to remember that eight design variables were changed simultaneously. Many low relationships have been discovered between design variables (for r_s analysis see Chap. 3). This means that no direct relationships exist between many design variables, i.e. a Pareto-optimal design variable is related to more than one single design variable. Some Pareto-optimal design variables are thus related in a complex manner and a specific study (either theoretical or experimental) would clarify the occurrence.

12.5 Conclusions

A contribution to the optimal design of composite material tubular compression helical springs has been given. Both theoretical studies and experimental activities were undertaken. From the theoretical point of view, both a fundamental study and a numerical application have been presented. The fundamental study has established (given the stiffness, the maximum deflection and the spring material properties) that the mass of tubular (thin-walled) metal springs can be, as a theoretical limit, at least *one half* of the mass of common bar metal springs. Given the stiffness, the maximum deflection and the spring material mechanical properties, tubular springs have the helical diameter and the tubular element outer diameter slightly greater than that for common bar springs. The mass reduction that can be obtained by employing composite material tubular springs (with respect to steel tubular ones) could be estimated too, and seems to be one order of magnitude approximately. Additionally, in the fundamental study, a number of important relationships among spring parameters have been highlighted. Local and global stability of the spring were not dealt with in the fundamental study. To overcome this limitation, a more refined, and inherently more specific, theoretical study has been produced on the basis of multi-objective programming. By this design method, it has been possible to obtain significant results referring to the optimal design of composite material tubular helical compression springs. Eight design variables related to the geometry of carbon–epoxy, glass–epoxy and steel springs were varied in order to obtain both the minimum mass and the

maximum stiffness, subject to six different constraints on local and global stability, on whole spring body vibration, on deflection, on strength.

The specific design problem that has been proposed and solved, although rather complex, is still relatively simple (due to the single loading force acting axially, the mathematical models used, etc.) but allowed to shed some light on the very complicated task of designing an optimal composite material helical spring. The mathematical models used to derive the optimal solutions were expressed in analytical form, for ease of interpretation of results. In particular, the expression of the critical load which causes local instability has been derived in symbolical form. The mathematical models were validated by comparing the experimental and the computed data referring to eight glass–epoxy springs, expressly built for the scope.

The results of the Optimisation (numerical application) show that the mass of helical compression springs can dramatically be reduced (by a factor ranging from 5 to 20) by adopting composite materials, with respect to the employment of steel springs.

Referring to the so-called Pareto solutions, optimal springs (minimum mass springs) have generally a thin-walled tubular section. The helix angle is higher for the composite springs with respect to the tubular steel ones, it has been shown that this depends mostly on the shear modulus values. All the optimal composite springs have several laminae (4–6). The fibre angle ranges from 25° to 35° in order to cope with the requirements on both torsion and bending strength.

The proposed optimal design method could be used to solve more complicated cases involving the design of structures made from composite material.

Future research could focus on comprehensive experimental assessment (ageing, creep and fatigue strength) of optimal springs selected from the Pareto set. The actual common adoption of tubular helical springs, either made from steel or composite material, depends on proper technologies to be developed for both guaranteeing the quality and reducing considerably the expected production cost of these components.

12.6 Appendix: Analytical Expression of Critical Load

The expression of $N_{xy,cr}$ has been derived symbolically and is given as a function of n, m, a_{kj}, b_{kj}, d_{kj}

$$N_{xy,cr} = -(-(b_{22}\,L^3\,n^3 + 2\,r\,(-2\,a_{22}\,L^3\,n + m\,\pi\,(-3\,b_{26}\,L^2\,n^2 + 2\,r\,(2\,a_{26}\,L^2 + m\,\pi\,(b_{12}\,L\,n + 2\,b_{66}\,L\,n - 2\,b_{16}\,m\,\pi\,r))))))\,(-2\,L\,r\,(a_{26}\,L^2\,n^2 + 2\,m\,\pi\,r\,(-a_{12}\,L\,n - a_{66}\,L\,n + 2\,a_{16}\,m\,\pi\,r))\,(-b_{26}\,L^3\,n^3 + 2\,r\,(2\,a_{26}\,L^3\,n + m\,\pi\,(b_{12}\,L^2\,n^2 + 2\,b_{66}\,L^2\,n^2 + 2\,r\,(-2\,a_{12}\,L^2 + m\,\pi\,(-3\,b_{16}\,L\,n + 2\,b_{11}\,m\,\pi\,r))))) - 2\,L\,r\,(a_{66}\,L^2\,n^2 + 4\,m\,\pi\,r\,(-a_{16}\,L\,n + a_{11}\,m\,\pi\,r))\,(b_{22}\,L^3\,n^3 + 2\,r\,(-2\,a_{22}\,L^3\,n + m\,\pi\,(-3\,b_{26}\,L^2\,n^2 + 2\,r\,(2\,a_{26}\,L^2 + m\,\pi\,(b_{12}\,L\,n + 2\,b_{66}\,L\,n - 2\,b_{16}\,m\,\pi\,r)))))) + 2\,L\,r\,(b_{26}\,L^3\,n^3 - 2\,r\,(2\,a_{26}\,L^3\,n + m\,\pi\,(b_{12}\,L^2\,n^2 + 2\,b_{66}\,L^2\,n^2 + 2\,r\,(-2\,a_{12}\,L^2 + m\,\pi\,(-3\,b_{16}\,L\,n + 2\,b_{11}\,m\,\pi\,r)))))\,((a_{22}\,L^2\,n^2 + 4\,m\,\pi\,r\,(-a_{26}\,L\,n + a_{66}\,m\,\pi\,r))\,(b_{26}\,L^3\,n^3 - 2\,r\,(2\,a_{26}\,L^3\,n + m\,\pi\,(b_{12}\,L^2\,n^2 + 2\,b_{66}\,L^2\,n^2 + 2\,r\,(-2\,a_{12}\,L^2 + m\,\pi\,(-3\,b_{16}\,L\,n + 2\,b_{11}\,m\,\pi\,r))))) - (a_{26}\,L^2\,n^2 + 2\,m\,\pi\,r\,(-a_{12}\,L\,n - a_{66}\,L\,n + 2\,a_{16}\,m\,\pi\,r))\,(b_{22}\,L^3\,n^3 + 2\,r\,(-2\,a_{22}\,L^3\,n + m\,\pi\,(-3\,b_{26}\,L^2\,n^2 + 2\,r\,(2\,a_{26}\,L^2 + m\,\pi\,(b_{12}\,L\,n + 2\,b_{66}\,L\,n - 2\,b_{16}\,m\,\pi\,r))))) + 2\,L\,r\,(a_{26}^2\,L^4\,n^4 - 4\,a_{26}\,L\,m\,n\,\pi\,r\,(a_{12}\,L^2\,n^2 + 2\,m\,\pi\,r\,(a_{16}\,L\,n - 2\,a_{11}\,m\,\pi\,r)) - a_{22}\,L^2\,n^2\,(a_{66}\,L^2\,n^2 + 4\,m\,\pi\,r\,(-a_{16}\,L\,n + a_{11}\,m\,\pi\,r)) + 4\,m^2\,\pi^2\,r^2\,(a_{12}^2\,L^2\,n^2 + 4\,(a_{16}^2 - a_{11}\,a_{66})\,m^2\,\pi^2\,r^2 + 2\,a_{12}\,L\,n\,(a_{66}\,L\,n - 2\,a_{16}\,m\,\pi\,r)))\,(d_{22}\,L^4\,n^4 + 8\,r\,(-b_{22}\,L^4\,n^2 - d_{26}\,L^3\,m\,n^3\,\pi + r\,(2\,a_{22}\,L^4 + m\,\pi\,(4\,b_{26}\,L^3\,n + m\,\pi\,(d_{12}\,L^2\,n^2 + 2\,(d_{66}\,L^2\,n^2 + r\,(-2\,b_{12}\,L^2 + m\,\pi\,(-2\,d_{16}\,L\,n + d_{11}\,m\,\pi\,r))))))))) / (32\,L^4\,m\,n\,\pi\,r^4\,(a_{26}^2\,L^4\,n^4 - 4\,a_{26}\,L\,m\,n\,\pi\,r\,(a_{12}\,L^2\,n^2 + 2\,m\,\pi\,r\,(a_{16}\,L\,n - 2\,a_{11}\,m\,\pi\,r)) - a_{22}\,L^2\,n^2\,(a_{66}\,L^2\,n^2 + 4\,m\,\pi\,r\,(-a_{16}\,L\,n + a_{11}\,m\,\pi\,r)) + 4\,m^2\,\pi^2\,r^2\,(a_{12}^2\,L^2\,n^2 + 4\,(a_{16}^2 - a_{11}\,a_{66})\,m^2\,\pi^2\,r^2 + 2\,a_{12}\,L\,n\,(a_{66}\,L\,n - 2\,a_{16}\,m\,\pi\,r)))) \tag{12.78}$$

13

Interactive Optimisation of a Flywheel

The first aim of this chapter is to present how multi-objective optimisation (MOO) (see Chap. 3) and global approximation (GA) (see Chap. 4) can be employed in order to speed up and improve the design of complex mechanical system. The second aim of this chapter is to propose a multi-objective interactive design methodology (see Sect. 3.6) based on Pareto surface sensitivity analysis. This methodology has been implemented by a software where the designer interacts with the multi-objective programming software to choose the preferred final solution especially suited for applications where computer simulations can reliably predict the properties of a system.

Many real-world engineering design problems involve the simultaneous optimisation of several conflicting objectives. Design engineers are often interested in identifying the Pareto-optimal set [166] when exploring a design space.

By considering Definition (2.7) of non-dominated solution the outcome of a MOO is not one optimal point but a set of Pareto-optimal solutions that represent the trade-off between objectives. A solution in a Pareto-optimal set cannot be considered better with respect to others in the set without including preference information to rank competing attributes. Interactive optimisation tasks (see Sect. 3.6) can help the designer (i.e. the decision maker) to define this ranking function.

This chapter develops a Pareto-optimisation method for use in real-world optimisation problems. This method is based on coupling design performances, expressed by objective functions and obtained by simulating the process to optimise, with approximation concepts [88,104,159,202] (see Chap. 4). Often the numerical analysis to obtain the objective function considered is computationally expensive, especially for large complex systems. Approximation concepts may help to limit the required analyses during the optimisation. The basic idea is to build approximation of the response-based objective functions and constraints that can be easily evaluated by the optimiser, without resorting to numerical analysis.

The approximating model may be an interface that may also open new ways to make the design optimisation of engineering system more controllable and accessible compared with a direct coupling of optimisation and analysis.

Another important remark that should be stressed is that the method presented gives a strong importance to the sampling of design variables to generate good approximation models. The techniques presented are based on Quasi-Monte Carlo sampling methods [183,246] (see Sect. 3.4.2). The sampling plain of design variables is a somewhat well-distributed plain to maximise the amount of information that can be obtained from a single numerical analysis.

This chapter shows how the Pareto-optimal set can be locally analysed through Pareto sensitivity analysis to eventually restrict the field of search of the final solution by the decision maker. This information is used by an interactive method [166] that allows the designer to compare the first solution with an alternative solution obtained by moving locally around the first solution. The final solution results from an interaction between the designer and the optimisation software. During this process the designer learns the possible performances of his system and can formulate in a satisfactory manner the target of his project.

This chapter will be dedicated to explain the interactive optimisation method and an illustrative example, based on the optimisation of a flywheel, will be shown.

13.1 System Model

The example illustrates the use of the proposed interactive optimisation strategies (Sect. 3.6). It deals with the design of a flywheel for enhanced dynamic, cost and structural performance.

The general aim of the optimal design of this mechanical component is to reduce, at a given rotational speed, the mass, the maximum stress and to increase the energy stored by the flywheel. This problem has been studied extensively in design literature [18, 84, 121, 131]. In this example the flywheel profile is defined by a spline curve. Complete structural analyses of the flywheel should be comprehensive both for the analysis of the shaft and the analysis of the flywheel. For reasons of simplicity the analyses are confined to wheel. The computation of the stresses of the structure is performed by considering the axis-symmetric properties of the structure. The variable flywheel thickness $h(r)$ is considered small, so the stresses σ_z, τ_{zr} and $\tau_{z\theta}$ can be considered as null (see Fig. 13.1).

The forces acting on an infinitesimal volume of the structure are (see Fig. 13.1) as follows:

$$F_r = \sigma_r h r d\theta \tag{13.1}$$

$$F_\theta = \sigma_\theta (h + dh/2) dr \tag{13.2}$$

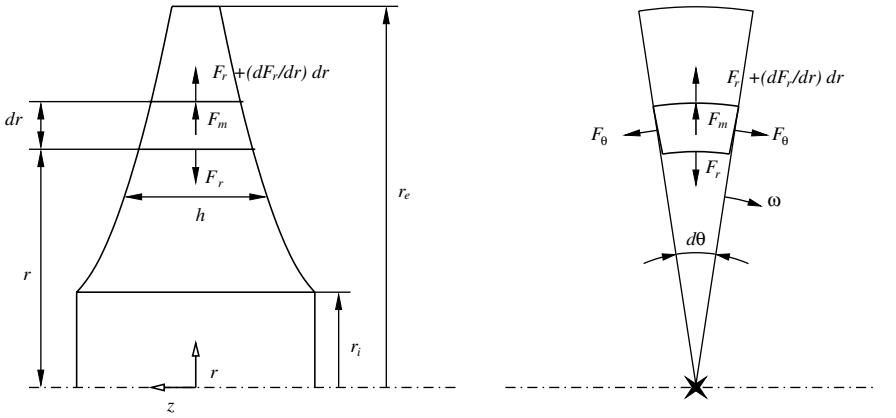


Fig. 13.1. Forces acting on a infinitesimal element of a flywheel having variable thickness

$$F_m = \rho\omega^2 r(h + dh/2)dr(r + dr/2)d\theta \tag{13.3}$$

The equilibrium along the radial direction gives

$$-F_r + F_r + \frac{dF_r}{dr}dr - 2F_\theta \sin\frac{d\theta}{2} + F_m = 0 \tag{13.4}$$

by substituting Eqs. (13.1)–(13.3) and neglecting the second-order terms

$$\frac{d}{dr}(\sigma_r hr) - \sigma_\theta h + \rho\omega^2 r^2 h = 0 \tag{13.5}$$

that are identically satisfied by introducing a stress function [249]:

$$\begin{aligned} \sigma_r &= \frac{\Gamma}{hr} \\ \sigma_\theta &= \frac{1}{h}\frac{d\Gamma}{dr} + \rho\omega^2 r^2 \end{aligned} \tag{13.6}$$

By considering the relations between strain and displacements we have

$$\begin{aligned} \varepsilon_r &= \frac{du}{dr} \\ \varepsilon_\theta &= \frac{u}{r} \end{aligned}$$

that, by eliminating u, becomes

$$\varepsilon_r - \frac{d}{dr}(\varepsilon_\theta r) = 0 \tag{13.7}$$

Consider the relationship between stress and strain

$$\varepsilon_r = \frac{1}{E}\sigma_r - \frac{\nu}{E}\sigma_\theta$$
$$\varepsilon_\theta = \frac{1}{E}\sigma_\theta - \frac{\nu}{E}\sigma_r$$

E is the Young's modulus and ν the Poisson's ratio. We can write Eq. (13.7) as a function of σ_r and σ_θ.

$$\sigma_r - \nu\sigma_\theta - \frac{d}{dr}(r(\sigma_\theta - \nu\sigma_r)) = 0 \tag{13.8}$$

We can perform the structural analysis by considering Eq. (13.8) only in the stress function

$$r^2\frac{d^2\Gamma}{dr^2} + r\frac{d\Gamma}{dr} - \Gamma + (3+\nu)\rho\omega^2 hr^3 = \frac{r}{h}\frac{dh}{dr}\left(r\frac{d\Gamma}{dr} - \nu\Gamma\right) \tag{13.9}$$

This equation cannot be solved analytically. This differential equation has been solved by using a fourth-order Runge–Kutta method and imposing the contour conditions at inner and outer radius ($\Gamma(r_i) = 0$, $\Gamma(r_e) = 0$). All integration points (h, dh/dr) necessary for the integration of Eq. (13.9) are obtained by considering h as a spline function of the radius. The failure criterion refers to Von Mises reference stress and is calculated by

$$\sigma_{VM} = \sqrt{((\sigma_r - \sigma_\theta)^2 + \sigma_r^2 + \sigma_\theta^2)/2} \tag{13.10}$$

If the stresses in the disk exceed the admissible value, failure occurs. The mass M is given by

$$M = 2\pi\rho\int_{r_i}^{r_e} hrdr \tag{13.11}$$

The kinetic energy K stored by the flywheel is obtained by the equation

$$K = \pi\rho\omega^2\int_{r_i}^{r_e} hr^3dr \tag{13.12}$$

13.2 Objective Functions

Optimising the flywheel design requires to maximise the kinetic energy and to minimise weight and stress level (near the admissible value). The multi-objective optimisation problem described is realised by transforming the objective function related to the maximum $\sigma_{VM} = \sigma_{VMM}$, the minimum mass M and the maximum kinetic energy K into a vector to be optimised.

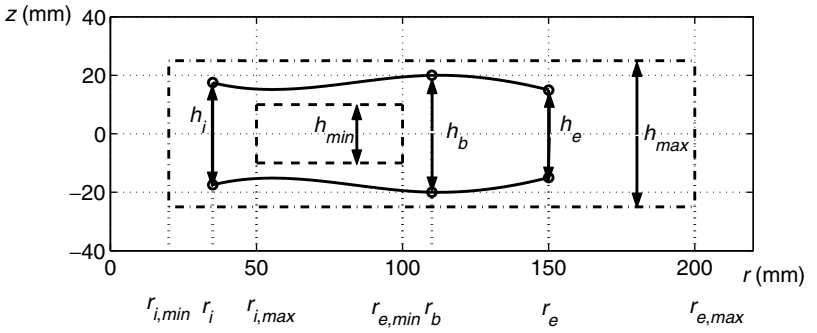


Fig. 13.2. Cross-section of the flywheel. Definition of design variables and design bounds

13.3 Design Variables

The design variables are defined as the coordinates of the control points of a cubic spline (see Fig. 13.2). These control points define the thickness of the flywheel over the entire area of the disk.

So we have the design variables vector (see Fig. 13.2)

$$\begin{aligned}\mathbf{x}^T &= (x_1, \ldots, x_8) \\ &= (r_i, r_e, r_b, h_i, h_e, h_b, (dh/dr)_i, (dh/dr)_e)\end{aligned}$$

by defining the smallest and the largest disk thickness respectively by h_m and h_M we can formulate the multi-objective optimisation problem as

$$\begin{aligned}&\min_{\mathbf{x}} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} = \min_{\mathbf{x}} \begin{bmatrix} M \\ -K \\ \sigma_{VMM} \end{bmatrix} \\ &g_1 = -h_m + h_{min} \leq 0 \\ &g_2 = h_M - h_{max} \leq 0 \\ &g_3 = -r_b + r_i - \Delta r \leq 0 \\ &g_4 = r_b - r_e - \Delta r \leq 0 \\ &g_5 = \sigma_{VMM} - \sigma_{adm} \leq 0 \end{aligned} \tag{13.13}$$

with design variables bounds:

$$\begin{aligned}&0.02\,\mathrm{m} \leq r_i \leq 0.05\,\mathrm{m} \\ &0.1\,\mathrm{m} \leq r_e \leq 0.2\,\mathrm{m} \\ &0.02\,\mathrm{m} \leq r_b \leq 0.2\,\mathrm{m} \\ &0.02\,\mathrm{m} \leq h_i \leq 0.05\,\mathrm{m} \\ &0.02\,\mathrm{m} \leq h_e \leq 0.05\,\mathrm{m}\end{aligned}$$

$$0.02\,\mathrm{m} \leq h_b \leq 0.05\,\mathrm{m}$$
$$-1 \leq \left(\frac{dh}{dr}\right)_i \leq 0$$
$$-1 \leq \left(\frac{dh}{dr}\right)_e \leq 0$$

Problem (13.13) refers to a flywheel that must be constrained to stay in a bounded volume and with maximum stress level equal to σ_{adm}(Table 13.1) The objective function f_2 has been defined equal to $-K$ because the kinetic energy has to be maximised.

Table 13.1. Flywheel problem data

h_{min} (m)	0.02
h_{max} (m)	0.05
Δr (m)	0.025
σ_{adm} (MPa)	280
ν	0.3
ω (rad/s)	1,570
ρ (kg/m^3)	7,800

13.4 Results

The global approximation model was based on a radial basis function neural network (RBFNN) (see Sect. 4.6.2). By using 2,048 points a Quasi-Monte Carlo sequence has been integrated into the interactive Optimisation loop (see Fig. 3.22).

The accuracy of the global approximation model is shown in Figs. 13.3–13.5, where the comparison of the profile of a flywheel computed by using the global approximation model is compared with that of the original physical model.

As much as 10^6 extrapolated points were obtained from the global approximation model to derive a set of non-dominated points close to the Pareto-optimal set. This set has been used in the first step of the design process, to choose a preferential starting solution for the interactive optimisation process through a utility function. In this example, the utility function is defined (see Sect. 3.5.1) by using as target values

- $\tilde{f}_1$ the value of the mass of the flywheel ($\int_V \rho\, dV$) with the minimum mass. ($\mathbf{x} = (50\,\mathrm{mm}, 100\,\mathrm{mm}, 75\,\mathrm{mm}, 20\,\mathrm{mm}, 20\,\mathrm{mm}, 20\,\mathrm{mm}, 0, 0)$). $f_1^0 = 3.678$ kg.
- $\tilde{f}_2$ the value of the maximum kinetic energy of the flywheel with maximum kinetic energy ($\frac{1}{2}\int_V \rho r^2 \omega^2\, dV$)not considering boundaries on stresses. $\mathbf{x} = (20\,\mathrm{mm}, 200\,\mathrm{mm}, 110\,\mathrm{mm}, 50\,\mathrm{mm}, 50\,\mathrm{mm}, 50\,\mathrm{mm}, 0, 0)$.

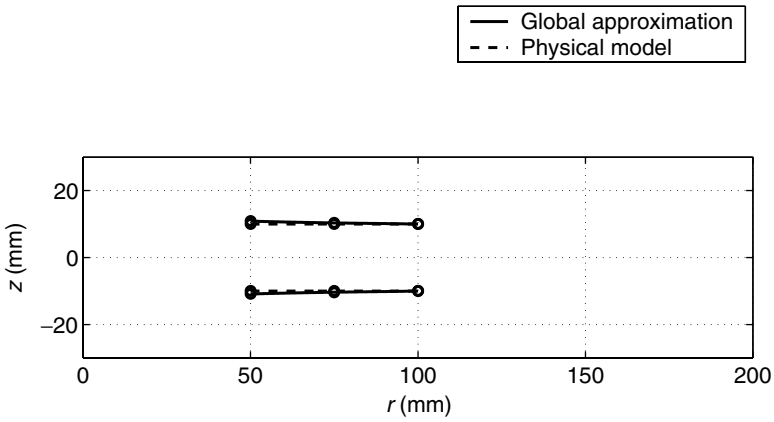


Fig. 13.3. Cross-section of the flywheel with the minimum mass obtained by optimising the global approximation model ('-') and the physical model ('- -')

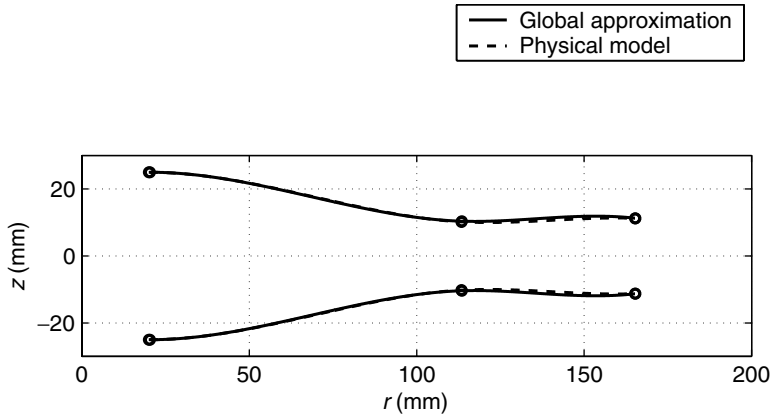


Fig. 13.4. Cross-section of the flywheel with the maximum kinetic energy obtained by optimising the global approximation model ('-') and the physical model ('- -')

$f_2^0 = 120.9 \times 10^{10}$ J.

- f_3 the limit stress of the flywheel considering as flywheel material C40 steel (UNI 7845 maximum admissible stress 420 MPa) and a safety margin of 1.5. So $f_3^0 = 280\,\text{MPa}$.

An optimal flywheel shape has been found by using the global approximation model (Fig. 13.6, the stresses are also plotted).

After that an optimal solution was found, this optimal solution will be taken into account as the reference optimal solution ('R'). The interactive optimisation procedure has been used where the designer explores the Pareto set in the neighbourhood of the current optimal solution. The interactive optimisation framework has been implemented in Matlab [153]. The solutions

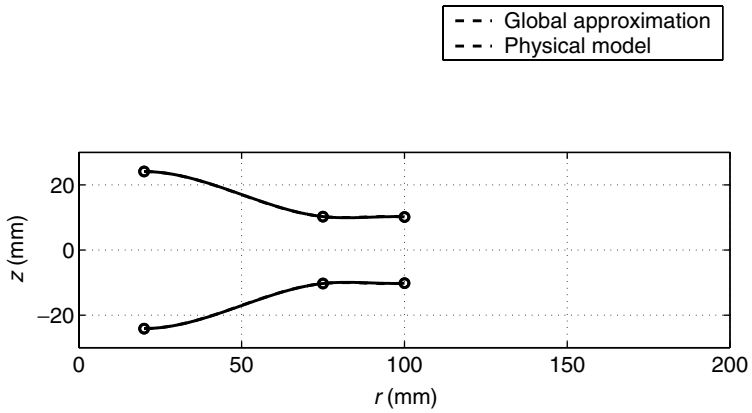


Fig. 13.5. Cross-section of the flywheel with the minimum level of stress (minimum σ_{rM}) obtained by optimising the global approximation model ('-') and the physical model ('- -')

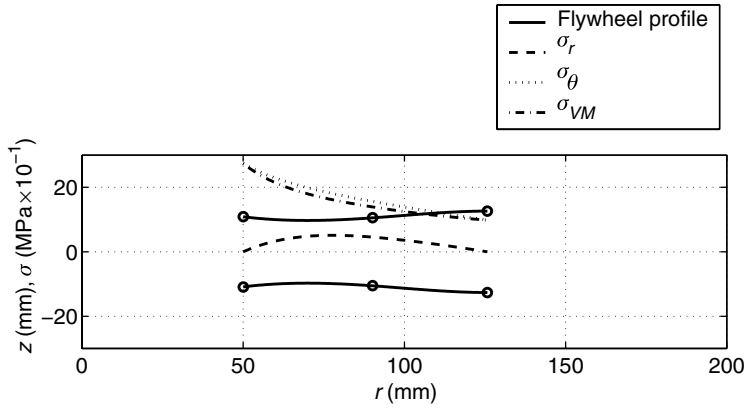


Fig. 13.6. Cross-section and stresses of the flywheel that optimise the utility function (see Sect. 3.5.1) by using the global approximation model. This optimal solution is taken as the reference optimal solution ('R' solution)

shown in Table 13.2 are those obtained by finding Pareto-optimal directions that minimise mass, maximise the stored kinetic energy and minimise the maximum stress.

Table 13.2 shows the results of the Pareto-optimal local sensitivity analysis of the solution shown in Fig. 13.6. The designer can look at this value to have an idea on how the direction in the predictor step (see Fig. 3.22) can influence the original solution.

Table 13.2. Pareto-optimal solutions obtained through utility function sensitivity analysis (remember that $\Delta f_2 > 0$ means a kinetic energy decrease)

	Δf_1 (kg)	Δf_2(J $\times 10^{10}$)	Δf_3 (MPa)
Solution 1	–0.161	0.280	–4.844
Solution 2	0.044	–0.076	0.935
Solution 3	–0.111	0.194	–7.319

The Δf_1, Δf_2, Δf_3 (see 13.13) are variations with respect to the reference optimal solution in Fig. 13.6

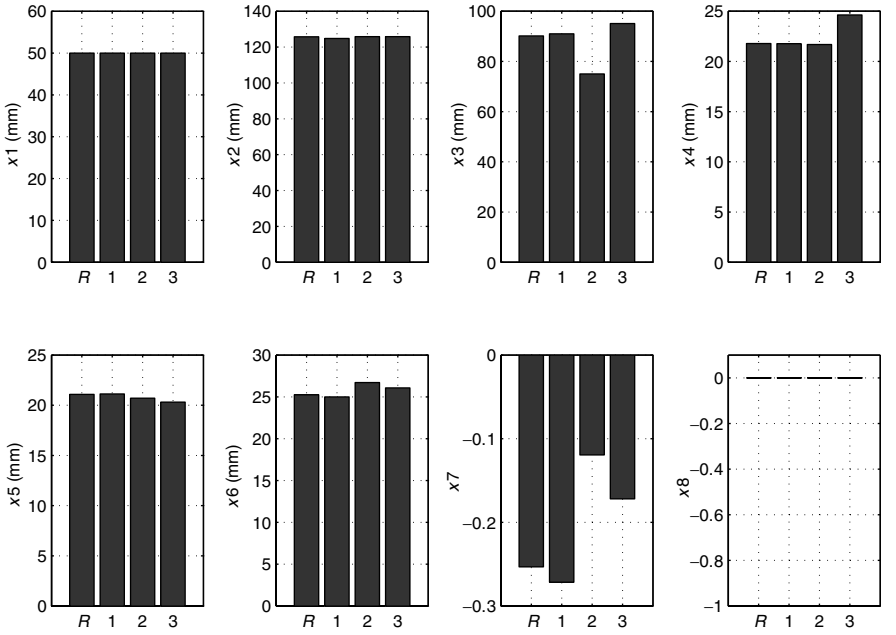


Fig. 13.7. Design variables of the reference solution (R) found during the interactive Pareto-optimal sensitivity analysis (Fig. 13.6). Solutions 1–3 refer to the corresponding solutions 1–3, in Table 13.2

The reference optimal solution (together with the solutions 1–3 in Table 13.2) is finally presented with bar charts (Figs. 13.7 and 13.8). This way of presenting the Pareto-optimal solutions clearly offers a better understanding on the trade-offs.

Following the procedure introduced in Sect. 3.6, the designer can find solutions 1–3 in Table 13.2 in a quick way, i.e. avoiding to complete the computation of the Pareto-optimal solutions by means of the physical system model but exploiting the global approximation model. In Figs. 13.9–13.11 a comparison is presented between Pareto-optimal solutions obtained by using

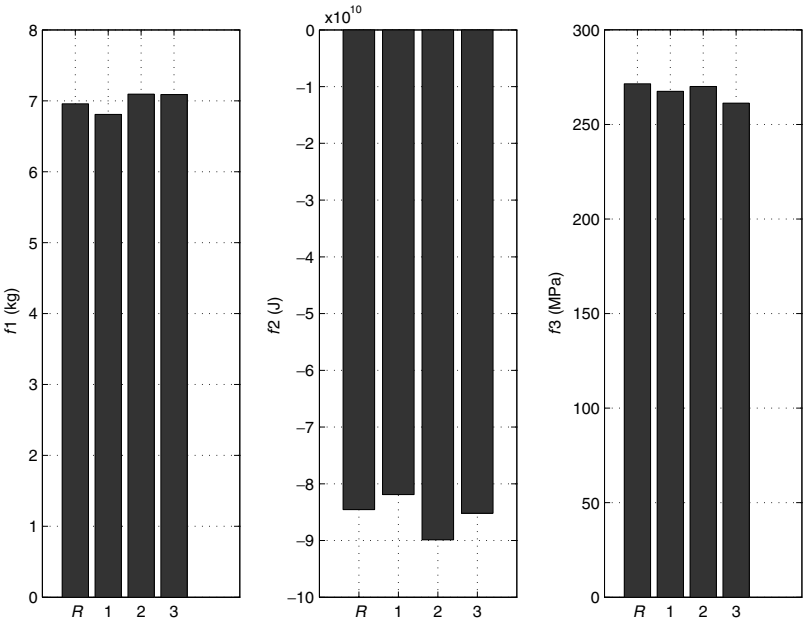


Fig. 13.8. Objective functions of the reference solution (R found during the interactive Pareto-optimal sensitivity analysis (Fig. 13.6). Solutions 1–3 refer to the corresponding solutions 1–3, in Table 13.2

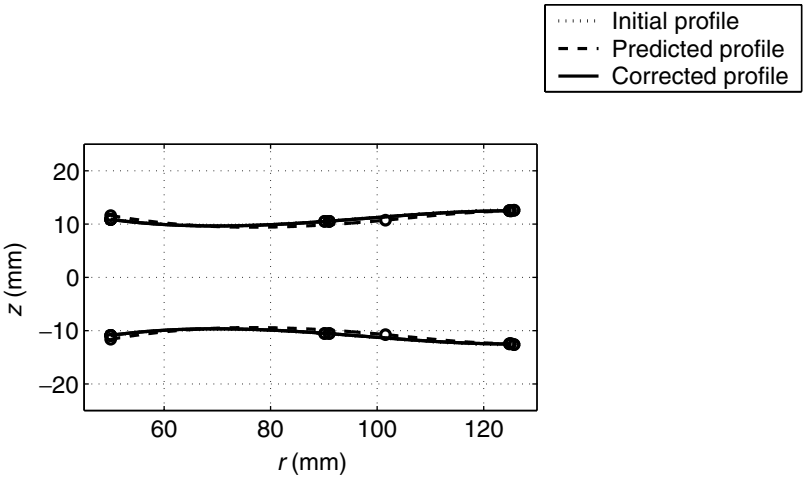


Fig. 13.9. Cross-section of the flywheel of Fig. 13.6 that optimises the utility function (see Sect. 3.5.1) by using the global approximation model compared to the solution obtained by using the prediction solution expressed by Eq. (3.58) choosing the direction that minimises more the flywheel mass and to the corrected solution obtained by solving the problem (3.59)

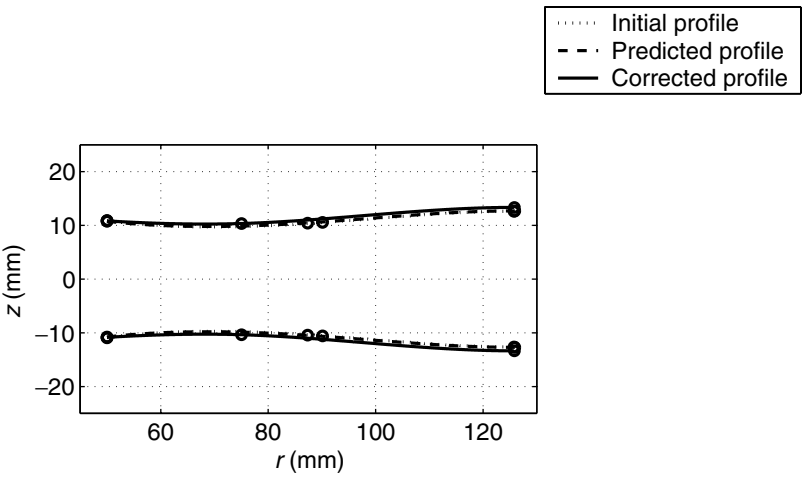


Fig. 13.10. Cross-section of the flywheel of Fig. 13.6 that optimises the utility function (see Sect. 3.5.1) by using the global approximation model compared to the solution obtained by using the prediction solution expressed by Eq. (3.58) choosing the direction that maximises more the flywheel kinetic energy and to the corrected solution obtained by solving the problem (3.59)

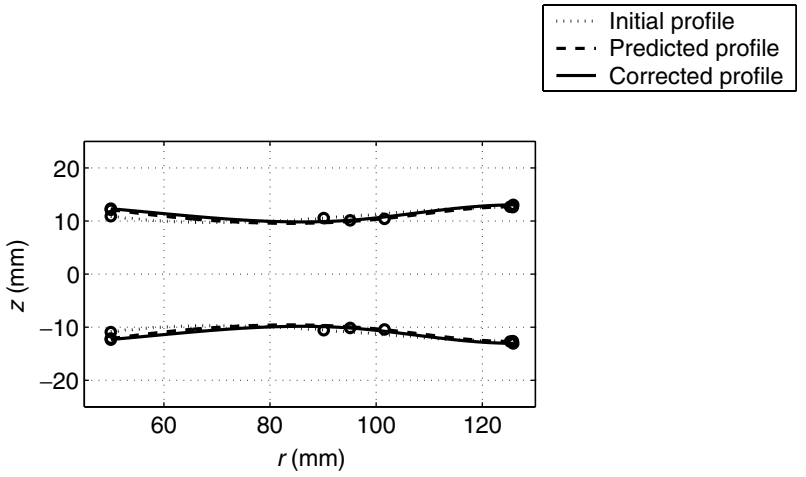


Fig. 13.11. Cross-section of the flywheel of Fig. 13.6 that optimises the utility function (see Sect. 3.5.1) by using the global approximation model compared to the solution obtained by using the prediction solution expressed by Eq. (3.58) choosing the direction that minimises more the flywheel maximum stress and to the corrected solution obtained by solving the problem (3.59)

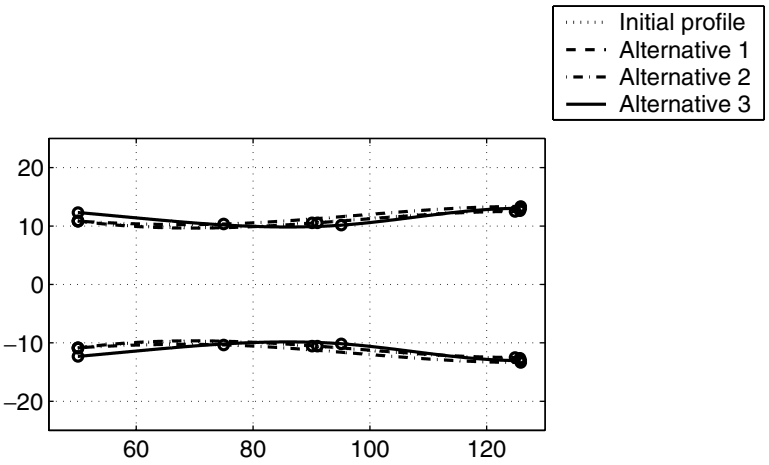


Fig. 13.12. Cross-section of the flywheel of Fig. 13.6 that optimise the utility function (see Sect. 3.5.1) by using the global approximation model compared to the alternative solution of Figs. 13.7 and 13.8.

the prediction framework presented in Sect. 3.6 (and shown in Figs. 13.7 and 13.8, also see Fig. 13.12) and Pareto-optimal solutions obtained by the global approximation model.

The designer can choose one of the solutions and eventually continue the Pareto sensitivity process proposed until a satisfying solution has been obtained.

References

1. VDI 2057:Schwingungseinwirkung auf den menschlichen Körper (1981)
2. N.A. Abdel-Halim et al.: Performance of multicone synchronizers for manual transmission. *Proc. IMechE Part D* (2000)
3. H. Adelman, J. Pitchard, J. Sobieszczanski-Sobieski: Optimization for minimum sensitivity to uncertain parameters. *AIAA J.* **34** (1996)
4. A. Alleyne and J.K. Hedrick: Application of nonlinear control theory to electronically controlled suspensions. *Vehicle Syst. Dyn.* **22** (1993)
5. J. An, A. Owen: Quasi-regression. Technical report, Department of Statistics, Stanford University (Internal report) (1999)
6. V. Van Tooren Antonelli, A. Antoniazzi: Feasibility study on the production of a composite helical spring. In: *Proc. 21st SAMPE Int. Conf.* (2000)
7. N.P. Suh: *Course on Axiomatic Design* (ATA, Firenze, 1999)
8. K. Atkinson: *Elementary Numerical Analysis* (Wiley, New York, 1985)
9. J. Aurell, N. Frojd, S. Nordmark: Correlation between objective handling characteristics and subjective perceptions of handling qualities of heavy vehicles. In: *Proc. AVEC 2000*, Ann Arbor, MI, 2000
10. E. Bakker, H. Pacejka, L. Linder: A new tyre model with an application in vehicle dynamics studies. *SAE Paper 890087* (1989)
11. C. Baldissera et al.: *Metodi di ottimizzazione e programmi di calcolo* (Clup, Milano, 1981)
12. A.R. Barron: Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Trans. Inform. Theor.* **39** (1993)
13. O.A. Bauchau, T.M. Krafchack, J.F. Hayes: Torsional buckling analysis and damage tolerance of graphite/epoxy shafts. *J. Composite Math.* **22** (1998)
14. Z.P. Bazant, L. Cedolin: *Stability of Structures* (Oxford University Press, Oxford, 1991)
15. E. Bazzaro, C. Gorla, S. Miccoli: *Lezioni di Tecnica delle Costruzioni* (Spiegel, Milano, 1997)
16. E. Bender: Optimum linear preview control with application to vehicle suspension. *Trans. ASME, J. Basic Eng.* **90** (1968)
17. J. Benoist: Connectedness of the efficient solutions for vector maximization problems. *J. Optim. Theor. Appl.* **96**(3) (1998)
18. M. Berger, I. Porat: Optimal design of a rotating disk for kinetic energy storage. *ASME Trans.* **55** (1988)

19. L. Breiman, J. Friedman, R.A. Olshen, C.J. Stone: *Classification and Regression Trees* (Wadsworth & Brooks/Cole, Belmont, CA, 1984)
20. D.S. Broomhead, D. Lowe: Multivariable functional interpolation and adaptive splines. *Complex Syst.* **2** (1988)
21. K.A. Bush: Orthogonal arrays of index unity. *Ann. Math. Stat.* **23** (1952)
22. C. Carlsson, R. Fullér: Multiple criteria decision making: the case for interdependence. *Comput. Oper. Res.* **22** (1995)
23. G. Castiglioni et al.: Active vehicle suspension with an active vibration absorber. In: *Proc. AVEC'92 Symp. JSAE*, Japan, 1992
24. J. Castrillon: The train in Spain now goes everywhere: A survey of the performance of the Talgo system. *Global Transp.* (1997)
25. R.M. Chalasani: Ride performance potential of active suspension systems. Technical report, Int. Rep. General Motor Research Laboratories (1983)
26. C.J. Chen et al.: F.E.M. analysis of a helically wound tubular and laminated composite material beam. *Comput. Struct.* **49** (1992)
27. S. Chen, C.F.N. Cowan, P.M. Grant: Orthogonal least squares learning algorithm for radial basis function networks. *IEEE Trans. Neural Netw.* **2**(2) (1991)
28. F.Y. Cheng, D. Li: Genetic algorithm development for multiobjective optimization of structures. *AIAA J.* **36**(6) (1998)
29. S.K. Clark, ed.: *Mechanics of Pneumatic Tyres.* U.S. Department of Transportation. National Highway Traffic Safety Administration (1980)
30. A. Colorni: *Ricerca Operativa.* Città Studi (1991)
31. Complex system. *Science* **284**(5411), 1–212 (April 2, 1999)
32. T.C. Crahan: Modeling steady-state suspension kinematics and vehicle dynamics of road racing cars – Part ii: Examples. *SAE Paper 942506* (1994)
33. D.A. Crolla, D.C. Chen, J.P. Whitehead: Vehicle handling behaviour: subjective v. objective comparisons. In: *Proc. FISITA World Automotive Congress, F98T210*, Paris, 1998
34. D.A. Crolla, D.C. Chen, J.P. Whitehead, C.J. Alstead: Vehicle handling assessment using a combined subjective–objective approach. In: *SAE Paper No. 980226, Proc. SAE World Congress and Exposition*, Detroit, 1998
35. D.A. Crolla, J.P. Whitehead: Reliable subjective assessment of vehicle handling by drivers – is it an elusive goal? *Advanced Vehicle Technologies, DE-vol. 112* (ASME, New York, 2001)
36. G. Cybenko: Approximations by superpositions of a sigmoidal function. *Math. Control, Signals, Syst.* **2** (1989)
37. C.H. Dagli, A. Kusiak: *Intelligent Systems in Design and Manufacturing* (ASME, New York, 1994)
38. I. Das: Optimise large systems via optimal multicriteria component assembly. In: *Proc. 7th AIAA/USAF/NASA/ISSMO Symp.* (AIAA, St. Louis, MO, 1998)
39. L. Davis: *Genetic Algorithms and Simulated Annealing* (Pitman, London, 1990)
40. L. Davis: *The Handbook of Genetic Algorithms* (Van Nostrand Reinhold, New York, 1991)
41. C. Bourcier de Carbon: Théorie mathématique et realisation pratique de la suspension amortié des vehicules terrestres. In: *Proc. of the S.I.A. Conf.*, Paris, 1950
42. A. Dean, D. Voss: *Design and Analysis of Experiments* (Springer, Berlin Heidelberg New York, 1999)

43. K. Deb: An efficient constraint handling method for genetic algorithms. *Comput. Methods Appl. Mech. Eng.* **186**, 311–338 (2000)
44. A. Dey, R. Mukerjee: *Fractional Factorial Plans* (Wiley, New York, 1999)
45. P. Diaconis: *Statistical Decision Theory and Related Topics IV*, vol. 1 (Springer, Berlin Heidelberg New York, 1988)
46. J. Dixon: *Tyres, Suspensions and Handling* (Cambridge University Press, Cambridge, UK, 1991)
47. C.J. Dodds, J.D. Robson: The description of road surface roughness. *J. Sound Vibr.* **2**(31), 175–183 (1973)
48. T. Dompke: S–Bahn–Triebzug Kopenhagen–neue Lösung für den Nahverkehr. *ZEV + DET Glas. Ann.* **118**(5) (1994)
49. C. Doniselli, G. Mastinu: Contributo alla soluzione di alcuni problemi di progettazione di sospensioni dell'autoveicolo (ATA, Nov. 1993)
50. C. Doniselli, G. Mastinu: Traction control for front wheel drive. In: *Proc. XIII IAVSD Symp.*, Chengdu, 1993
51. C. Doniselli, G. Mastinu: New concepts for urban and sub-urban trains. In: *Proc. World Congress Railway Res. 94*, Paris, Nov. 1994
52. C. Doniselli, G. Mastinu: Capitolato tecnico di fornitura per il prototipo del treno innovativo per la Metropolitana di Milano–Linea 1. *Technical report*, Politecnico di Milano, Milan, Italy (1996)
53. L. Donnel: Stability of thin walled tubes under torsion. *Technical report*, NACA Report No. 479 (1933)
54. A. D'Orazio et al.: Multi cone synchronizer dynamic modeling and experimental bench test rig to improve manual transmission shiftability. In: *Proc. MTP2001 Int. Conf.* (JSME, Fukuoka, Japan, 2001)
55. A. Dornig: *Le molle: calcolo e dimensionamento* (CLUP Ed., Milano, 1973)
56. F.Y. Edgeworth: *Mathematical Psychics* (Kegan Paul, London, 1881)
57. M. Ehinger et al.: Nina–Programmsystem zur Untersuchung und Festlegung der Hauptabmessungen von Schienenfahrzeugen. *ZEV + DET Glas. Ann.* **122** (1998)
58. M. Ehrgott, K. Klamroth: Connectedness of the efficient solutions in multiple criteria combinatorial optimization. *Eur. J. Oper. Res.* **97**(1) (1997)
59. H. Eschenauer et al.: *Multicriteria Design Optimisation* (Springer, Berlin Heidelberg New York, 1990)
60. C. Esveld: *Modern Railway Track* (MRT Productions, The Netherlands, 1989)
61. H. Faure: Discrépances des suites associeés à un système de numération (en dimension s). *Acta Arith.* **41** (1982)
62. A. Fiacco, G. McCormick: *Nonlinear Programming: Sequential Unconstrained Minimisation Techniques* (Wiley, New York, 1968)
63. L. Filtri, G. Rovera, L. Ippolito: Functional integration of vehicle systems. In: *Proc. 3rd EAEC Int. Conf. "Vehicle Dyn. Powertrain Eng."*, Strasbourg, 1991
64. P. Fleming, A. Pashkevich: Genetic algorithms for multi-objective optimisation: formulation, discussion and generalisation. In: *Proc. IEEE Control '85 Conf.*, London, 1985
65. R. Fletcher: *Practical Methods of Optimisation* (Wiley, New York, 1980)
66. W. Foag, G. Gruebel: Multi-criteria control design for preview suspension system. In: *Proc. 10th IFAC World Congress on Autom. Control*, Munich, 1987
67. C. Fonseca, P. Fleming: An overview of evolutionary algorithms in multiobjective optimization. *Evol. Comput.* **3**(1) (1995)

68. R.L. Fox: *Optimisation Methods for Engineering Design* (Addison-Wesley, Reading, MA, 1971)
69. F. Frederich: Die Zukunft der Eisenbahn. *ZEV-Glasers Ann.* **110**(5) (1986)
70. F. Frederich: Nulleben–Konzept der Spurführung. *ZEV + DET Glas. Ann.* **123**(7/8) (1999)
71. F. Frederich: Vom Dregestell zum INTEGRAL. *ZEV + DET Glas. Ann.* **123**(1) (1999)
72. J.H. Friedman: Multivariate adaptive regression splines (with discussion). *Ann. Stat.* **19** (1991)
73. R. Fröhling: Low frequency dynamic vehicle–track interaction: modelling and simulation. *Proc. 15th IAVSD Symp. Suppl. Vehicle Syst. Dyn.* **29** (1997)
74. K. Fujita, N. Hirokawa, S. Akagi, T. Hirata: Design optimisation of multi-link suspension system for total vehicle handling and stability. In: *Proc. 7th AIAA/USAF/NASA/ISSMO Symp.* (AIAA, St. Louis, MO, 1998)
75. Y. Furukawa, N. Yuhara, S. Sano, H. Takeda, Y. Matushita: A review of four-wheel steering studies from the viewpoint of vehicle dynamics and control. *Vehicle Syst. Dyn.* **18***, 151–186 (1989)*
76. V.K. Garg, R.V. Dukkipati: *Dynamics of Railway Vehicle Systems* (Academic Press, Toronto, 1984)
77. S. Geman, E. Bienenstock, R. Doursat: Neural networks and the bias-variance dilemma. *Neural Comput.* **4** (1992)
78. G. Genta: *Meccanica dell' autoveicolo* (Levrotto & Bella, Torino, 1989)
79. G. Genta: *Motor Vehicle Dynamics* (World Scientific, Singapore, 1997)
80. J.S. Gero: Design prototypes: a knowledge representation for schema design. *Artif. Intell. Mag.* **11** (1990)
81. P.E. Gill, W. Murray, M.H. Wright: *Practical Optimisation* (Academic Press, London, 1981)
82. T. Gillespie: *Fundamentals of Vehicle Dynamics* (SAE, Warrendale, PA 1992)
83. F. Girosi, M. Jones, T. Poggio: Regularization theory and neural network architectures. *Neural Comput.* **7** (1995)
84. A.A. Giunta, L.T. Watson: A comparison of approximation modelling techniques: polynomial versus interpolating models. In: *Proc. 7th AIAA/USAF/NASA/ISSMO Symp.* (AIAA, St. Louis, MO, 1998)
85. M. Gobbi: *Design of complex mechanical systems, road vehicles related examples.* PhD thesis, Politecnico di Milano (1998)
86. M. Gobbi: Approximation of the pareto-optimal set in multi-objective optimisation. *Technical report*, Politecnico di Milano (2000)
87. M. Gobbi, C. Doniselli, G. Mastinu, L. Guglielmetto, E. Pisino: Optimal & robust design of a road vehicle suspension system. In: *Proc. XVI IAVSD Symp.*, Pretoria, South Africa, 1999
88. M. Gobbi, G. Mastinu: Global approximation: performance comparison of different methods, with application to road vehicle system engineering. In: *Innovat. Vehicle Des. Dev., ASME DE*, vol. 101, pp. 15–24. *ASME Int.* (1999)
89. M. Gobbi, G. Mastinu: On the optimal design of composite material tubular helical springs. *Meccanica* **304** (2001)
90. M. Gobbi, G. Mastinu: Symbolic description and optimisation of the dynamic behaviour of vehicles running on rough road. *J. Sound Vibr.* **254**(3) (2001)
91. M. Gobbi, G. Mastinu: Symbolical multi-objective optimisation of the dynamic behaviour of actively suspended road vehicles. *Int. J. Vehicle Des.* **28**(1–3) (2002)

92. M. Gobbi, G. Mastinu, C. Doniselli: Advances in the optimal design of vehicle subsystems. In: *Proc. 6th EAEC Int. Conf.* (EAEC, Cernobbio (CO), Italy, 1997)
93. M. Gobbi, G. Mastinu, C. Doniselli: Optimising a car chassis. *Vehicle Syst. Dyn.* **32**(2/3) 149–170 (1999)
94. M. Gobbi, G. Mastinu, C. Doniselli, L. Guglielmetto, E. Pisino: Optimal & robust design of a road vehicle suspension system. *Vehicle Syst. Dyn.* **33**, 3–22 (1999)
95. M. Gobbi, G. Mastinu, A. D'Orazio, M. Caudano, G. Faustini: On the optimisation of a double cone synchroniser for improved manual transmission shiftability. *DE – Advanced Vehicle Technologies*, ISBN 0-7918-1692-3 (2003)
96. D.E. Goldberg: *Genetic Algorithms in Search, Optimisation, and Machine Learning* (Addison-Wesley, Reading, MA, 1989)
97. D.E. Goldberg, J. Richardson: Genetic algorithms with sharing for multi-modal function optimisation. In: *Proc. 2nd Int. Conf. Gene. Algorith.* (1987)
98. G.H. Golub, M. Heath, G. Wahba: Generalised cross validation as a method of choosing good ridge parameter. *Technometrics* **21** (1979)
99. S.L. Grassie et al.: The dynamic response of railway track to high frequency vertical/lateral/longitudinal excitation. *J. Mech. Eng. Sci.* **24** (1982)
100. D.E. Grierson, P. Hajela, eds.: *Emergent Computing Methods in Engineering Design – Applications of Genetic Algorithms and Neural Networks* (Springer, Berlin Heidelberg New York, 1996)
101. Y. Guang, A.D. de Pater: The determination of the nonlinear motion of a railway vehicle. In: *Proc. IAVSD Symp.*, Lyon, France, 1991
102. K. Guo, H. Guan: Modelling of the driver/vehicle directional control system. *ASME DE* **40** (1991)
103. K.H. Guo, C.F. Zong, C.F. Kong, M.L. Chen: Objective evaluation correlated with human judgement – an approach to the optimization of vehicle handling control system. *Int. J. Vehicle Des.* (2002)
104. R.T. Haftka, E.P. Scott, J.R. Cruz: Optimization and experiments: A survey. *Appl. Mech. Rev.* **51**(7) (1998)
105. P. Hajela: Stochastic search in discrete structural optimisation. In: *Discrete Structural Optimisation*, International Centre for Mechanical Sciences, ed. by CISM (CISM, Udine, 1996)
106. P. Hajela: Strategies for modelling, approximation, and decomposition in genetic algorithms based multidisciplinary design. *The Oswatitsch Session, Emerging methods for Treating Multidisciplinary Optimisation Problems* (CISM, Udine, 2000)
107. P. Hajela, J. Yoo: Constraint handling in genetic search using expression strategies. *AIAA J.* **34**(12) (1996)
108. A. Hamid, T.L. Yang: Analytical description of track geometry variation. *ENSCO, Transportation Technology Engineering Division* (Springfield, Virginia, 1981)
109. S. Haykin: *Neural Networks. A Comprehensive Foundation*, 2nd edn. (Prentice Hall, Upper Saddle River, NJ, 1999)
110. E. Bakker, H.B. Pacejka: The magic formula tyre model. In: *Proc. 1st Tyre Colloquium*, Delft, Oct. 1991. *Suppl. Vehicle Syst. Dyn.* **21** (1993)
111. J.K. Hedrick: Introduction to active suspensions. *Technical report*, CCG Course on Control of Vehicle Ride and Handling (1990)

112. J. Hendry, C. Probert: Carbon fibre coil springs. *Mater. Des.* **7**(6) (1986)
113. J. Hertz, A. Krogh, R.G. Palmer: *Introduction to the Theory of Neural Computation* (Addison Wesley, Reading, MA, 1991)
114. Y. Hirano, E. Ono: Nonlinear robust control for an integrated system of 4wd and 4ws. In: *Proc. AVEC'94 Symp. JSAE*, Japan, 1994
115. H. Hiroaki: Simulation on synchronization mechanism of transmission gearbox. In: *Proc. 1998 North American ADAMS User Conference*, Ann Arbor, MI, 1998.
116. H. Hiroaki: Analysis on synchronization mechanism of transmission gearbox. *SAE 199-01-0734* **1** (1999)
117. J. Holland: *Adaptation in Natural and Artificial Systems* (University of Michigan Press, Ann Arbor, MI, 1975)
118. K. Hornik, M. Stinchcombe, H. White: Approximation by superpositions of sigmoidal functions. *Math. Control, Signals, Syst.* **2** (1989)
119. C.R. Houck, J.A. Joines, M.G. Kay: A genetic algorithm for function optimization. *Internal report*, North Carolina State University (1997)
120. D. Hrovat: Applications of optimal control to advanced automotive suspension design. *Trans. ASME, J. Dyn. Syst. Meas. Control* **115** (1993)
121. J. Huang, G.M. Fadel: Heterogeneus flywheel modeling and optimization. *Mater. Des.* **21** (2000)
122. Helical compression and extension spring terminology. *SAE J1121* (1999)
123. K.M.A. Kamash, J.D. Robson: The application of isotropy in road surface modelling. *J. Sound Vibr.* **1**(57), 89–100 (1978)
124. D. Karnopp, A. Trikha: Comparative study of optimization techniques for shock and vibration isolation. *Trans. ASME, J. Eng. Ind.*, **91** (1969)
125. W. Kortuem et al.: Progress in integrated system analysis and design software for controlled vehicles. In: *Proc. XIII IAVSD Symp.*, Chengdu, 1993
126. W. Kortuem et al.: Analysis and design of controlled vehicle using multibody simulation models. In: *Proc. AVEC'94 Symp. JSAE*, Japan, 1994
127. B. Kroöse, P. van der Smagt: *An Introduction to Neural Networks* (University of Amsterdam Press, Amsterdam, 1993)
128. M. Kubat: Decision tree can initialize radial-basis function networks. *IEEE Trans. Neural Netw.* **9** (1998)
129. M. Kubat, I. Ivanova: Initialization of RBF networks with decision trees. In: *Proc. 5th Belgian–Dutch Conference on Machine Learning, BENELEARN'95* (1998)
130. R. Lang et al.: Active roll reduction. In: *Proc. EAEC'91 Conf.*, Strasbourg, 1991
131. U. Lautenschlager, H.A. Eschenauer, F. Mistree: Multiobjective flywheel design: a doe-based concept exploration task. In: *Proc. DETC'97* (ASME, Dallas, TX, 1997)
132. D. Liepman, G. Stephanopoulos: Developement and global sensitivity analysis of a closed ecosystem model. *Ecol. Model.* **30** (1985)
133. B.Z. Lin, K.H. Guo: Analysis and optimization of handling performance of a car performing a lane change manoeuvre. *Int. J. Vehicle Des.* (1998)
134. J. Looman: *Zahnradgetriebe. Kostruktionsbücher. Band 26* (Springer, Berlin Heidelberg New York, 1998)
135. K.A. Lou et al.: Buckling of circular composite cylindrical shells under axial compression and bending loads. *J. Composite Mater.* **25** (1991)

136. D. Lowe: On the use of nonlocal and non positive definite basis functions in radial basis function networks. In: *IEEE 4th Int. Conf. on Artif. Neural Netw.*, Perth, Australia, 1995
137. Xiao Pei Lu et al.: A design procedure for optimization of vehicle suspensions. *Int. J. Vehicle Des.* **5** (1984)
138. D. Margolis: All wheel independent torque control. *SAE Paper 881135* (1988)
139. D. Margolis et al.: Integrated torque and steering control for improved vehicle handling. *ASME DE* **40** (1991)
140. G. Mastinu: Escursione della sospensione dell' autoveicolo: derivazione analitica della risposta in presenza di eccitazione stocastica. *Technical report.* Int. Rep. Politecnico di Milano (1988)
141. G. Mastinu: Passive automobile suspension parameter adaptation. In: *Proc. the IMechE Conf. – Adv. Suspens.* (Institution of Mechanical Engineers, London, 1988)
142. G. Mastinu: Non linear models for the study of automobile handling. In: *Proc. EAEC Conf.*, Strasbourg, 1991
143. G. Mastinu: Automotive suspension design by multi-objective programming. In: *Proc. AVEC'94 Symp. JSAE*, Japan, 1994
144. G. Mastinu: Integrated controls and interactive multi-objective programming for the improvement of ride and handling of road vehicles. In: *Smart Vehicles*, ed. by J.P. Pauwelussen, H.B. Pacejka (Lisse, The Netherlands, 1995)
145. G. Mastinu: Note sull' impiego dei materiali compositi per la costruzione di veicoli ferroviari. *Trasporti Trazione* **5** (1995)
146. G. Mastinu: Comfort vibrazionale del veicolo ferroviario: Nozioni di base. *Technical report*, Politecnico di Milano (1996)
147. G. Mastinu: A method to design optimal railway vehicles. In: *Proc. WCRR97 World Congress*, Florence, 1997
148. G. Mastinu, G. Babbel, E. Lugner, P. Margolis, D. Mittemayr, B. Richter: Integrated controls of lateral vehicle dynamics. In: *Proc. XIII IAVSD Symp.*, Chengdu, 1993
149. G. Mastinu, M. Gobbi: Advances in the optimal design of mechanical systems. Course coordinated by CISM (International Centre for Mechanical Sciences), Available at www.europeindia.org (Birla Science Centre, Hyderabad, 1999)
150. G. Mastinu, M. Gobbi: On the optimal design of railway passenger vehicles. *Proc. Inst. Mech. Eng. Part F, J. Rail Rapid Transit* **215** (2001)
151. G. Mastinu, M. Gobbi, G.D. Pace: Analytical formulae for the design of railway vehicle suspension system. *Proc. Inst. Mech. Eng. Part C, J. Mech. Eng. Sci.* **216** (2001)
152. *Optimisation Toolbox User's Guide* (The Mathworks, San Marcos, TX, 1996)
153. *Optimisation Toolbox User's Guide* (The Mathworks, San Marcos, TX, 1999)
154. J.B. Matusov: *Multicriteria Optimisation and Engineering* (Chapman & Hall, New York, 1995)
155. M. McKay, R. Beckman, W. Conover: A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* **21** (1979)
156. G.J. McRae, J.W. Tilden, J.H. Seinfeld: Global sensitivity analysis – a computational implementation of the Fourier amplitude sensitivity test (fast). *Comput. Chem. Eng.* **6** (1982)

157. C. Miano, M. Gobbi, G. Mastinu: Multi-objective optimization of the handling performances of a road vehicle: a fundamental study on tire selection. *ASME J. Mech. Des.* **126**, 687 (2004)
158. C.M. Miano: *Optimisation Methods of Complex Mechanical System with Particular Reference to Ground Vehicles* (in Italian). PhD thesis, Politecnico di Milano (1998)
159. C.M. Miano, M. Gobbi, G. Mastinu: A tutorial on present and future applications of global approximation issues with application to vehicle design problems. *Advanced Vehicle Technologies, DE*, vol. 112 (ASME, New York, 2001)
160. C.M. Miano, M. Gobbi, G. Mastinu: Interactive multiobjective optimisation methods related to the design of complex mechanical system. *Internal Report*, Politecnico di Milano, Milan, 2002.
161. C.M. Miano, M. Gobbi, G. Mastinu, R. Cesarini: On the integrated design of tyre-suspension system of a racing car. *Advanced Vehicle Technologies, DE*, vol. 106 (ASME, Orlando, 2000)
162. Z. Michalewicz: *Genetic Algorithms + Data Structures = Evolution Programs.* (Springer, Berlin Heidelberg New York, 1994)
163. N. Michelena, T. Jaing, P. Papalambros: Decomposition of simultaneous analysis and design models. In: *Structural and Multidisciplinary Optimisation*, ed. by G. Rozvany, N. Olthoff (Elsevier, Amsterdam, 1995)
164. N. Michelena, P. Papalambros: A hypergraph framework for optimal model-based decomposition of design problems. *Comput. Optim. Appl.* **8**(2) (1997)
165. N.F. Michelena, A.M. Agogino: Multiobjective hydraulic cylinder design. *J. Mech. Transmiss. Automat. Des. – Trans. ASME* **110** (1988)
166. K. Miettinen: *Nonlinear Multiobjective Optimisation* (Kluwer, Boston, 1999)
167. M. Mitschke: *Dynamik der Kraftfahrzeuge* (Springer, Berlin Heidelberg New York, 1990)
168. M. Mitschke, E. Ahring: Control loop for driver–vehicle with four wheel steering. In: *Proc. AVEC'94 Symp. JSAE*, Japan, 1994
169. M. Miyamoto, E. Satou: Dynamics of a bogie with independently rotating wheels. In: *Proc. IAVSD Symp.*, Lyon, France, 1991
170. F. Molinaro: *Proposta di un nuovo treno a due piani per alta velocita'*. Master's thesis, Technical University of Milan (1996)
171. A.M. Mood, F.A. Graybill, D.C. Boes: *Introduction to the Theory of Statistics* (McGraw-Hill, New York, 1974)
172. J. Moody: The effective number of parameters: an analysis of generalisation and regularisation of non linear learning systems. *Neural Information Processing Systems 4* (Morgan Kauffmann, San Mateo, CA, 1992)
173. A. Morelli: Progetto dell autoveicolo, concetti di base. *CELID* (1999)
174. M. Morelli, S. Polinari: *Un metodo per la progettazione di veicoli ferroviari passeggeri.* Master's thesis, Politecnico di Milano (1998)
175. J. Mottershead: Finite elements for dynamical analysis of helical rods. *Int. J. Mech. Sci.* **22** (1980)
176. P. Mueller, W.O. Schiehlen: *Linear Vibrations* (Martinus Nijoff, Dordrecht, 1985)
177. S. Murata et al.: Syncronizer and shift system optimization for improved manual transmission shiftability. *SAE Paper 891998* **1** (1989)
178. J.A. Nelder, R. Mead: A simplex method for function minimiztion. *Comput. J.* **7** (1965)

179. R. De Neufville, M. McCord: Unreliable measurement of utility: significant problems for decision analysis. *Operational Research'84* (Elsevier, Amsterdam, 1984)
180. C.G. Newton et al.: *Analytical Design of Linear Feedback Controls*, Appendix E (Wiley, New York, 1957)
181. K. Newton, W. Steeds, T. Garret: *The Motor Vehicle.* (Butterworths, 1989)
182. W.Y. Ng: *Interactive Multi-Objective Programming as a Framework for Computer-Aided Control System Design* (Springer, Berlin Heidelberg New York, 1989)
183. H. Niederreiter: Low-discrepancy and low-dispersion sequences. *J. Number Theor.* **(30)**, 51–70 (1988)
184. H. Niederreiter: *Random Number Generation and Quasi Montecarlo Methods* (SIAM, Philadelphia, PA, 1992)
185. Y. Okazaki, K. Komatsuzaki: Double cone synchronizer with paper lining for medium duty trucks. *SAE Paper 902278* **1** (1990)
186. E. Ono et al.: Coordination of vehicle steering and suspension systems by integrated control strategy. In: *Proc. AVEC'92 Symp. JSAE*, Japan, 1992
187. J.J.M. Van Oosten, E. Bakker: Determination of magic tyre model parameters. In: *Proc. 1st Tyre Colloquium*, Delft, 1991
188. Power spectral density of track irregularities. *ORE C116 RP1* (1971)
189. M. Orr, J. Hallam, A. Murray, T. Leonard: Assessing RBF networks using delve. *Int. J. Neural Syst.* **10** (2000)
190. A. Osycza, S. Kundu: A modified distance method for multicriteria optimization, using genetic algorithms. *Comput. Ind. Eng.* **30**(4) (1996)
191. A. Osyczka: *Multicriterion Optimisation in Engineering* (Ellis Horwood, Chichester, 1984)
192. A. Owen, J. Kohler: *Computer Experiments. Handbook of Statistics*, vol. 13 (Elsevier Science, New York, 1996)
193. Ü Özgüner, K.A. Ünyelioğlu, C. Hatipoğlu: An analytical study of vehicle steering control. In: *Proc. IEEE Conf. Control Appl* (IEEE, Albany, NY, 1995)
194. H.B. Pacejka: *Tyre and Vehicle Dynamics* (Elsevier, Amsterdam, 2006)
195. H.B. Pacejka, E. Bakker: The magic formula tyre model. In: *Proc. 1st Tyre Colloquium*, Delft, 1991
196. H.B. Pacejka, T. Takahashi: Pure slip characteristics of tyres on flat and on undulated road surfaces. In: *AVEC 1992* (1992)
197. G. Pahl, W. Beitz, *Engineering Design: A Systematic Approach*, 2nd edn. (Springer, London, 1996)
198. R. Panagin: *La dinamica del veicolo ferroviario* (Levrotto & Bella, Torino, 1990)
199. R. Panagin: Il fattore di accrescimento nei rotabili tradizionali e innovativi veloci. *Ingegneria Ferroviaria* **4** (1996)
200. P.Y. Papalambros, D.J. Wilde: *Principles of Optimal Design* (Cambridge University Press, Cambridge, UK, 2000)
201. V. Pareto: *Manuale di economia politica* (Societá Editrice Libraria, Milano, 1906)
202. P. Hajela: Neural networks in discrete structural optimisation. *Technical report*, Rensselaer Polytechnic Institute, Troy, New York (1996)
203. D.T. Pham, Y. Yang: Optimization of multi-modal discrete functions using genetic algorithms. *Proc. Inst. Mech. Eng.* **207** (1993)

204. L. Della Pietra: The dynamic coupling of torsional and flexural strains in cylindrical helical springs. *Meccanica* (1975)
205. O. Polách et al.: Gekoppelte Einzelachsfahrwerke FEBA mir Radialeinstellung Erfolgreich erprobt. *Eisenbahn Rev.* **10**, 424–429 (1999)
206. E. Polak: *Optimisation, Algorithms and Consistent Approximations* (Springer, Berlin Heidelberg New York, 1997)
207. E. Ponslet, W. Miller: Coil springs with constrained layers visco-elastic damping for passive isolation. *Proc. SPIE* **3327** (1998)
208. M.G. Pottinger: The Flat-Trac II ® machine, the state-of-the-art in tire force and moment measurement. *Tire Sci. Technol.* (1992)
209. P.J. Powell: A fast algorithm for nonlinearly constrained optimization calculations. *Num. Anal.* **630** (1978)
210. W. Press et al.: *Numerical Recipes* (Cambridge University Press, Cambridge, UK, 1992)
211. I.S. Raju, J.T. Wang: Classical laminate theory models for woven fabric composites. *J. Comp. Technol. Res.* **16**(4) (1994)
212. J.O. Rawlings: *Applied Regression Analysis* (Wadsworth & Brooks/Cole, Pacific Grove, CA, 1988)
213. G. Reusing et al.: Low cost active suspension systems. In: *Proc. AVEC'92 Symposium JSAE*, Japan, 1992
214. B. Richter et al., Evolutionsstrategie – ein Hilfsmittei bei der Löesung fahrzeuge technischer Aufgabe, *ATZ* **84**, 331–337 (1982)
215. J.P. Riff: L' evolution des materiels roulants metro. *Revue Generale des Chemins de Fer* (Oct. 1993)
216. G. Rudolph: On a multi-objective evolutionary algorithm and its convergence to the pareto set. *IEEE* (1998)
217. D. Mc Ruer, R. Allen, H. David: New results in driver steering control models. *Hum. Factors* (1977)
218. J. Sacks, W.J. Welch, T.J. Mitchell, H.P. Wynn: Design and analysis of computer experiments. *Stat. Sci.* **4** (1989)
219. A. Saltelli, T.H. Andres, T. Homma: Sensitivity analysis of model output: an investigation of new techniques. *Comput. Stat. Data Anal.* **15** (1993)
220. A. Saltelli, T. Homma: Importance measures in global sensitivity analysis of nonlinear models. *Reliabil. Eng. Syst. Saf.* **52** (1996)
221. S. Sato et al.: Integrated chassis control system for improved vehicle dynamics. In: *Proc. AVEC'92 Symp. JSAE*, Japan, 1992
222. J.D. Schäffer: *Some Experiments in Machine Learning Using Vector Evaluated Genetic Algorithms*. PhD thesis, Vanderbilt University, Nashville (1984)
223. W.E. Schmitendorf, G. Moriarty: A simple derivation of necessary conditions for pareto-optimality. *IEEE Trans. Autom. Control* (1995)
224. J. Schuller, I. Haque, G. Fadel: Methods for simulation-based optimisation of vehicle handling behavior. *Advanced Vehicle Technologies, DE*, vol. 106 (ASME, Orlando, 2000)
225. D.J. Schulring, W. Pelz, M.G. Pottinger: A model for combined tire cornering and braking forces. *SAE Paper 960180* (1996)
226. D.E. Sen, J.B. Yang: *Multiple Criteria Support in Engineering Design* (Springer, Berlin Heidelberg New York, 1998)
227. R. Sharp, D. Crolla: Road vehicle suspension system design – a review. *Vehicle Syst. Dyn.* **16**(3) (1987)

228. Y. Shibahata et al.: The improvement of vehcile monoeuvrability by direct yaw moment control. In: *Proc AVEC'892 Symp. JSAE*, Japan, 1992
229. W. Shyy, N. Papila, R. Vaidyanathan, K. Tucker: Global design optimization for aerodynamics and rocket propulsion components. *Prog. Aerosp. Sci.* **37** (2001)
230. T.W. Simpson, J. Peplinski, P.N. Koch, J.K. Allen: On the use of statistics in design and implications for deterministic computer experiments. In: *Proc. DTM'97*, no. DECT97/DTM-3881 (ASME, Dellas, TX, 1997)
231. C.C. Smith, D.Y. Mc Gehee, A.J. Healey: The prediction of passenger riding comfort from acceleration data. *Trans. ASME, J. Dyn. Syst. Meas., Control* **100** (1978)
232. I.M. Sobol: The distribution of points in a cube and the approximate evaluation of integrals. *USSR Comput. Math. Math. Phys.* **7**, 86–112 (1967)
233. I.M. Sobol: Sensitivity analysis for nonlinear mathematical models. *Math. Model. Comput. Exp.* **1** (1993)
234. R.J. Socin, K.L. Walters: Manual transmission syncronizers. *SAE Paper 680008* **1** (1968)
235. R. Sommerer: INTEGRAL – ein Gliederzugsystem für den Vollbahnbetrieb. *ZEV + DET Glas. Ann.* **123**(2) (Feb. 1999)
236. N. Srinivas, K. Deb: Multiobjective optimization using nondominated sorting in genetic algorithms. *Evol. Comput.* **2**(3) (1994)
237. W. Stadler: Multicriteria optimization in mechanics. *Appl. Mech. Rev.* **37**, 277–286 (1984)
238. W. Stadler: *Multicriteria Optimisation in Engineering and in the Sciences* (Plenum, New York, 1988)
239. W. Stadler: Caveats and boons of multicriteria optimization. *Microcomput. Civil Eng.* **10** (1995)
240. G. Sun, J. Hansen: Optimal design of laminated–composite circular–cylindrical shells subject to combined loads. *Trans. ASME, J. Appl. Mech.* **55** (1988)
241. M. Sun, A. Stam, R.E. Steuer: Interactive multiple objective programming using tchebycheff programs and artificial neural networks. *Comput. Oper. Res.* **27** (2000)
242. A. Tabiei, G. Simitses: Buckling of moderately thick, laminated cylindrical shells under torsion. *AIAA J.* **32**(3) (1994)
243. D. Tan: Torsional buckling analysis of thin and thick shells of revolution. *Int. J. Solids Struct.* **37** (2000)
244. B. Tang: Orthogonal-based Latin hypercubes. *J. Am. Stat. Assoc.* **88** (1993)
245. `UNI ISO 7401`: Veicoli stradali. procedura di prova a regime transitorio con accelerazione laterale, Gennaio (1992)
246. S. Tezuka: *Uniform Random Numbers: Theory and Practice* (Kluwer, Dordretch, 1995)
247. A.G. Thompson: Suspension design for optimum road-holding. *SAE Paper 830663* (1983)
248. A.N. Tikhonov: On solving incorrectly posed problems and method of regularisation. *Dokl. Akad. Nauk USSR* **151** (1963)
249. S. Timoshenko, J.N. Goodier: *Theory of Elasticity* (McGraw-Hill, New York, 1951)
250. P. Venhovens: *Optimal Control of Vehicle Suspensions*. PhD thesis, Delft University of Technology, Delft, The Netherlands (1994)

251. R. Viennet, C. Fonteix, I. Marc: Multicriteria optimization using a genetic algorithm for determining a pareto set. *Int. J. Syst. Sci.* **27**(1) (1996)
252. I. Crivelli Visconti: *Materiali compositi: tecnologie e progettazione* (Tamburini, Italy, 1975)
253. A.M. Wahl: *Mechanical Springs* (Mc-Graw Hill, New York, 1963)
254. M. Walker, T. Reiss, S. Adali: Multiobjective design of laminated cylindrical shells for maximum torsional and axial buckling loads. *Comput. Struct.* **62**(2) (1997)
255. H. Wallentowitz: Integration of chassis and traction control systems. What is possible? What makes sense? What is under development? In: *Proc. AVEC'92 Symp. JSAE*, Japan, 1992
256. F.M. Waltz: An engineering approach: hierachical optimization criteria. *IEEE Trans.* **12** (1967)
257. A. Warburton: Quasiconcave vector maximization: connectedness of the sets of pareto-optimal and weak pareto-optimal alternatives. *J. Optim. Theor. Appl.* **40**(4) (1983)
258. S. Warren: Neural networks and statistical models. In: *Proc. 19th SAS Users Int. Conf.*, Omaha, NE, 1994
259. A.H. Wickens: The dynamics of railway vehicles – from Stephenson to Carter. *Proc. Inst. Mech. Eng. – Part F J. Rail Rapid Transit* **212**(3) (1998)
260. H.P. Willumeit et al.: Identification of fuzzy control systems by evolution strategy. In: *Proc. AVEC'94 Symp.*, Japan, 1994
261. W.H. Wittrick: On elastic wave propagation in helical springs. *Int. J. Mech. Sci.* **8** (1966)
262. X. Xia, J.N. Willis: The effects of tire cornering stiffness on vehicle handling linear behavior. *SAE Trans.* (1996)
263. N. Yamaki: Elastic stability of circular cylindrical shells. *North-Holland Ser. Appl. Math. Mech.* **XIII** (1984)
264. M. Yamamoto: Active control strategy for improved handling stability. *SAE Publ.* (1991)
265. E. Yeh et al.: A genetic algorithm based fuzzy system for semi-active suspension system design. In: *Proc. AVEC'94 Symp.*, Japan, 1994
266. T. Yokota et al.: A solution method for optimal weight design problem of helical spring using genetic algorithms. *Comput. Ind. Eng.* **33**(1/2), 71–76 (1997)
267. K. Yoshikawa, H. Tani, H. Watanabe, H. Suzuki: Analysis of shift operating phenomena for manual transmission. In: *CSAT*, Czech Republic (1968)
268. S.S. You, Y.H. Chai: Multi-objective control synthesis: an application to 4ws passenger vehicles. *Mechatronics* **9** (1999)
269. S.S. You, S.K. Jeong: Controller design and analysis for automatic steering of passenger cars. *Mechatronics* **12** (2002)
270. W.C. Young: *Roark's Formulas for Stress and Strain* (McGraw-Hill, Singapore, 1989)
271. J.R. Zahng, S.J. Xu, A. Rachid: Robust sliding mode observer for automatic steering of vehicles. In: *IEEE Intell. Transport. Syst.*, vol. 1, Dearborn, MI (2000)
272. A. Zaremba, R. Hampo, D. Hrovat: Optimal active suspension design using constrained optimization. *J. Sound Vibr.* **207** (1997)
273. W.H. Zhang, H.C. Yang: Efficient gradient calculation of the pareto optimal curve in multicriteria optimization. *Struct. Multidiscipl. Optim.* **23** (2002)

Index