Linear and Reconfigurable Control of Wing Damaged Aircraft

By

PASCAL NESPECA
B.S. (University of California, Davis) 2003
M.S. (University of California, Davis) 2006

DISSERTATION

Submitted in partial satisfaction of the requirement for the degree of
Doctor of Philosophy

in

Mechanical and Aeronautical Engineering

in the

OFFICE OF GRADUATE STUDIES

of the

UNIVERSITY OF CALIFORNIA

DAVIS

Approved:

Prof. Nesrin Sarigul-Klijn, chair _____

Prof. Ron Hess _____

Prof. A. Nazli Gundes _____

Committee in Charge

2009

Table of Contents

*Pascal Nespeca

**December 2009

***Mechanical and Aerospace Engineering

Linear and Reconfigurable Control of Wing Damaged Aircraft

Abstract

Recently, there has been an interest in researching control techniques that might improve the overall safety of flight. The goal is to create an autopilot control system which could safely land a wing damaged aircraft.

Spanwise Full-Loss (SFL) is defined as the entire removal of wing section along the chord of the wing, starting from the tip and moving toward the root. Based upon computational models of a rigid aircraft with varying SFL, obvious force-moment imbalances are likely to be the primary factor affecting survivability. Rigid aircraft with more effective ailerons or additional rolling control surfaces are more likely to survive wing damage.

Computer models of wing damage suggest that wing loss in the range of 0-50% SFL will not create an abnormal dynamic instability of a rigid aircraft with a standard linear autopilot. Dynamic instability is not present because the SFL linear model is mostly triangular with longitudinal variables almost exclusively effecting lateral variables. Closed loop performance is not compromised in the range of 0-20% SFL. Resizing ailerons may be needed to accommodate wing damage beyond 20 to 30% SFL.

For a flexible aircraft, wing damage that reduces the torsional stiffness of the wing could cause roll-control reversal. Roll control reversal can create closed loop instability with undamaged aircraft. SFL actually increases the torsional stiffness of the wing. However, real world wing damage may not be limited to a spanwise wing loss.

Conventional control techniques are introduced by several design examples and successfully extended to wing damaged aircraft. Reconfigurable, switching and conventional control techniques are found to possess acceptable levels of technical merit for flight control. With reconfigurable and switching flight control techniques, one can avoid known instabilities due to time varying gain by simply waiting 6 to 20 seconds between controller switches. Many direct adaptive control and indirect adaptive control techniques encounter problems with instability. Stability problems with adaptive control techniques can

be fixed by gain limiting. However, performance is still unpredictable due to the rapid time varying gains in adaptive control systems. Therefore, adaptive control is not recommended for flight control.

N.1 Mathematical Notation

$A_{i,j}$ = the i,j entry of the matrix A

$A_{i,\bullet}$ = i^{th} row of the matrix A

$A_{\bullet,j}$ = j^{th} column of the matrix A , (j is an integer)

$\left(A_y\right)_{i,j}$ = the i,j entry of the subscripted matrix A_y

A^T = Transpose of the matrix A

$\lambda[A]$ = Set of eigenvalues of the matrix A

$\boldsymbol{det}[A]$ = Determinant of the matrix A

$\|A\|$ = A generic norm of the matrix A

$\sigma[A]$ = Set of singular values of the matrix A

$\overline{\sigma}(A)$ = Maximum singular value of the matrix A

$\underline{\sigma}(A)$ = Minimum singular value of the matrix A

$\boldsymbol{Re}\{y\}$ = Real portion of the complex variable y

$\boldsymbol{Im}\{y\}$ = Real portion of the complex variable y

$\mathbf{C}$ = The set of complex numbers

$\Re$ = The set of real numbers

$\varnothing$ = The null or empty set

N.2 Control Notation

x = state vector, dimension n x 1

$\dot{x}$ = time derivative of the state vector, dimension n x 1

y = output vector, dimension p x 1

u = input vector, dimension m x 1

t = time, sec.

s = Laplace variable, $\partial/\partial t$, scalar

z = Digital forward shift operator, $x(t+1) = x(t)z$, scalar

j = $\sqrt{-1}$, Complex number

ω = frequency, rad./sec.

e^{-tds} = time delay of *td* seconds

$\frac{1}{s}$ = integrator

s = differentiator

$\varepsilon_L(s)$= left interactor matrix

A = state transition matrix, dimension $n \; x \; n$

B = state-input matrix, dimension $n \; x \; m$

C = state-output matrix, dimension $p \; x \; n$

D = input-output matrix, dimension $m \; x \; p$

$G(s)$ = Plant, dimension $m \; x \; p$, $G(s) = C(sI - A)^{-1} B + D$

$C(s)$ = Controller or Compensator, dimension $p \; x \; m$

$L(s)$ = Loop Shape, dimension $p \; x \; p$

$T(s)$ = Compimentary Sensitivity at output, dimension $p \; x \; p$

$S(s)$ = Sensitivity at output, dimension $p \; x \; p$

$S(s)^{-1}$ = Return difference, dimension $p \; x \; p$

$T_I(s)$ = Compimentary Sensitivity at input, dimension $p \; x \; p$

$S_I(s)$ = Sensitivity at input, dimension $p \; x \; p$

$\Delta(s)$ = General uncertainty matrix, dimension varies

$\Delta_I(s)$ = Uncertainty matrix at input, dimension $m \; x \; m$

$\Delta_o(s)$ = General uncertainty matrix, dimension $p \; x \; p$

K_x = State feedback matrix, dimension $m \; x \; n$

L_o = Observer matrix, dimension $n \; x \; p$

K_y = Proportional output feedback matrix, dimension $p \; x \; m$

K = generic gain, state feedback or output feedback

σ = real portion of an s value

$\sigma(G(j\omega))$ = set of singular values of G at frequency ω

$G(s)_{22}\big|_{y1 \to u1}$ = the 2,2 transfer function of G with y1 paired with u1

GM = Gain Margin

PM = Phase Margin

ω_{BW} = Bandwidth frequency, the frequency at which $\|T(j\omega)\|$ crosses the -3dB line

$RGA(G(s))$ = The relative gain array of the plant

N.3 Aircraft Notation

δ_a = Aileron deflection angle, usually rad.

δ_r = Rudder deflection angle, usually rad.

δ_e = Elevator deflection angle, usually rad.

δ_T = Thrust, usually lbs. or N

α = Angle of Attack (AOA), rad., $\alpha \approx {}^{w}/_{U_o}$

β = Sideslip angle, rad., $\beta \approx {}^{v}/_{U_o}$

U_o = Body-fixed forward speed (for linearization) , m./sec. or ft./sec.

u = Body-fixed forward speed perturbation, m./sec. or ft./sec

w = Body-fixed downward velocity, m./sec. or ft./sec.

v = Body-fixed Lateral velocity, m./sec. or ft./sec.

p = Body-fixed roll rate, rad./sec.

q = Body-fixed pitch rate, rad./sec.

r = Body-fixed yaw rate, rad./sec.

SL = Sea-Level

FRL = Fuselage Reference Line

AOA = Angle of Attack

M = Mach number

$\bar{q}$ = dynamic pressure

dmg = portion of wing which is removed, starting from the tip

SFL = Spanwise Full Loss, method of wing damaging by progressively removing wing from tip to root

PA = Power Approach

Acknowledgements

The author would like to thank all of those who offered direction, advice, opinion and support during the process of writing this dissertation.

Chapter 1 – Introduction

Recently, there has been an interest in using control techniques that might improve the overall safety of flight. The overall goal is to create an autopilot control system which could adjust to off-nominal in-flight scenarios and to land safely at an airport, as shown in Fig. 1.1.

Fig. 1.1. The goal of this work is to assess the possibility of safe landing or crashing after wing damage and to recommend safe control strategies.

An example of such a scenario would be icing. Icing occurs when ice forms over the surface of a wing or other control surface such as a rudder. This could cause decreased lift or possibly make a control surface immovable. In 1994, icing caused a commuter flight to crash with no survivors [1.1].

Pilots can sometimes recover from these abnormal flight scenarios. Icing does not always cause a crash, as the pilot of American Eagle Flight 3008 found in 2005 [1.2]. One pilot was able to land an aircraft with a wing missing or a severely disfigured wing. The photo shown in Fig. 1.2 is of a F-15 fighter aircraft that suffered a midair collision during a training exercise [1.3]. The pilot was able to land the aircraft safely.

Pilots do not always respond correctly to upsets. If a pilot improperly handles a situation, aircraft that are not damaged at all can be destroyed. For example, American Airlines Flight 587, an A300-600, crashed on 12 November 2001 because of over-aggressive rudder inputs from the pilot. The pilot over-

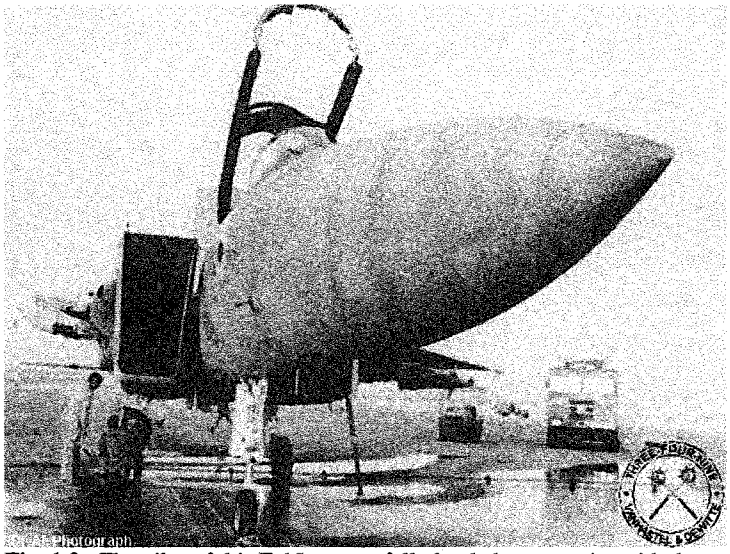


Fig. 1.2. The pilot of this F-15 successfully landed on a carrier with the right wing torn off completely. This resulted from a mid-air collision during a training excercise.

compensated due to some turbulence from a nearby B-747. The over-aggressive response from the pilot overstressed the aircraft's vertical stabilizer, and caused it to snap off entirely.

Ideally, the controller would do what a good pilot would do in the event of an upset/damage incident. In this dissertation, the primary focus will be on autopilot design and then applying these controllers to wing damaged aircraft. Some decoupling design techniques are extended to wing damaged aircraft as well.

1.1 Practical Limitations of Reconfigurable Autopilots

There are at least two known limiting factors associated with designing reconfigurable autopilots.

The first is the limitation of situational awareness. This means that it is not really possible to know exactly what is wrong with the aircraft, if there even is anything wrong with the aircraft. In truth, one can only form an educated guess of what is happening.

The second limitation is related to control system design. There are actually a great many limitations associated with control system design. More on the subject of control systems limitations can be found in Refs. [1.4 - 1.8] and is briefly discussed in sections 2.27 and 2.28.

1.1.1 Perfect Situational Awareness is Impractical

Some aviation accidents which undergo thorough investigations of wreckage, flight data recorders (FDR) and cockpit voice recorders (CVR) do not reach conclusive findings. From 1999 to 2008, a total of 120 fatalities from commercial jet airplanes do not have a known cause [1.9].

A panel of aviation experts cannot always determine what exactly went wrong on an aircraft given nearly all available information even months after the incident. Therefore, it should be impractical

to expect an engineer to design an omnipotent computer algorithm which will perfectly diagnose an aircraft fault during its flight. Inevitably, errors will be made in fault detection and situational awareness.

1.1.2 Limitations of Feedback Control

Feedback control systems are limited by the hardware which is available. Suppose we made the analogy between a human being and an aircraft. An aircraft has moveable surfaces such as a rudder which move the aircraft around. A human has legs and arms which help the human move. These things are called actuators. Aircraft also have things which tell the aircraft where it is moving and how it is oriented. A human has sight, sound, smell, taste and touch to make a human aware of where it is and what it is doing. These things are called sensors.

Suppose that an aircraft had an actuator failure and could no longer move its rudder. This would be somewhat analogous to a human with an arm or leg missing. It would be unreasonable to expect the aircraft or the human to move normally.

Similarly, if an aircraft had a sensor failure and could not determine its pitch angle, then one might reasonably expect the aircraft to climb rapidly, stall, and then fall to the ground. Also, if someone suddenly lost function of their inner ear mechanism, then one would expect that person to lose balance and fall down or at least find difficulty walking straight.

Sensor failures will inhibit a control system's ability to see what is happening. The potential to see everything is technically referred to as full state observability. Actuator failures will also negatively affect a control system's ability to do things. The potential to completely control things is called state controllability. A more restrictive subset of controllability which is related to inverse dynamics is called functional controllability.

An aircraft which loses observability and/or controllability does not really become a very strong candidate for control. Control, like any control system, must have reasonable goals. For instance, if a human lost a pinky or ring finger, one can still write, so this is a reasonable control task. If a small flap on an aircraft became stuck, then the aircraft should still be controllable and observable. This is a reasonable focus of reconfigurable control.

Limitations of performance and stability of control systems is also an important topic in control. More on the subject of control systems limitations can be found in section 2.29, 2.30 and Refs. [1.4 - 1.8].

1.2 The Challenge of Improving Safety with Reconfigurable Autopilots

We now express safety concerns using conditional probabilities [1.10]. We suppose that there are two designs, D_0 and D_1. The first design, D_0, is designed for the normal aircraft alone. The second design, D_1, is reconfigurable and can switch between an autopilot designed for the upset/damaged aircraft and an autopilot designed for the normal aircraft. We also suppose that there are two events E_0 and E_1. The first event, E_0, is the normal flight condition of the aircraft. The second event, E_1, is the abnormal or damaged flight condition. The C symbol represents a crash.

Suppose that the first design, D_0, is 100% safe for the normal flight condition. Therefore, we say that the probability of crash, C, with the first design, D_0, given normal flight, E_0, is $P(CD_o|E_o)=0$. Suppose that the first design, D_0, is only 50% safe for the abnormal or damaged flight condition. We say that the probability of crash, C, with the first design, D_0, given abnormal flight condition, E_1, is $P(CD_o|E_1)=0.5$.

Now suppose that there was a second, reconfigurable design, D_1, that tried to figure out what was happening and switched the autopilot around. Suppose that this reconfigurable design was successful at preventing an accident given E_1 with 99% success. Thus, the probability of crash, C, with the second design, D_1, given abnormal flight condition E_1 is $P(CD_1|E_1)=0.01$. Also, suppose that the second design, D_1, misdiagnosed the normal flight condition and used an autopilot for an abnormal flight condition for the normal flight condition which caused a crash with 1% probability. The probability of crash, C, with the second design, D_1, given normal flight condition E_0 is $P(CD_0|E_1)=0.01$.

We now calculate the total probability of a crash to compare both designs with $P(E_0)=0.999$ and $P(E_1)=0.001$.

$$P(CD_0) = P(CD_0|E_0)P(E_0) + P(CD_0|E_1)P(E_1) = 0(0.999) + 0.5(0.001)$$
$$P(CD_1) = P(CD_1|E_0)P(E_0) + P(CD_1|E_1)P(E_1) = 0.01(0.999) + 0.01(0.001)$$
$$P(CD_0) = 0.0005$$
$$P(CD_1) = 0.01$$

Notice that the first design, D_0, is safer by a factor of 20.

Now suppose that abnormal flight conditions were more frequent and $P(E_0) = 0.9$ and $P(E_1) = 0.1$. We recalculate the following conditional probabilities below.

$$P(CD_0) = P(CD_0|E_0)P(E_0) + P(CD_0|E_1)P(E_1) = 0(0.9) + 0.5(0.1)$$
$$P(CD_1) = P(CD_1|E_0)P(E_0) + P(CD_1|E_1)P(E_1) = 0.01(0.9) + 0.01(0.1)$$
$$P(CD_0) = 0.05$$
$$P(CD_1) = 0.01$$

Now notice that the second design that tries to figure out what is going on is now safer by a factor of 5.

In order to really justify a reconfigurable autopilot, one really needs to show that the probability of having abnormal flight conditions is high. Otherwise, it is possible that a reconfigurable autopilot can do more harm than good.

1.2.1 Application of Reconfigurable Autopilots to Commercial Air Transportation

With commercial air transportation, it is possible that a reconfigurable autopilot may do more harm than good. In 2003, the most dangerous year between 1987 and 2006, aircraft accidents occurred 0.000518 % of the time [1.11]. Most of these accidents were non-fatal.

Now, suppose that there were a detection algorithm that could correctly identify an upset/damage/abnormal flight scenario 95% of the time and select the correct autopilot designed for the upset like the scheme in Ref. [1.12]. Further suppose that a detection algorithm gave a false positive result, selected the wrong autopilot and caused an accident with a 0.001% probability.

If such a detection/selection scheme were in place in 2003, accidents would have occurred at a rate of 0.0010259 %, effectively doubling the likelihood of having an accident. This is due entirely to the 0.001% occurrence of false positive and misdiagnosis.

While autopilot designers may have their eye on the upset/damage/abnormal flight scenario, it is important that the controller of the damaged aircraft be completely non-threatening to the normal aircraft. This is difficult to motivate because an upset controller that is 99% non-threatening to the nominal aircraft together with a false positive and misdiagnosis at a rate of 0.01% would produce that accident rate of 0.001%.

What is important to note is that in order for there to be any improvement in safety, the reconfigurable autopilot must first be 100% or 99.9999% benign to the nominal aircraft. Nearly no new failures can be introduced at all. It is therefore a challenge to introduce an autopilot specifically designed for a failure scenario without introducing new failiures. The autopilot designed for the failure scenario will inevitably be applied to normal flight conditions.

1.2.2 Application of Reconfigurable Autopilots to Military Aircraft

It is perceived that military aircraft in active combat zones are more prone to abnormal and damaged flight conditions than commercial aircraft. Although no specific data is known to the author, the probability of a damage or upset flight condition which is recoverable is most likely higher than commercial aviation. Therefore, one might find motivation for designing a reconfigurable autopilot for a military aircraft. The current F/A-18 super hornet design has a reconfigurable control system [1.13].

Failure rates for military aircraft, especially fighter aircraft, are more relaxed because parts undergo more strain during service. This can be due to both high sortie rates and more aggressive maneuvering during service.

This does not mean that these challenges have disappeared. The C-17 was designed to meet a catastrophic failure rate of $1x10^{-9}$ per mission [1.14]. This is quite small. It is still a challenge to meet these rigorous standards. It is also a challenge to design and implement tests proving that such failure rates have actually been met.

1.3 Survivability of Wing Damaged Aircraft

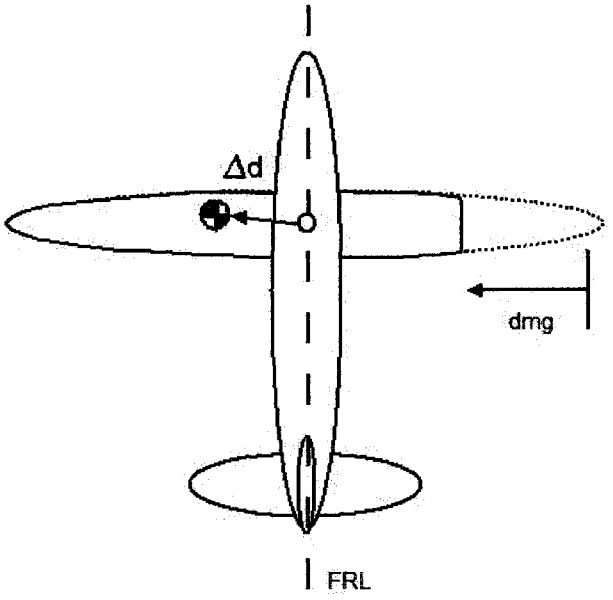


Fig. 1.3. Definition of Spanwise Full Loss (SFL) wing damage, which is given a dmg value between 0 and 1.

There are several components to the assessing survivability of wing damaged aircraft. First, one must show that all of the forces and moments can be balanced. Balancing forces and moments is called "trimming" the aircraft. If an aircraft can be trimmed, then it is *statically* stable. Second, one must show that the aircraft is closed loop stable with a pilot model or autopilot. Thus, we would say that the aircraft is *dynamically* stable. Third, one might optionally show that the aircraft meets some performance criteria in the presence of damage.

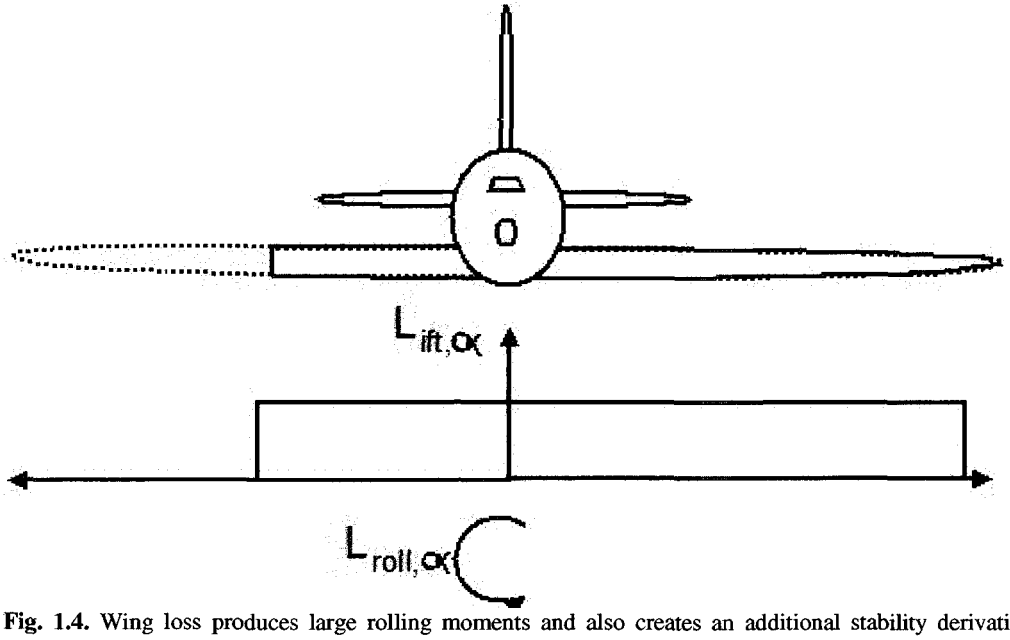


Fig. 1.4. Wing loss produces large rolling moments and also creates an additional stability derivative, which is rolling moment with respect to angle of attack, $L_{roll,\alpha}$.

The definition of wing damage will be a progressive loss of the wing from the tip to the root, as shown in Fig. 1.3. The damage parameter, dmg, ranges from 0 to 1.

It should be fairly intuitive that the aircraft is susceptible to rolling over in the presence of wing damage. This is probably the most significant concern for the survivability of a wing damaged aircraft.

From Fig. 1.4, one can see that the lift distribution is clearly imbalanced. The aircraft control surfaces must be capable of balancing these imbalances. Additionally, this force imbalance will vary with angle of attack (AOA).

1.3.1 Trim Conditions of Wing Damaged Aircraft

Large control surfaces should help with survivability with regard to wing damage. Below is a table comparing 3 various aircraft's fraction of total aileron area to total wing area (total aileron area consists of any control surface that creates rolling moment on the wings).

Table 1.1 Comparison of various sizes of aircraft aileron area and wing area. One can see that the author can tend to overestimate the size of ailerons from drawings, though only by about 1 square meter in the case of the F-15.

Aircraft	Wing Area (m^2)	Total Aileron Area (m^2)	Fraction	Source
C-17	353	11.83	0.033	Jane's All the World's Aircraft 2007-2008 [1.15]
F-15	56.49	2.46	0.043	Jane's All the World's Aircraft 2007-2008 [1.15]
F-15	56.49	3.5	0.063	Author estimated from 3-view drawing
A-10	47.02	5.51	0.117	Author estimated from 3-view drawing
P-17	353	35 (equivalent)	0.099	This is a fictionally modified mathematical model of a C17. Plain aileron, asymmetric flap and differential tail are used as a "mega-aileron"

One should also note that maximum aileron deflection should also be a significant factor in the survivability of an aircraft's ability to survive wing damage. The author was not able to find reliable data

on the maximum control surfaces for these aircraft, but control surface deflections are usually set somewhere between 5 and 30 degrees. One would expect fighter aircraft to have larger control surface limits than transport aircraft due to the need for greater maneuverability. These limits will vary based upon airspeed and altitude. Control surface limits are used so that the pilot or autopilot cannot cause a control induced upset. Yet, resilience to larger wing damage will most likely require large control inputs. Designing an aircraft which may potentially undergo wing damage may place the normal aircraft at an increased risk.

It is rumored that the A-10 was originally designed to survive in the presence of 50% wing damage. Notice that the ailerons make up about 10% of the total wing area. It is known that one F-15 survived a full wing loss scenario [1.3]. Yet, its fraction of aileron area to wing area is only about 4.3%. This may be due to the fact that the pilot was flying very fast and had sufficiently large aileron deflections available. One can see this from the trend in Fig. 1.5.

Aileron areas and deflection limits will be larger for fighter aircraft than for transport aircraft. One should not expect large transport aircraft to survive the same amount of wing damage.

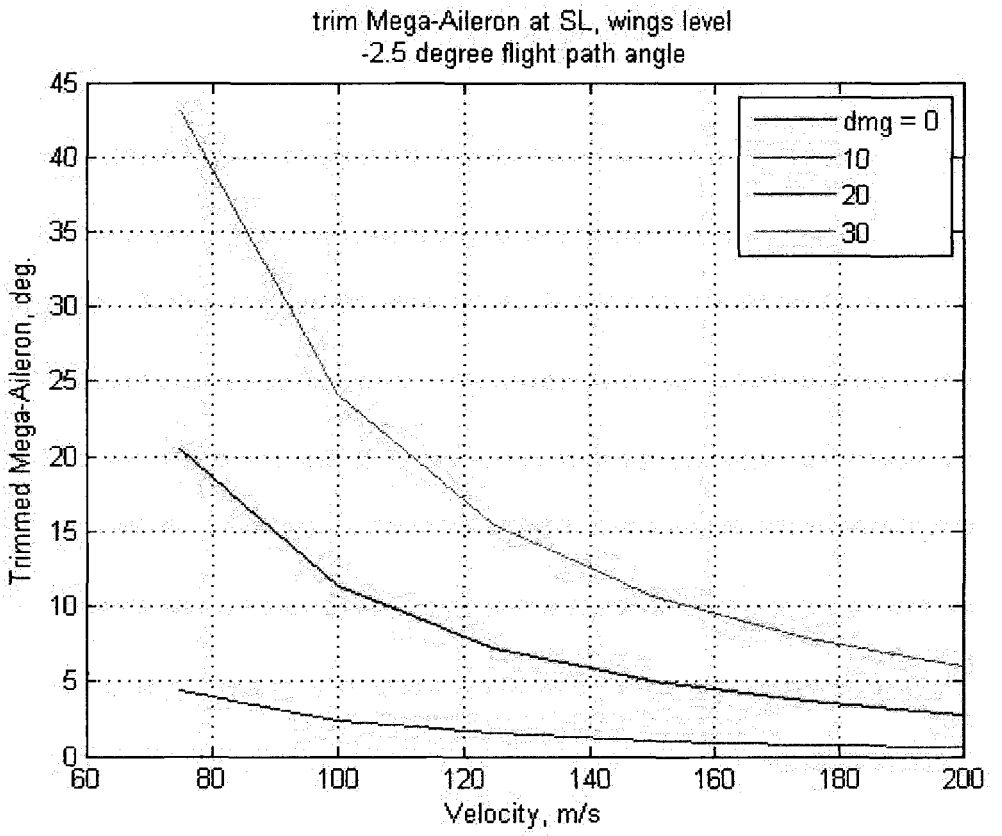


Fig. 1.5. It takes a lot of effort from the mega-aileron to counteract basic rolling moments produced by wing damage. This data comes from rigid body dynamics. Trimmed sideslip angles were less than 1 degree. Trimmed angle of attack angles did deviate by more than 2 degrees with the damage less than 30%. This is for the aircraft without any cargo. With cargo, it is expected that the trimmed mega-aileron deflection would increase by about 5 to 10 degrees at low speed because the lateral c.g. shift would be less pronounced.

The special P-17 is outfitted with a mega-aileron which consists of an oversized plain aileron, asymmetric flap and differential tail. The mega-aileron is limited not to exceed 30 degrees of deflection. From Fig. 1.5, one can see that even with the mega-aileron, the aircraft is not likely to survive damage at 30% of wing loss at low speed. Large transport aircraft are usually designed to land at about 80 m./sec. and sometimes slower, it is certainly questionable as to whether such a large aircraft could land safely at velocities above 100 to 140 m./sec.

Trimmed sideslip angle, angle of attack, elevator deflection and rudder deflection values were all investigated for the modeled P-17 using a sophisticated trimming subroutine. Trimmed sideslip angles did not deviate by more than 1 degree. Trimmed angle of attack did not increase by more that 2 degrees for

Table 1.2 Comparison of wing damage incidents involving large transport aircraft

Aircraft	Wing Damage, %	Survivors	Probable Cause	Year	Operator	Source
Lockheed Electra 188C	100	0	Structural Fatigue	1960	North West Airlines	[1.16]
Boeing 707	100	0	Lightning struck wing, igniting fuel	1963	Pan Am Airlines	[1.16]
Boeing 777	1 (relative)	All	Landing gear punctured wing	2008	Saudi Airlines	[1.17]
Airbus A300	10 (relative)	All	Terrorist shot missile at wing, Only thrust control available to pilots	2003	DHL	[1.18]

damage less than 40%. Trimmed elevator and rudder deflections also did not show a significant change.

Consider the following incidents in table 1.2. Large aircraft do not have a history of recovering from large or total wing loss. However, large transport aircraft do have a history of recovering with small amounts of wing damage. The International Civil Aviation Organization (ICAO) does *not* consider the following things to be *substantial* damage: engine failure or damage limited to one engine, bent fairings or cowlings, dents in the skin, small puncture holes in the skin, damage to wheels, damage to tires, damage to flaps, damage to engine accessories and damage to wingtips [1.9].

1.3.2 Dynamic Stability of Wing Damaged Aircraft

Studies have been conducted which show that open loop stability of large wing damaged aircraft is likely [1.19]. Closed loop stability with an ordinary autopilot is also expected to occur. This is investigated in sections 4.5, 5.7, and 7.15.

1.3.3 Performance of Wing Damaged Aircraft

Wing damaged aircraft should not be expected to perform like normal aircraft. Decoupling control design procedures pursued in section 5.7 with a modest amount of modeling error failed to produce substantial benefits in performance given limitations in bandwidth.

1.4 Literature Review of Control Techniques

Currently, there are several popular methods in designing fault tolerant flight control systems. Among these methods are reconfigurable control methods, conventional and robust control methods, and adaptive control methods. Recently, a knowledgeable review of adaptive control (not including reconfigurable or conventional control systems) has been published [1.20].

1.4.1 Direct Adaptive Control

The author naïvely became interested in direct adaptive control methods at the beginning of his dissertation work [1.21]. The vast majority of direct adaptive control methods are dangerous because they try to enhance performance by increasing the gain to very high levels while the plant is in operation.

Since its inception in 1959 by H.P. Whitaker [1.22], direct adaptive control methods have suffered from instability due to a tendency to use very high gain. In the earlier part of the 1980's, some publications pointed out the tendency of these adaptive control methods to integrate to infinite gain [1.23, 1.24].

In response to these problems, the academic community has sought to improve direct adaptive control by making small changes to differential equations that govern adaptation. Some of these modifications include sigma modification [1.25] and e-modification [1.26]. Neither method works well, although these methods are popular in literature. The best thing to do is to model the dynamics of the system and use gain limiting or parameter projection. This is discussed in chapter 8.

The solution to making direct adaptive control schemes stable is available. However, it involves having at least an approximate model of the dynamics of the plant. In a way, this destroys the purpose of adaptive control: to design controller for a completely unknown plant while the plant is in operation.

Having stated the purpose of adaptive control in a sentence; one can see that there might be problems. Ordinary control design always requires at least an approximate model for the dynamics of the plant. Designing a controller without a model of the plant is simply not a good idea. Also, designing a controller while the plant is in operation is also not a good idea due to the potential for instability with time varying gain, as seen in chapter 8.

1.4.2 Real Time Indirect Adaptive Control

This method of adaptive control is less fundamentally flawed than direct adaptive control. The idea is that one performs conventional control synthesis while performing system identification, such as

Recursive Least Squares (RLS) [1.27]. System identification methods use input and output data alone to create a model of the plant. This is close to what an ordinary control designer would be doing, except that it is performed while the plant is in operation. This leads to potential problems. One problem is that these authors sometimes neglect that instability can develop merely from using time varying gain [1.28]. The second problem is that there is typically no check to see whether or not the system identification is giving reasonable data. Real time system identification routines, such as RLS identification, can diverge. This typically causes unreasonable controllers to be designed and can cause instabilities. Real time indirect adaptive control is also a research area to avoid for much of the same reasons that direct adaptive control should be avoided. There is sometimes a desire in this research area for a totally generalized controller that is always adjusting its parameters.

1.4.3 Iterative Identification and Control

Iterative identification and control is an improved version of real time indirect adaptive control. The idea is that one performs identification and control synthesis separately. Furthermore, there is verification that system identification is correct before performing control synthesis [1.29]. Additionally, batch processing system identification is typically used instead of real time system identification methods. Controller updates are considerably slower than real time indirect adaptive control. Therefore, instability from time varying control gain is less likely.

Iterative identification and control is very close to how conventional control systems are designed. The only difference is that a control engineer must put all of his or her thoughts on control techniques into a single control subroutine. This is somewhat difficult, but a computer code known as CONDUIT [1.30] apparently works fairly well for helicopter and aircraft control synthesis. Also, a computer code known as CIPHER [1.31] is a widely used batch system identification code used for helicopters and aircraft.

This method of adaptive control is less unreasonable than previously mentioned adaptive control techniques. The problem which is uniquely associated with flight control is that a gain scheduled controller might be needed for an aircraft with a large flight envelope. *System identification only estimates an aircraft model for what has happened in past flight conditions. However, to develop a full gain scheduled autopilot in flight, one must also know what will happen in future flight conditions.*

Clearly, this is not possible with system identification alone. Without clairvoyance, a test pilot or something similar will always be needed to get flight data before a high quality autopilot can exist.

Despite this inherent and unavoidable problem, one might simply hope that the immediate in-flight control design is robust to changes in flight condition. In this case, the author recommends that controller updates are separated by at least 6 seconds for lateral control of large transport aircraft. Switching controllers every 20 seconds or greater is safer. Reasons for this are discussed in chapter 8.

Assuming this method would actually work, the remaining problem with this method is that it takes time to get data for system identification and control design. To get high quality aircraft data for a large frequency range, one typically needs between 60 to 100 seconds, at least. If one is only interested in dynamics of frequency ranges faster than about 0.5 rad./sec., about 5 to 20 seconds of flight data is sufficient. In the meantime, who or what is flying the aircraft? It seems as though this method can only work for damage scenarios which are quite mild and do not require immediate attention.

However, this does not one should completely disregard this method. One might successfully use this for precision landing approaches with uncertain dynamics.

1.4.4 Reconfigurable Control

Reconfigurable control requires specific knowledge of the plant and possible failures which can occur. Designing reconfigurable controllers takes more design effort than adaptive controllers. The reward for designing reconfigurable controllers is that one can assess the impact of a reconfigurable control system. Reconfigurable control systems can be flight tested to verify efficacy because the total reconfigurable system does not change in flight after it has been designed. Adaptive controllers cannot be thoroughly tested because the controller may do something different each day of testing. Reconfigurable control is restricted to a finite set of situations and control schemes.

Reconfigurable control consists of two components: fault detection and isolation and switching control.

Fault detection and isolation is a kind of situational awareness to aid or replace pilot awareness. First, there is the recognition that there is a fault. This process is called fault detection. Next, there is an effort to determine which fault has occurred. This is called isolation of the fault.

An introduction to techniques for fault detection and isolation can be found in Ref. [1.32]. Beware that some of the observer-based fault detection schemes found in Ref. [1.32] are flawed due to the way that these observers are placed in the feedback loop. A better implementation can be found in Ref. [1.33]. Also, one should note that using an observer really just filters some input and output data. It is still the task of the engineer to look at data and determine what exactly constitutes a fault and what does not. Observer design is not always necessary and may needlessly distract the engineer from thinking about what is physically going on with the aircraft. Designing and placing new sensors specifically for one type of fault may be necessary.

The second design aspect of reconfigurable control is the switching control aspect. Switching control inevitably involves the usage of time varying gain. As mentioned before, time varying gain can cause instability purely because the gain varies with time [1.28]. The author recommends that controller switches are separated by at least 6 seconds for lateral control of large transport aircraft. Switching controllers every 20 seconds or greater is safer. Reasons for this are discussed in chapter 8.

1.4.5 Conventional and Robust Control

Conventional and robust control methods are superior to reconfigurable control in that it does not require switching. Stability issues with time varying control are not present with this method. However, one of the drawbacks with these methods are that one might have difficulties simultaneously satisfying performance and stability requirements for separate damage/upset scenarios with only one controller. For aircraft control, it is sometimes not the case that the control designer finds himself or herself in such a dilemma. Simultaneously designing a longitudinal controller for a F-16 with an unstable short period mode and a F-16 with a stable short period mode has a challenge. Feeding back angle of attack into the elevator stabilizes the unstable F-16. However, this feedback also increases the frequency of the short period mode for the stable F-16. Despite this inherent trade-off, a reasonable balance can be made and a single controller can work for both flight conditions.

Various methods of non-adaptive and non-switching controllers are available. Usage of high gain and sliding mode control methods [1.34] are more difficult than conventional control design. However, these methods are still achievable in an engineering setting. Single controllers which are robust to sensor and actuator failures are possible. Some autopilots can be designed for sensor and actuator

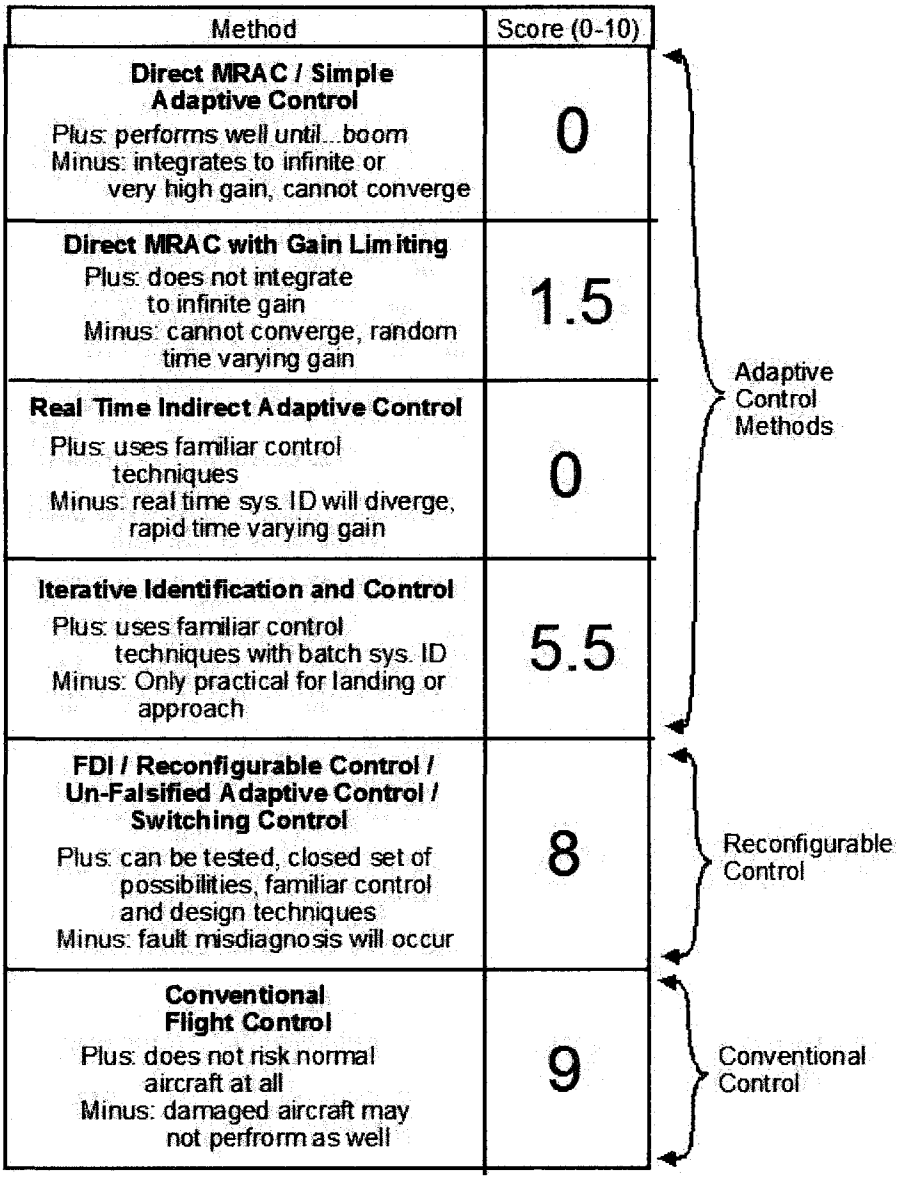


Fig. 1.6 Comparison of control methods for damage/upset aircraft

failures. Analyzing this property is called broken loop analysis [1.35]. A theoretical discussion of the feasibility and design of these controllers can be found in Ref. [1.36].

It is always preferable to solve problems with conventional control instead of reconfigurable control. One should attempt conventional and non-switching control methods before considering reconfigurable control.An overall rating of control methods is shown in Fig. 1.6.

Chapter 2 – Background Material – Dynamics and Control

There are many thorough texts in the area of system dynamics and control. Single-Input Single-Output (SISO) concepts can be found in Refs. [2.1,2.2]. Extensions to Multi-Input-Multi-Output (MIMO) can be found in Refs. [2.3-2.5]. Advanced perspective on linear systems theory can be found in Ref. [2.6].

2.1 Transfer Functions

Many things which occur as a function of time, $y(t)$, can be expressed in an "s" domain with a Laplace transform shown in eq. 2.1.

$$y(s) = \int_0^\infty y(t) e^{-st} dt \tag{2.1}$$

The idea of the "s" domain is that s represents a complex number which consists of an imaginary number and a real number, $s = \sigma + j\omega$. From euler's identity, $e^{st} = e^{\sigma t}(\cos(\omega t) + j\sin(\omega t))$. If a pair such as $s = \sigma \pm j\omega$ were transferred into the time domain with an inverse laplace transform of eq.

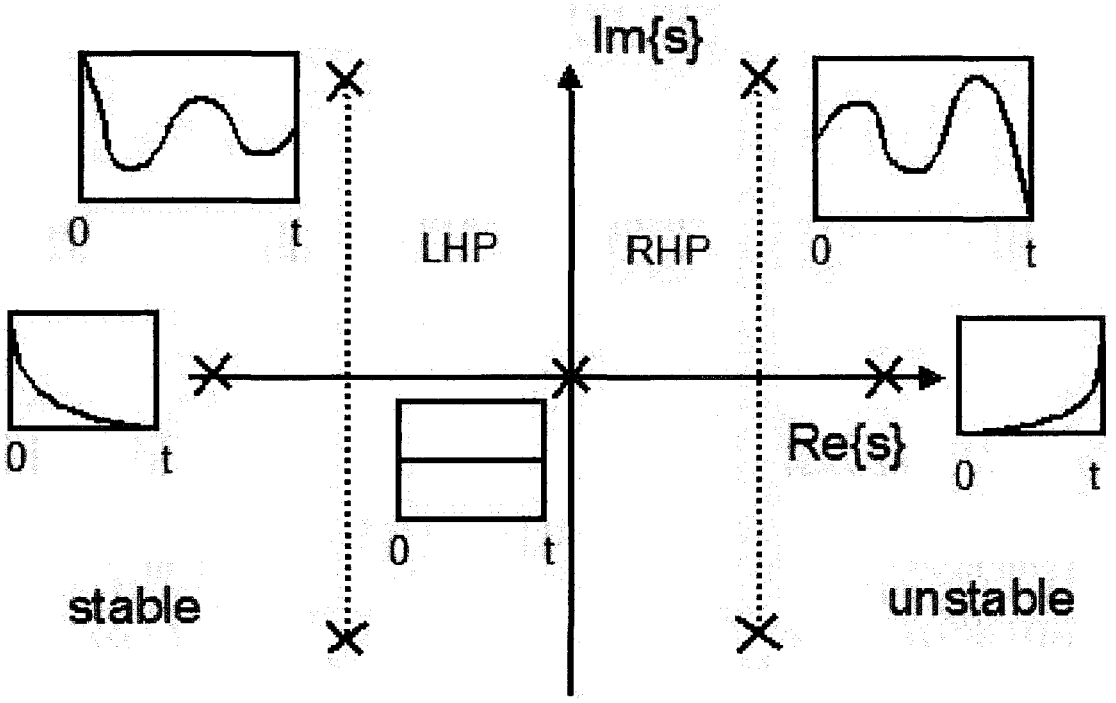


Fig. 2.1 The s-plane mapping of complex numbers is very useful for control design. The x-axis is the real portion of the s value and the y-axis is the imaginary value of the s value. The X values represent "poles" which are the same as eigenvalues of the A matrix. Poles with imaginary values always exist in symmetric pairs as indicated by the dotted lines. These pairs are complex conjugates of each other. Stable poles are on the Left Half Plane (LHP) and unstable poles are on the Right Half Plane (RHP).

2.2, one would get an oscillating signal with an exponentially growing or shrinking magnitude. Most control designers simply have the image of Fig. 2.1 permanently engraved in their minds and do not think about inverse laplace transforms during control design.

$$y(t)=\frac{1}{2j\pi}\int_{\nu-j\infty}^{\nu+j\infty}e^{st}y(s)ds \tag{2.2}$$

One should note that the inverse laplace transform typically is not terribly useful for control systems design. Control designers typically think in the *s*-domain, frequency domain, or state space representation. If one really needs to do an inverse laplace transform, one typically refers to a table which can be found in Ref. [2.1,2.2].

Another property of these representations involves the convolution integral of eq. 2.3. Basically, if one were to multiply two things $y_1(s)$ and $y_2(s)$ in the s-domain and then use the inverse laplace

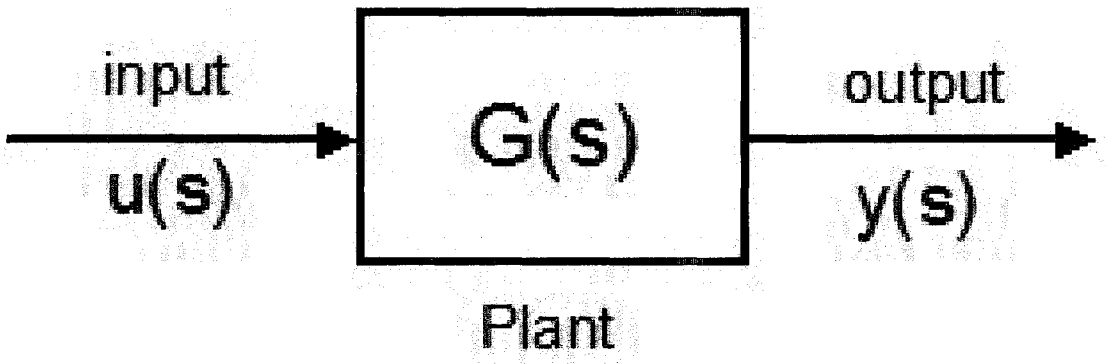


Fig. 2.2 The plant is the thing to be controlled by looking at outputs y and manipulating inputs u.

transform, the time domain representation would be the convolution integral $\int_0^\infty y_1(\tau)y_2(t-\tau)d\tau$.

$$y_1(s)y_2(s)\Leftrightarrow\int_0^\infty y_1(\tau)y_2(t-\tau)d\tau \tag{2.3}$$

This is particular useful for understanding transfer functions of plants, which have inputs, u and outputs y. In control design, the plant is the thing to be controlled. For aircraft control, the plant is the aircraft.

In the time domain, we can represent the outputs as in eq. 2.4. The *A* matrix is an [n x n] matrix, where n represents the order of the plant. The *B* matrix is an [n x m] matrix, where m represents the number of inputs. The *C* matrix is an [p x n] matrix, where p represents the number of outputs. The D matrix is a [p x m] matrix. There is a thing called a state vector, x, which has dimensions [n x 1]. The state vector represents the internal dynamics of the system. The state vector at time = 0 is denoted as $x(0)$.

$$y(t) = Ce^{At}x(0) + \int_0^t Ce^{A\tau}Bu(\tau)d\tau + Du(t) \tag{2.4}$$

The term e^{At} is a matrix exponential term and is shown in eq. 2.5. The exclamation points are factorial operations.

$$e^{At} \equiv I + At + A^2t^2/2! + A^3t^3/3! + A^4t^4/4! + \ldots \tag{2.5}$$

These *A,B,C,* and *D* matrices together form what is called the state space representation. These can be summarized below in eq. 2.6 where $\dot{x} \equiv dx/dt = sx$ and x is the state vector.

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{2.6}$$

A state space representation typically presents itself after one models a mechanical or electrical system using methods described in Ref. [2.7,2.8]. Sometimes, these models are non-linear, but most often they can be linearized around a particular region of x and u. This is the case for aircraft control and this area of x and u is called "trim".

In the s-domain, the transfer function can be represented below in eq. 2.7.

$$\begin{aligned} y(s) &= G(s)u(s) \\ G(s) &= C(sI - A)^{-1}B + D \end{aligned} \tag{2.7}$$

2.2 Poles

Poles of the plant are values of s that cause the transfer function to be undefined. From eq. 2.8, we can see that $G(s)$ will be undefined if $\boldsymbol{det}(sI - A) = 0$.

$$G(s) = C\frac{Adj(sI - A)}{\boldsymbol{det}(sI - A)}B + D \tag{2.8}$$

The expression $\boldsymbol{det}(sI - A) = 0$ is referred to as the characteristic polynomial, $\chi(A) = \boldsymbol{det}(sI - A)$, of the *A* matrix.

Poles of SISO and MIMO systems can be thought of as the *s* values in the denominator of the transfer function. For example, the following transfer function in eq. 2.9, has poles of -10, -1+*j* and -1-*j*.

$$G(s) = \frac{s(s+0.1)}{(s^2+2s+2)(s+10)} \tag{2.9}$$

A state space realization of the transfer function in eq. 2.9 is listed in eq. 2.10.

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -20 & -22 & -12 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad C = \begin{bmatrix} 0 & 0.1 & 1 \end{bmatrix} \tag{2.10}$$

Poles of SISO systems and MIMO systems can also be calculated from the eigenvalues of the *A* matrix, which are defined by the same expression $\chi(A) = \boldsymbol{det}(\lambda I - A)$, only the symbol, λ, is used instead of *s*.

An interesting property of the characteristic polynomial, $\chi(A) = \boldsymbol{det}(\lambda I - A)$, is that the *A* matrix always fulfills its own characteristic polynomial. This is known as the Cayley-Hamilton theorem. With the previous transfer function, one can show that the following is true.

$$A^3 + 12A^2 + 22A + 20I = 0 \tag{2.11}$$

2.3 Similarity Transforms

State space representations of a transfer function are not unique. Sometimes, it is helpful to rearrange the state space for different purposes. The idea is that one chooses a transition matrix, T, which is an [n x n] matrix that "rotates" the state vector so that the state space is in a more convenient form.

$$\begin{aligned} T\bar{x} &= x \\ T\dot{\bar{x}} &= AT\bar{x} + Bu \quad \dot{\bar{x}} = T^{-1}AT\bar{x} + T^{-1}Bu \\ y &= CT\bar{x} + Du \quad y = CT\bar{x} + Du \end{aligned} \tag{2.12}$$

We could rearrange the state space representation of 2.10 with the following T matrix.

$$T=\begin{bmatrix}3 & 1 & 1\\ -2 & 1 & -10\\ -2 & -4 & 100\end{bmatrix}\quad T^{-1}=\begin{bmatrix}0.1463 & -0.2537 & -0.0268\\ 0.5366 & 0.7366 & 0.0683\\ 0.0244 & 0.0244 & 0.0122\end{bmatrix} \tag{2.13}$$

We now see that a different state space for the same transfer function is listed below.

$$\dot{\bar{x}}=T^{-1}AT\bar{x}+T^{-1}Bu \quad y=CT\bar{x}$$
$$\dot{\bar{x}}=\begin{bmatrix}0 & 1 & 0\\ -2 & -2 & 0\\ 0 & 0 & -10\end{bmatrix}\bar{x}+\begin{bmatrix}-0.0268\\ 0.0683\\ 0.0122\end{bmatrix}u \quad y=\begin{bmatrix}-2.2 & -3.9 & 99\end{bmatrix}\bar{x} \tag{2.14}$$

Remarks:

1. State space representations of transfer functions are not unique. There are infinitely many state space representations of a single transfer function.
2. The transition matrix, T, is sometimes the eigenvectors of A. This is a modal decomposition where the transformed A matrix, $A_\lambda = T^{-1}AT$, is diagonal with individual eigenvalues along the diagonal. Some control texts have a strong emphasis on these decompositions [2.6].

2.4 Zeros of SISO Plants

Zeros of SISO systems are defined by the values of s for which the transfer function is equal to 0. These are like the roots of the numerator instead of the denominator. In the previous example of eq. 2.9, the zeros can be thought of as s = -0.1, and 0.

Suppose that a SISO transfer function can be represented by the following numerator portion, $G_N(s)$, and denominator, $G_D(s)$, portions in eq. 2.15.

$$G(s)=\frac{G_N(s)}{G_D(s)} \tag{2.15}$$

Then the zeros, s_o, of the SISO plant would be defined as the roots of the numerator portion as in eq. 2.16.

$$G_N(s_o)=0 \tag{2.16}$$

2.5 Stability with the Routh-Hurwitz Criterion

The Routh-Hurwitz criterion emerged at the turn of the 20th century. Routh tables become large for plants with many poles, so a 4th order example is given. Given the denominator portion of a plant or other transfer function, one can test for stability by initializing a Routh table as shown in table 2.1

1. Initialize the Routh Table with $G_D(s) = a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0$.

Table 2.1 Initial Routh Table

s^4	a_4	a_2	a_o
s^3	a_3	a_1	0
s^2			
s^1			
s^0			

2. Populate the Routh table using the pattern below. Note: a,b,c,d numbers are unrelated to the state space

Table 2.2 Populated Routh Table

s^4	a_4	a_2	a_o
s^3	a_3	a_1	0
s^2	$\frac{-det\left(\begin{bmatrix} a_4 & a_2 \\ a_3 & a_1 \end{bmatrix}\right)}{a_3} = b_1$	$\frac{-det\left(\begin{bmatrix} a_4 & a_0 \\ a_3 & 0 \end{bmatrix}\right)}{a_3} = b_2$	$\frac{-det\left(\begin{bmatrix} a_4 & 0 \\ a_3 & 0 \end{bmatrix}\right)}{a_3} = 0$
s^1	$\frac{-det\left(\begin{bmatrix} a_3 & a_1 \\ b_1 & b_2 \end{bmatrix}\right)}{b_1} = c_1$	$\frac{-det\left(\begin{bmatrix} a_3 & 0 \\ b_1 & 0 \end{bmatrix}\right)}{b_1} = 0$	0
s^0	$\frac{-det\left(\begin{bmatrix} b_1 & b_2 \\ c_1 & 0 \end{bmatrix}\right)}{c_1} = d_1$	$\frac{-det\left(\begin{bmatrix} b_1 & 0 \\ c_1 & 0 \end{bmatrix}\right)}{c_1} = 0$	

3. Interpret the Routh table. The number of sign changes in the first column of the Routh table are equal to the number of unstable poles or roots of $G_D(s)$.

4. Special cases exist when entries of the first column of the Routh table become 0. In this case, one sets a number like b_1 equal to a very small number and proceeds.

2.3.1 Application of Routh Table to Regulator Design

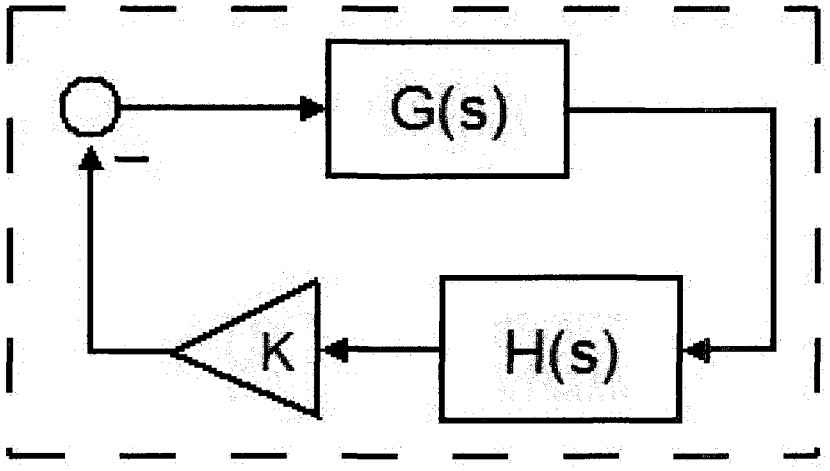


Fig. 2.3 A regulator system "places" poles of the closed loop in desireable locations

Suppose we wanted to design a regulator, $KH(s)$, for a plant in the configuration of Fig. 2.3.

The total regulated system, $G^{reg}(s)$, is shown below with $H(s)=H_N(s)/H_D(s)$ in eq. 2.17.

$$G^{reg}(s)=\frac{G_N(s)H_D(s)}{G_D(s)H_D(s)+KG_N(s)H_N(s)} \tag{2.17}$$

Suppose that $G(s)=\frac{-s(s-10)}{(s^2+0.1s+1)(s+10)}$ and $H(s)=1$ was a low pass filter. Then we can express the denominator of $G^{reg}(s)$ as follows below in eq. 2.18.

$$G_D^{reg}(s)=(s^2+0.1s+1)(s+10)-Ks(s-10)=s^3+(10.1-K)s^2+(2+10K)s+10 \tag{2.18}$$

We now initialize the Routh table as follows below.

Table 2.3 Initial Routh Table for regulator system

s^3	1	2+10K	0
s^2	10.1-K	10	0
s^1			
s^0			

Following the Routh pattern, we populate the routh array as in Table 2.4. We can see that if K = 1 or 2, then the first column of the routh array remains completely positive, which means that K = 1 or 2 would be a stabilizing feedback gain for the system. However, if we chose a really large gain, like K = 20, there would 2 sign reversals in the first column in the Routh array. This would mean that there would be 2 poles in the RHP. With this in mind, we simply choose a gain of K = 2.

Table 2.4 Populated Routh Table

s^3	1	2+10K	0
s^2	10.1-K	10	0
s^1	$\frac{-10+(10.1-K)(2+10K)}{10.1-K}$	0	
s^0	10		

2.6 Relative Degree

The relative degree of a SISO plant is the order of polynomial in the denominator minus the order of the polynomial in the numerator.

$$G(s)=\frac{c_m s^m + c_{m-1}s^{m-1} + \ldots + c_o}{s^n + a_{n-1}s^{n-1} + \ldots + a_o}, \quad \text{rel. deg.} = m - n \tag{2.19}$$

Relative degree of MIMO plants can sometimes be more difficult to ascertain. One can think of the interactor matrix, introduced in Chapter 6, as a measure of MIMO relative degree.

Sometimes MIMO plants have uniform relative degree. This means that each of the individual transfer functions, $G(s)_{ij}$, have the same number of poles and zeros.

2.4.1 Strictly Proper Plants

If the relative degree of the plant is 1 or greater, then the plant is said to be strictly proper. State space representations of strictly proper plants have $D = 0$.

2.4.2 Proper Plants

If the relative degree of the plant is 0, then the plant is said to be proper. State space representations of strictly proper plants have a non-zero D matrix.

2.4.3 Biproper Plants

If the relative degree of the plant is 0 and the plant is square and invertible, then the plant is said to be biproper. State space representations of strictly proper plants have an invertible D matrix, $det(D) \neq 0$.

2.4.4 Improper Transfer Functions

Improper transfer functions do not have a state space representation. Physical processes are never improper transfer functions. The only improper transfer function used in this work is the interactor matrix, discussed in Chapter 6. The interactor matrix is an artificial thing that differentiates until the plant is biproper or proper at least. One might think of the interactor matrix as a thing that shifts velocity or position outputs to acceleration.

2.7 Classification of Plants with Pole and Zero Locations

The location of the plant's poles and zeros will largely determine what is possible to achieve with feedback control. It is important to understand these classification schemes.

2.5.1 Minimum-Phase and Stable Plants

If all real parts of each pole and zero have negative values (i.e. located in the Left-Half Plane (LHP)), then the plant is said to be stable and minimum phase. The plant of eq. 2.9 is an example of a stable and minimum phase plant. It is easy to design a controller for a minimum phase and stable plants.

2.5.2 Non-minimum-Phase and Stable Plants

If all real parts of each pole have negative values (i.e. located in the Left-Half Plane (LHP)), then the plant is said to be stable. If even one zero has a positive real part (i.e. located in the Right-Half-Plane (RHP)), then the plant is non-minimum phase. An equivalent statement is that a non-minimum phase plant will have at least one unstable zero. A plant with a time delay is also a non-minimum phase plant.

The most significant thing concerning non-minimum phase plants is that there are restrictions on what is achievable with a non-minimum phase plant. One cannot use large amounts of feedback at frequencies near unstable zeros. An example of a stable and non-minimum phase plant is shown below.

$$G(s) = \frac{(s-10)}{(s+0.5)(s+1)}$$

2.5.3 Non-minimum-Phase and Unstable Plants

If a plant has at least one unstable pole (i.e. a pole in the RHP) then the plant is said to be unstable. If a plant has at least one unstable zero (i.e. a zero in the RHP) then the plant is said to be non-minimum phase. Stabilizing an unstable pole with feedback requires high gain. However, one must use small gain at frequencies near unstable zeros. Therefore, designing controllers for unstable and non-minimum phase plants is difficult and sometimes almost impossible when unstable poles and zeros are close to each other in frequency.

Ideally, one would like unstable poles and zeros to be separated by a factor of 6 to 10 in frequency. The plant below is an example of a plant for which it is reasonable to design a controller.

$$G(s)=\frac{-(s-20)}{(s-1)(s+20)}$$

One would have a more difficult time designing a controller for the plant below.

$$G(s)=\frac{-(s-2)}{(s-1)(s+2)}$$

2.5.4 Minimum-Phase and Unstable Plants

If a plant has at least one unstable pole (i.e. a pole in the RHP) then the plant is said to be unstable. If the plant is minimum phase, then all of the zeros are stable. It is easy to design controllers for these plants, because one can simply use high gain without penalties. An example is shown below.

$$G(s)=\frac{1}{(s-1)}$$

Remarks:

1. One might also include relative degree for classifying plants. For example, one might find a control paper which only discusses control design for stable and non-minimum phase plants of uniform relative degree 1.
2. MIMO plants can be additionally classified by the number of inputs and outputs. A plant with more outputs than inputs, $p > m$, can be referred to as a tall plant. A plant with more inputs than outputs, $p < m$, can be called a fat plant. A plant is said to be square if it has an equal number of inputs and outputs, $p = m$.

3. Almost any physical plant is non-minimum phase at some very high frequency due to time delay. Many minimum phase plant models only represent plant dynamics at lower frequencies. One should design controllers with this in mind.
4. Ideally, one always seeks to build plants which are stable and minimum phase because they are easy to design controllers for them. For MIMO plants, it is good to have an equal number of inputs and outputs. Almost any feedback control technique can work for square and minimum phase plants.

2.8 State Observability

The notion of state observability determines whether or not all of the state dynamics, x, can be "seen" from the outputs, y. One can form the observability matrix, $Obsv$, from the C and the A matrix given that the order of the system is n.

$$Obsv = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix} \tag{2.20}$$

If the rank of the observability matrix, $Obsv$, is less than n, then the system is not fully observable. Otherwise, the system is fully observable.

$$if \quad rank[Obsv] = n \quad then \quad plant \quad is \quad completely \quad observable$$

Alternatively, one can use the Popov-Bellman-Hautus (PBH) test to see whether or not a specific given mode/eigenvalue/pole of the plant is unobservable.

$$\text{if} \quad rank\begin{bmatrix} \lambda_i I - A \\ C \end{bmatrix} < n$$

$$\text{then} \quad \lambda_i \quad \text{is} \quad \text{unobservable}$$

2.9 State Controllability

The notion of state controllability determines whether or not all of the states, x, can have their dynamics modified by feedback. This assumes that all of the states would present themselves for

feedback. One can form the controllability matrix, $Cntrb$, from the A and the B matrix given that the order of the system is n.

$$Cntrb = \begin{bmatrix} B & AB & A^2B & \dots & A^{n-1}B \end{bmatrix} \tag{2.21}$$

If the rank of the controllability matrix, $Cntrb$, is less than n, then the system is not fully controllable. Otherwise, the system is fully controllable.

$$if \quad rank[Cntrb] = n \quad then \quad plant \quad is \quad completely \quad controllable$$

Alternatively, one can use the Popov-Bellman-Hautus (PBH) test to see whether or not a specific given mode/eigenvalue/pole of the plant is uncontrollable.

$$\text{if} \quad rank[\lambda_i I - A \quad B] < n$$
$$\text{then} \quad \lambda_i \quad \text{is} \quad \text{uncontrollable}$$

2.10 Functional Controllability

The idea of functional controllability is exclusively related to MIMO plants. Basically, if a plant is invertible, then it is functionally controllable with the selected outputs. Most control texts do not discuss the notion of functional controllability. There is also no conventionally accepted term for this concept. Functional controllability is simply called system invertibility in Ref. [2.6]. The term functional controllability is coined in Ref. [2.9], however, it is not completely defined.

1. A plant which has more outputs than inputs, p > m, is not functionally controllable.
2. A plant which has at least as many inputs as outputs, $p \leq m$, *might* be functionally controllable.
3. A plant must be invertible in order for the plant to be functionally controllable. If one cannot find an interactor matrix for the plant, discussed in chapter 6, no inverse exists and the plant is not functionally controllable.
4. For a square plant, a plant whose determinant is zero in the entire *s*-domain is functionally uncontrollable. If $det(G(s)) = 0 \quad \forall \quad s$, then the plant is functionally uncontrollable.
5. Functional controllability is really only relevant to tracking systems. If one only wants a regulator design, then one does not need to consider functional controllability.

2.11 Kalman Decomposition

The Kalman decomposition shows hidden modes. A hidden mode could be both unobservable and uncontrollable. A hidden mode could be unobservable and controllable or also uncontrollable and observable. The Kalman decomposition is named after its creator, Rudolf Kalman. The Kalman decomposition is also discussed in Refs. [2.4,2.5].

After calculating the controllability matrix and the observability matrix, one can calculate the Kalman decomposition. First, one must form a basis for the range of the controllability matrix. Next, one must calculate the null space of the observability matrix.

1. Calculate the controllability matrix.
2. Calculate the observability matrix.
3. Form a basis for the controllability matrix such that $Range[Cntrb] = v^c_{\bullet,1}c_1 + v^c_{\bullet,2}c_2 + \ldots + v^c_{\bullet,r}c_r$, and $c_1, c_2, \ldots, c_r$ are arbitrary constants.
4. Form a basis for the null space of the observability matrix, such that
$$Null[Obsv] = \begin{bmatrix} v^o_{\bullet,1} & v^o_{\bullet,2} & \ldots & v^o_{\bullet,b} \end{bmatrix}$$
$$Obsv\begin{bmatrix} v^o_{\bullet,1} & v^o_{\bullet,2} & \ldots & v^o_{\bullet,b} \end{bmatrix} = 0$$
5. Find the intersection of two vector spaces, $CU = Range[Cntrb] \cap Null[Obsrv]$. Form the following matrix from the previous vectors, $W_{\bullet,\bullet,1} = \begin{bmatrix} v^c_{\bullet,1} & v^c_{\bullet,2} & \ldots & v^c_{\bullet,r} & v^o_{\bullet,1} \end{bmatrix}$. If $rank[W_{\bullet,\bullet,1}] > rank\begin{bmatrix} v^c_{\bullet,1} & v^c_{\bullet,2} & \ldots & v^c_{\bullet,r} \end{bmatrix}$, then $v^o_{\bullet,1}$ is not a member of CU. Otherwise, $v^o_{\bullet,1}$ is a member of CU. Repeat this with $W_{\bullet,\bullet,2} = \begin{bmatrix} v^c_{\bullet,1} & v^c_{\bullet,2} & \ldots & v^c_{\bullet,b} & v^o_{\bullet,2} \end{bmatrix}$. If $rank[W_{\bullet,\bullet,2}] > rank\begin{bmatrix} v^c_{\bullet,1} & v^c_{\bullet,2} & \ldots & v^c_{\bullet,b} \end{bmatrix}$, then $v^o_{\bullet,2}$ is not a member of CU. Otherwise, $v^o_{\bullet,2}$ is a member of CU. Continue this process for all vectors in the null space of the observability matrix. Let ncu be the total number of vectors that form a basis for CU. Let $Range[CU] = v^{cu}_{\bullet,1}c_1 + v^{cu}_{\bullet,2}c_2 + \ldots + v^{cu}_{\bullet,ncu}c_{ncu}$, and $c_1, c_2, \ldots, c_{ncu}$.
6. Use the first $1 \ldots (r - ncu)$ vectors to form $V_{\bullet,\bullet,1} = \begin{bmatrix} v^c_{\bullet,1} & v^c_{\bullet,2} & \ldots & v^c_{\bullet,(r-ncu)} \end{bmatrix}$.

7. Use the first 1... *ncu* vectors to form $V_{\bullet,\bullet,2} = \begin{bmatrix} v_{\bullet,1}^{cu} & v_{\bullet,2}^{cu} & \cdots & v_{\bullet,ncu}^{cu} \end{bmatrix}$

8. Choose $V_{\bullet,\bullet,3}$ and $V_{\bullet,\bullet,4}$ such that $span\left(V_{\bullet,\bullet,1} \oplus V_{\bullet,\bullet,2} \oplus V_{\bullet,\bullet,3} \oplus V_{\bullet,\bullet,4}\right) = \Re^n$.

9. The transition matrix $T = \begin{bmatrix} V_{\bullet,\bullet,1} & V_{\bullet,\bullet,2} & V_{\bullet,\bullet,3} & V_{\bullet,\bullet,4} \end{bmatrix}$ forms the Kalman decomposition

where $\overline{A} = T^{-1}AT = \begin{bmatrix} \overline{A}_{11} & 0 & \overline{A}_{13} & 0 \\ \overline{A}_{21} & \overline{A}_{22} & \overline{A}_{23} & \overline{A}_{24} \\ 0 & 0 & \overline{A}_{33} & 0 \\ 0 & 0 & \overline{A}_{43} & \overline{A}_{44} \end{bmatrix}$ $\overline{B} = T^{-1}B = \begin{bmatrix} \overline{B}_1 \\ \overline{B}_2 \\ 0 \\ 0 \end{bmatrix}$ $\overline{C} = CT = \begin{bmatrix} \overline{C}_1 & 0 & \overline{C}_2 & 0 \end{bmatrix}$

The important thing to note is that the eigenvalues corresponding to $\overline{A}_{22}$ are the unobservable, yet controllable modes. The eigenvalues corresponding to $\overline{A}_{33}$ are the uncontrollable, yet observable modes. The eigenvalues corresponding to $\overline{A}_{44}$ are the uncontrollable and unobservable modes. All of the modes in $\overline{A}_{22}$, $\overline{A}_{33}$, and $\overline{A}_{44}$ are called hidden modes.

Example 2.1

$$A = \begin{bmatrix} 0 & 2 & -2 \\ -1 & -3 & 1 \\ 1 & 1 & -3 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \quad C = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$$

Step 1- $Cntrb = \begin{bmatrix} B & AB & A^2B \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 4 \\ 1 & -2 & 4 \end{bmatrix}$

Step 2- $Obsv = \begin{bmatrix} C \\ CA \\ CA^2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & -3 \\ -4 & -4 & 8 \end{bmatrix}$

Step 3- $rank\left[Cntrb\right] = 1$, so we just take the first column to be $v_{\bullet,1}^{c} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$, $r = 1$

Step 4- $rank[Obsv]=2$, and by inspection, we see that $\begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & -3 \\ -4 & -4 & 8 \end{bmatrix}\begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}=0$, therefore the

vector $v^{o}_{\bullet,1}=\begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}$, $b=1$

Step 5- $W_{\bullet,\bullet,1}=\begin{bmatrix} 0 & 1 \\ 1 & -1 \\ 1 & 0 \end{bmatrix}$, $rank[W_{\bullet,\bullet,1}] > rank\begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$ because 2 > 1, therefore $CU=\varnothing$, $ncu=0$

Step 6,7,8 - $V_{\bullet,\bullet,1}=\begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$, $V_{\bullet,\bullet,3}=\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$, $V_{\bullet,\bullet,4}=\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$, $V_{\bullet,\bullet,2}=\varnothing$

Example 2.2

$$A=\begin{bmatrix} -1 & 0 & -2 & -2 \\ 0 & -3 & 0 & 0 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & -4 \end{bmatrix} \quad B=\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad C=\begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}$$

Step 1- $Cntrb=\begin{bmatrix} B & AB & A^2B & A^3B \end{bmatrix}=\begin{bmatrix} 1 & -1 & 1 & -1 \\ 1 & -3 & 9 & -27 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$

Step 2- $Obsv=\begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{bmatrix}=\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -3 & 0 & 0 \\ 0 & 9 & 0 & 0 \\ 0 & -27 & 0 & 0 \end{bmatrix}$

Step 3- $rank[Cntrb]=2$, so we just take the first two columns to be $v^{c}_{\bullet,1}=\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$, $v^{c}_{\bullet,2}=\begin{bmatrix} -1 \\ -3 \\ 0 \\ 0 \end{bmatrix}$, $r=2$

Step 4- $rank[Obsv]=1$, and by inspection, we see that simply taking $v^o_{\bullet,1}=\begin{bmatrix}1\\0\\0\\0\end{bmatrix}$, $v^o_{\bullet,2}=\begin{bmatrix}0\\0\\1\\0\end{bmatrix}$ and

$v^o_{\bullet,3}=\begin{bmatrix}0\\0\\0\\1\end{bmatrix}$, we form a basis for the null space of the observability matrix.

Step 5- $W_{\bullet,\bullet,1}=\begin{bmatrix}1&-1&1\\1&-3&0\\0&0&0\\0&0&0\end{bmatrix}$, $rank[W_{\bullet,\bullet,1}]=rank\begin{bmatrix}1&-1\\1&-3\\0&0\\0&0\end{bmatrix}$, therefore $v^{cu}_{\bullet,1}=\begin{bmatrix}1\\0\\0\\0\end{bmatrix}$.

$W_{\bullet,\bullet,2}=\begin{bmatrix}1&-1&0\\1&-3&0\\0&0&1\\0&0&0\end{bmatrix}$, $rank[W_{\bullet,\bullet,2}]>rank\begin{bmatrix}1&-1\\1&-3\\0&0\\0&0\end{bmatrix}$, therefore $v^o_{\bullet,2}\notin CU$,

$W_{\bullet,\bullet,3}=\begin{bmatrix}1&-1&0\\1&-3&0\\0&0&0\\0&0&1\end{bmatrix}$, $rank[W_{\bullet,\bullet,3}]>rank\begin{bmatrix}1&-1\\1&-3\\0&0\\0&0\end{bmatrix}$, therefore $v^o_{\bullet,3}\notin CU$, $ncu=1$

Step 6,7,8 - $V_{\bullet,\bullet,1}=\begin{bmatrix}1\\1\\0\\0\end{bmatrix}$, $V_{\bullet,\bullet,2}=\begin{bmatrix}1\\0\\0\\0\end{bmatrix}$, $V_{\bullet,\bullet,3}=\begin{bmatrix}0\\0\\1\\0\end{bmatrix}$, $V_{\bullet,\bullet,4}=\begin{bmatrix}0\\0\\0\\1\end{bmatrix}$

2.12 Minimal Realizations

The idea of minimality is that there are no extra or unnecessary states. There should be no hidden modes. After the Kalman decomposition is done, one can use the following state space representation to express a minimal realization of the transfer function.

$$G(s)=C(sI-A)^{-1}B \Rightarrow kalman\quad decomposition \Rightarrow G(s)=\overline{C}_1(sI-\overline{A}_{11})^{-1}\overline{B}_1$$

Now the transfer function does not have any extra states. The MATLAB command "minreal( )" achieves this and can optionally work with a relative tolerance if the syntax "mineral(sys,TOL)" is used. The minreal( ) command is used extensively throughout this work to make pole-zero cancellations and help make transfer functions more manageable.

2.13 SISO Control Design with the Root Locus

The root locus is a graphical plot that shows where the poles of the closed loop travel as the gain, K, is increased. Closed loop poles travel from their open-loop locations towards open-loop zero locations as the gain is increased.

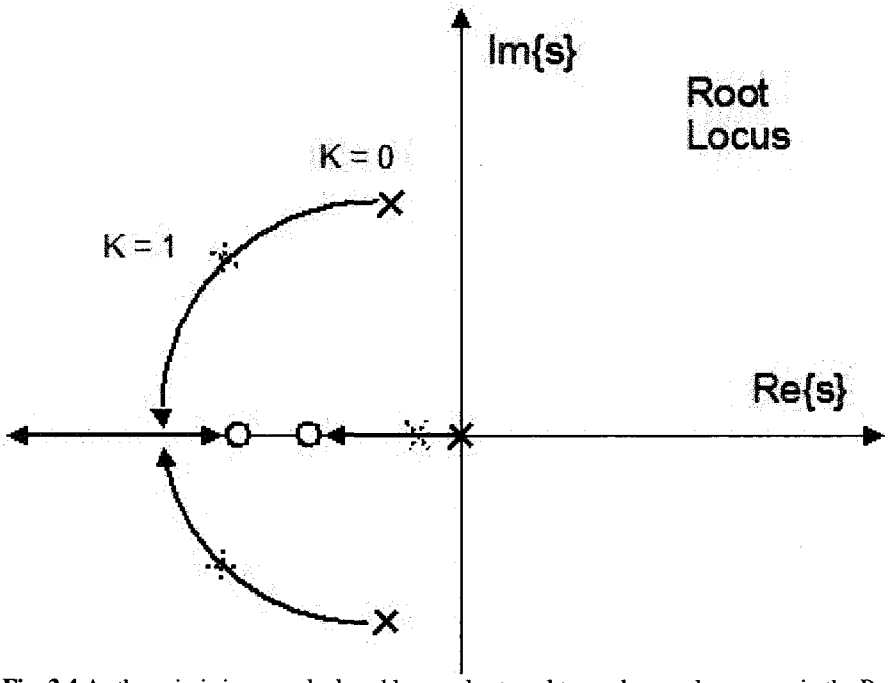


Fig. 2.4 As the gain is increased, closed loop poles travel towards open loop zeros in the Root Locus

2.11.1 Drawing the Root Locus for Plants with a Positive Gain

This is for plants where the gain of $KG(s)H(s)$ must be positive. As an example, this root locus drawing technique will work for $G(s)=\frac{1}{s+1}$, $H(s)=1$, $0 \le K \le \infty$ or $G(s)=\frac{-1}{s+1}$, $H(s)=1$, $-\infty \le K \le 0$. In the regulated system of Fig. 2.3, the root locus is defined by the values of s that satisfy eq. 2.22 below.

$$1+KG(s)H(s)=0 \tag{2.22}$$

This means that we need to find values of s that make $KG(s)H(s) = -1$. Performing an exhaustive search of the s plane for points that satisfy this criterion is difficult. One can obtain a fairly decent sketch of the root locus using the steps below or use the MATLAB "rlocus( )" command.

1. If one were to sum all the angles made with poles and subtract all angles made with zeros and the total was an odd multiple of 180 degrees, then that point would be on the root locus. If point P, shown in Fig. 2.5, were on the root locus, then $A_1 + A_2 - A_3 = (2k+1)180^\circ$, for $k = 0, \mp 1, \mp 2, \ldots$. Suppose that $A_1 = 135^\circ$, $A_2 = 90^\circ$ and $A_3 = 45^\circ$, then $A_1 + A_2 - A_3 = 180^\circ$ and the point P would be on the root locus.
2. The gain corresponding to the point on the root locus can be calculated from the following formula $K = \dfrac{\prod pole \quad lengths}{\prod zero \quad lengths}$. From Fig. 2.5, we could calculate the gain as $\dfrac{L_1 L_2}{L_3}$.
3. The root locus is always symmetric about the real axis.
4. Real axis segments of the root locus are to the *left* of odd numbered real axis poles and/or real axis zeros.

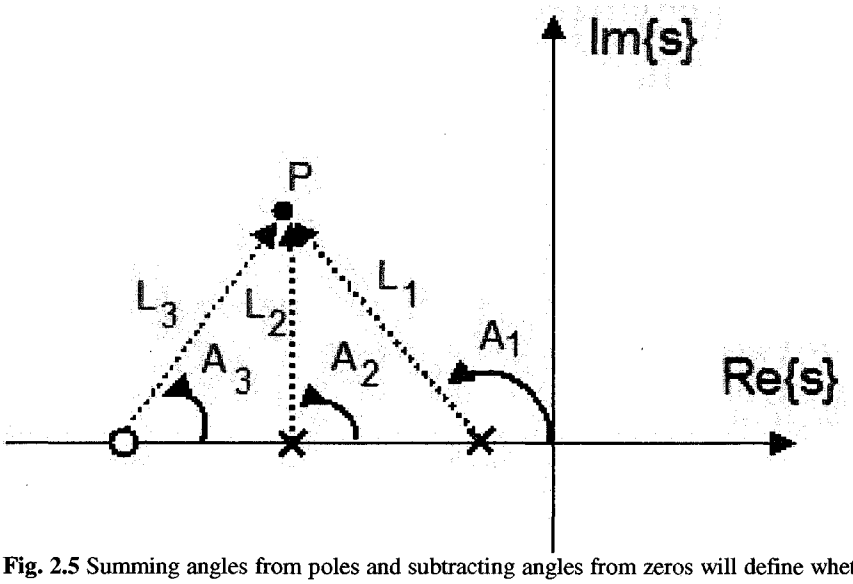


Fig. 2.5 Summing angles from poles and subtracting angles from zeros will define whether or not the point P is on the root locus. The sum of angles from poles minus angles from zeros should sum to an odd multiple of 180 degrees.

5. The root locus has asymptotes that intersect at a location σ_{asy} on the real axis and has angles, A_{asy} determined by the following equation, $\sigma_{asy} = \frac{\sum finite\ \ poles - \sum finite\ \ zeros}{\#\ finite\ \ poles - \#\ finite\ \ zeros}$, and $A_{asy} = \frac{(2k+1)180^{\circ}}{\#\ finite\ \ poles - \#\ finite\ \ zeros}$ for $k = 0, \mp 1, \mp 2, \ldots$.

6. The root locus will "break away" from the real axis in between a pair of real axis poles with no zeros between them. The root locus will "break in" real axis segments near real axis zeros. There are other combinations of break away and break in points, though this will describe where to find most of them. Break away and break in points can be solved mathematically using: $\sum_{k=1}^{\#zeros} \frac{1}{\sigma_{br} - s_{o,k}} = \sum_{k=1}^{\#poles} \frac{1}{\sigma_{br} - s_{p,k}}$, where $s_{o,k}$ represents the k[th] zero, $s_{p,k}$ represents the k[th] pole, and σ_{br} represents a break away or break-in point.

Example 2.3

$$G(s)=\frac{-1}{(s^2+2s+2)} \quad H(s)=-s/(s+6), \; G(s)H(s)=\frac{s}{(s^2+2s+2)(s+6)}$$

We solve for the location and angle of the asymptotes of the root locus.

$$\sigma_{asy}=\frac{(-1-j)+(-1+j)+(-6)-0}{3-1}=-4 \text{ and } A_{asy}=\frac{\pm 180^\circ}{2}=\pm 90^\circ.$$

We also solve for the break in and break away points of the root locus.

$$\frac{1}{\sigma_{br}}=\frac{1}{\sigma_{br}+1-j}+\frac{1}{\sigma_{br}+1+j}+\frac{1}{\sigma_{br}+6} \text{ which reduces to the following cubic expression}$$

$2\sigma_{br}^3+8\sigma_{br}^2-12=0$ and roots of $\sigma_{br}=-3.51,-1.57,1.08$.

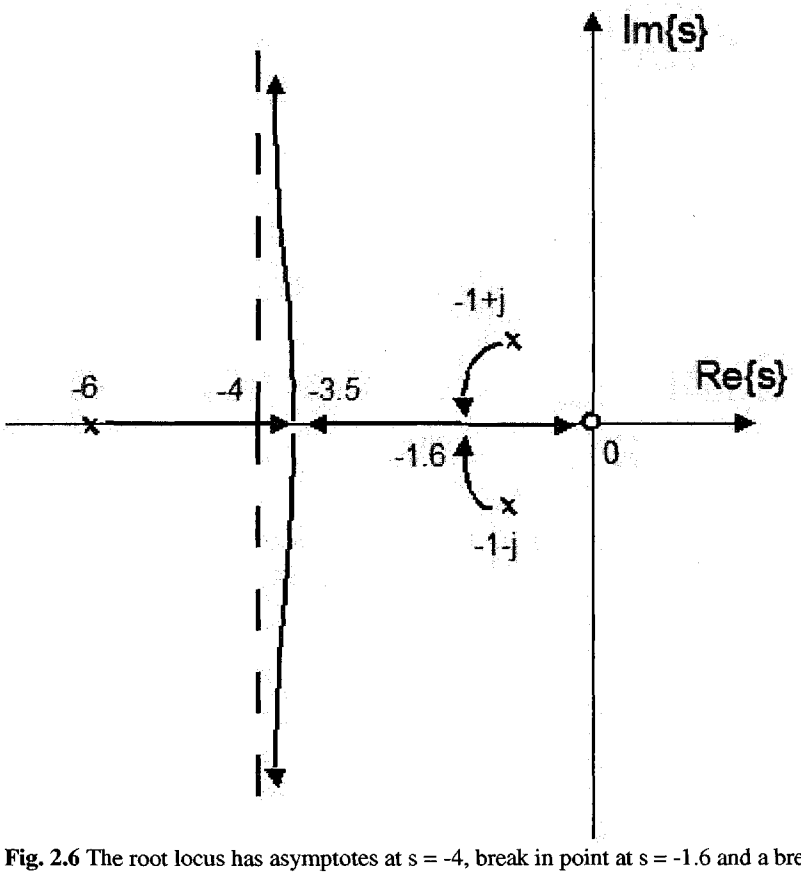


Fig. 2.6 The root locus has asymptotes at s = -4, break in point at s = -1.6 and a break away point at s = -3.5.

Now suppose that we want to choose a gain that places poles near the first break in point of approximately -1.6. The poles at -1+j and 1-j are both approximately a length of 1.2 away from the break in point at -1.6. The pole at -6 is a distance of 4.2 away from the break in point at -1.6. The zero at 0 is a

distance of 1.6 away from the break in point at -1.6. Therefore, the gain should be

$K = \frac{(1.2)(1.2)(4.2)}{(1.6)} = 3.78$. We now have all the poles of the closed loop system on or very near the real axis.

2.11.2 Drawing the Root Locus for Plants with a Negative Gain

This is for plants where the gain of $KG(s)H(s)$ must be positive. As an example, this root locus drawing technique will work for $G(s) = \frac{s-6}{s+1}$, $H(s) = -1$, $0 \le K \le \infty$ or $G(s) = \frac{(s-6)}{s+1}$, $H(s) = 1$, $-\infty \le K \le 0$. In the regulated system of Fig. 2.3, the root locus is defined by the values of s that satisfy eq. 2.23 below.

$$1 - KG(s)H(s) = 0 \tag{2.23}$$

This means that we need to find values of s that make $KG(s)H(s) = 1$. Things are very similar to the positive gain case shown earlier, similarities and differences are listed below.

1. Points on the root locus have anglular contributions which sum to a multiple of 360 degrees, instead of 180 degrees. From Fig. 2.5, point P would be on the root locus if

   $A_1 + A_2 - A_3 = 0°, 360°, -360°$.

2. Asymptote locations are the same. Asymptote directions are calculated by

   $$A_{asy} = \frac{360° k}{\#\ finite\ \ poles - \#\ finite\ \ zeros}, \text{ for } k = 0, \mp 1, \mp 2, \mp 3, \ldots.$$

3. Real axis segments of the root locus are to the *right* of odd numbered real axis poles and/or real axis zeros.

4. Breakaway and break in points are calculated in the same way as was previously done

Example 2.4

$$G(s)=\frac{-(s-10)}{s(s+10)} \quad H(s)=1,\ G(s)H(s)=\frac{-(s-10)}{s(s+10)}$$

We solve for the location and angle of the asymptotes of the root locus.

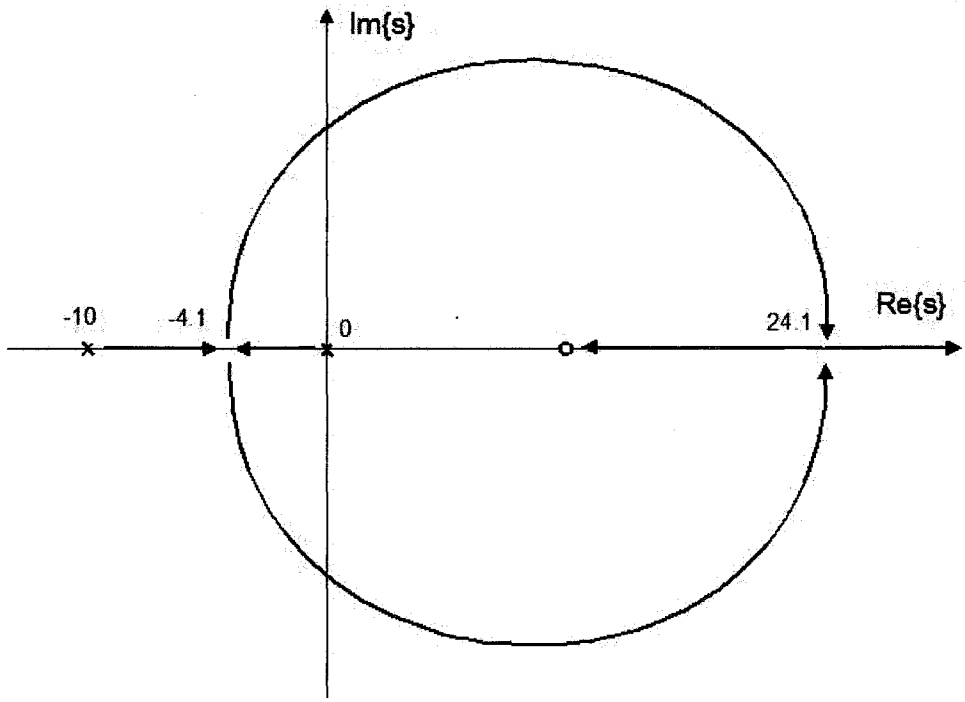


Fig. 2.7 The root locus breaks in at s = 24.1 and a breaks away at s = -4.1. Notice that with large gain, this system will be destabilized.

$$\sigma_{asy}=\frac{(-10)+(0)-(+10)}{2-1}=-20 \text{ and } A_{asy}=\frac{\pm 360^\circ}{1}=\pm 360^\circ.$$

We also solve for the break in and break away points of the root locus. $\frac{1}{\sigma_{br}}+\frac{1}{\sigma_{br}+10}=\frac{1}{\sigma_{br}-10}$

which reduces to the following cubic expression $\sigma_{br}^2-20\sigma_{br}-100=0$ and roots of

$\sigma_{br}=24.14,-4.14.$

Suppose that we want to place the closed loop poles near the 4.1 spot. Then, we notice that

$K=\frac{(4.1)(5.9)}{14.1}=1.71$ places the closed loop poles near the break away point. Alternatively, we could

choose a gain of $K=1$ and place closed loop poles near -1.

2.14 Approximate Bode Plots

Suppose that a plant has a transfer function $G(s)$ which then is transferred into the frequency domain, $G(j\omega) = |Mag|e^{j(Phase)}$. We then plot the magnitude, $|Mag|$ and phase angle versus frequency.

2.14.1 Real Axis Poles

Stable Poles: Stable poles of the form $\frac{a}{s+a}$ will exhibit a low frequency magnitude of approximately

1. This is followed by a steady decrease in the phase angle of -45 deg./dec. (degrees per decade) after one

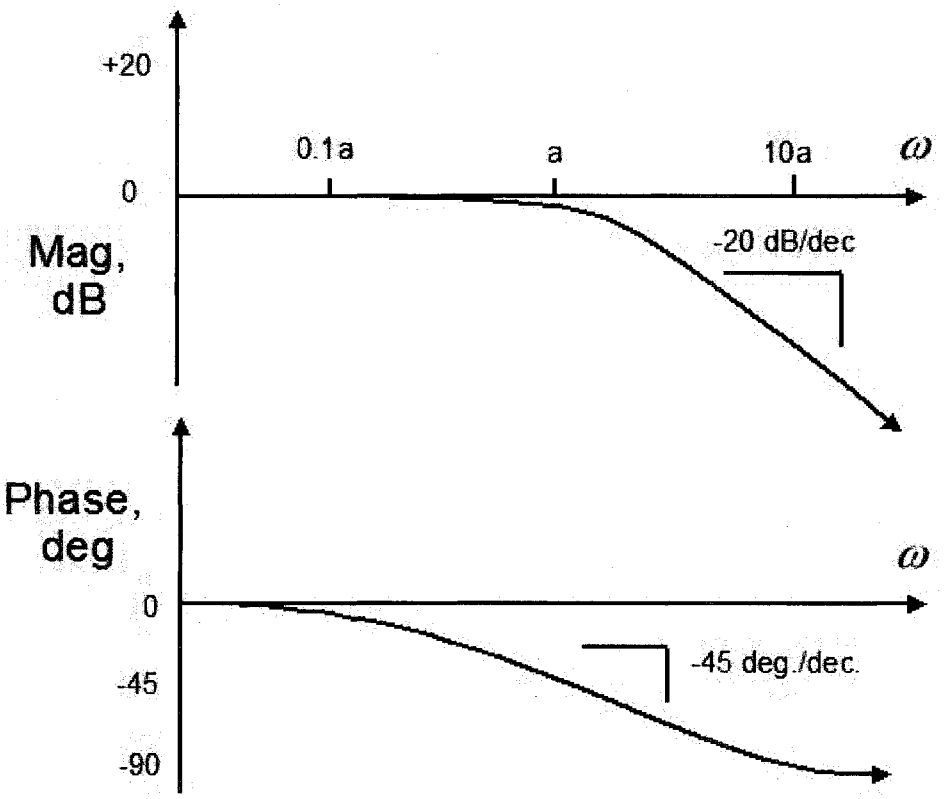


Fig. 2.8. Bode plot of a stable and real axis pole

tenth of the frequency of the pole. At $\omega \approx a$, the magnitude will be approximately -3dB and will sharply decrease at -20 dB/dec. (decibels per decade) thereafter.

Unstable Poles: Unstable poles of the form $\frac{a}{s-a}$ will exhibit a low frequency magnitude of approximately -1, which means that the phase must be -180 or +180. This is followed by a steady increase in the phase angle of -45 deg./dec. (degrees per decade) after one tenth of the frequency of the pole. At $\omega \approx a$, the magnitude will be approximately -3dB and will sharply decrease at -20 dB/dec. (decibels per decade) thereafter.

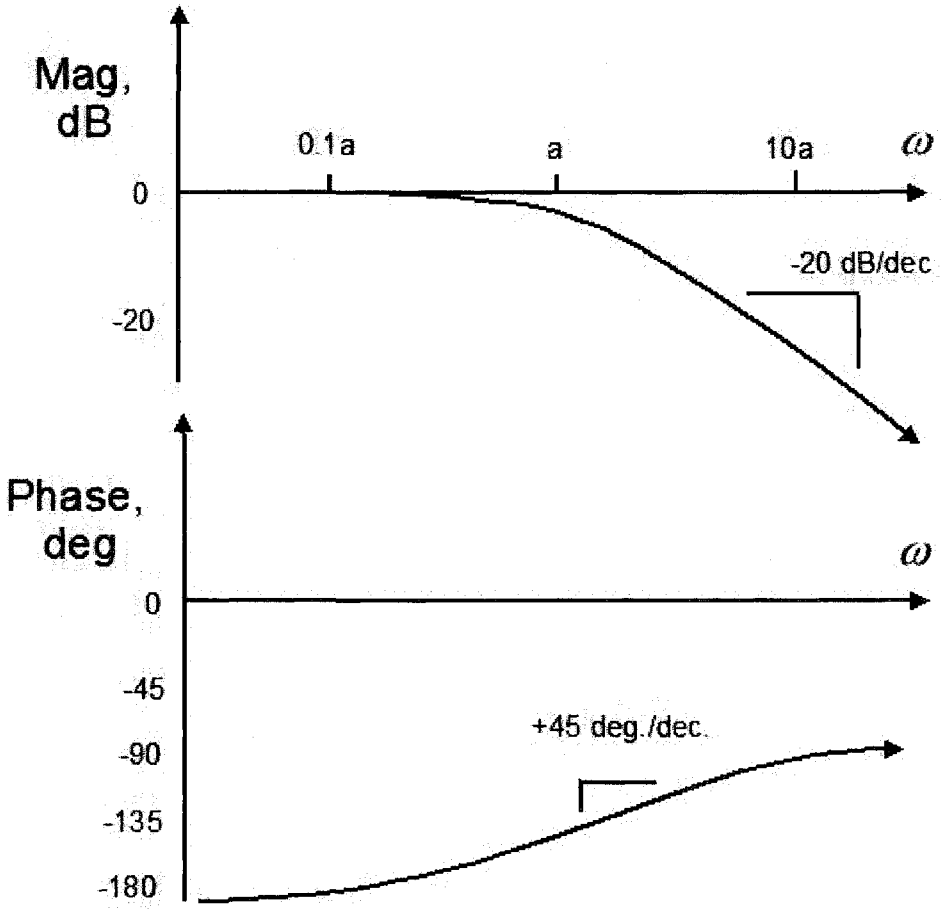


Fig. 2.9. Bode plot of an unstable real axis pole

2.14.2 Real Axis Zeros

Stable Zeros: Stable zeros of the form $\frac{(s+a)}{a}$ will exhibit a low frequency magnitude of approximately 1. This is followed by a steady increase in the phase angle of +45 deg./dec. (degrees per decade) after one tenth of the frequency of the zero. At $\omega \approx a$, the magnitude will be approximately +3dB and will sharply increase at +20 dB/dec. (decibels per decade) thereafter.

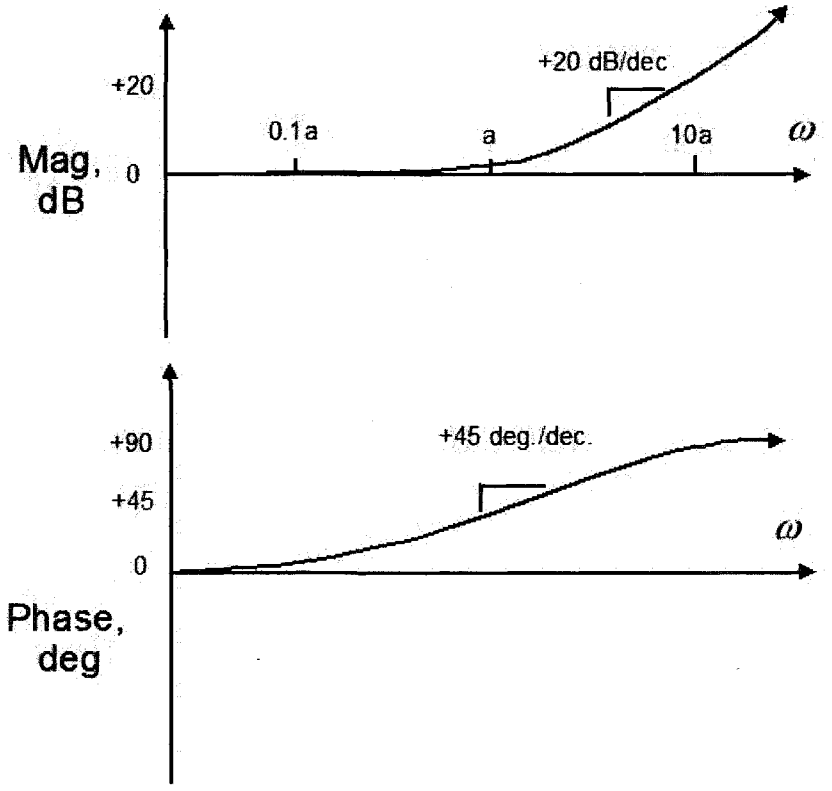


Fig. 2.10. Bode plot of a real axis stable zero

Unstable Zeros: Stable poles of the form $\frac{(s-a)}{a}$ will exhibit a low frequency magnitude of approximately -1. This is followed by a steady decrease in the phase angle of -45 deg./dec. (degrees per decade) after one tenth of the frequency of the zero. At $\omega \approx a$, the magnitude will be approximately +3dB and will sharply increase at +20 dB/dec. (decibels per decade) thereafter.

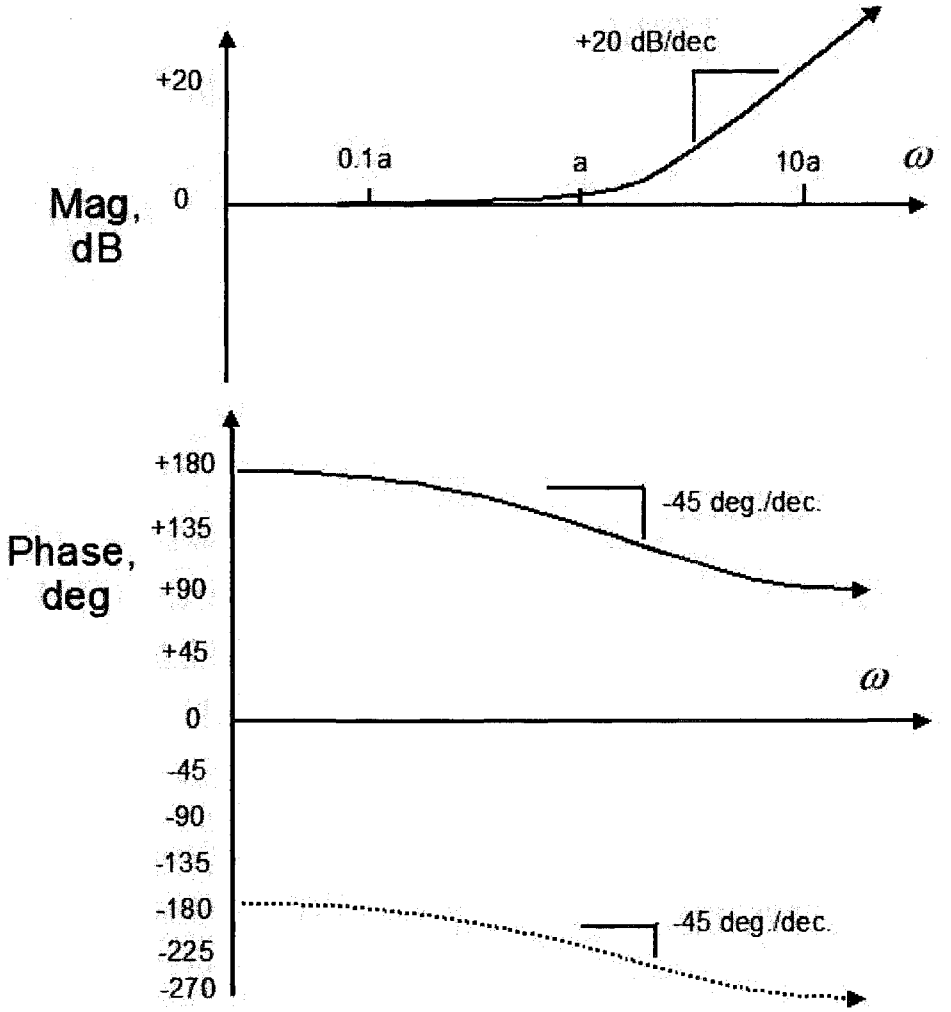


Fig. 2.11. Bode plot of an unstable real axis zero

2.14.3 Oscillatory Poles

Stable Oscillatory Poles: Stable poles of the form $\frac{\omega_n^2}{\left(s^2 + 2\zeta\omega_n s + \omega_n^2\right)}$ will exhibit a low frequency magnitude of approximately 1. If the damping, ζ, is less than about 0.1, there will be a sharp decrease of -180 deg. at $\omega \approx \omega_n$. If the damping, ζ, is closer to 1, then there will be a steady decrease of -90 deg./dec. at one tenth of the natural frequency.

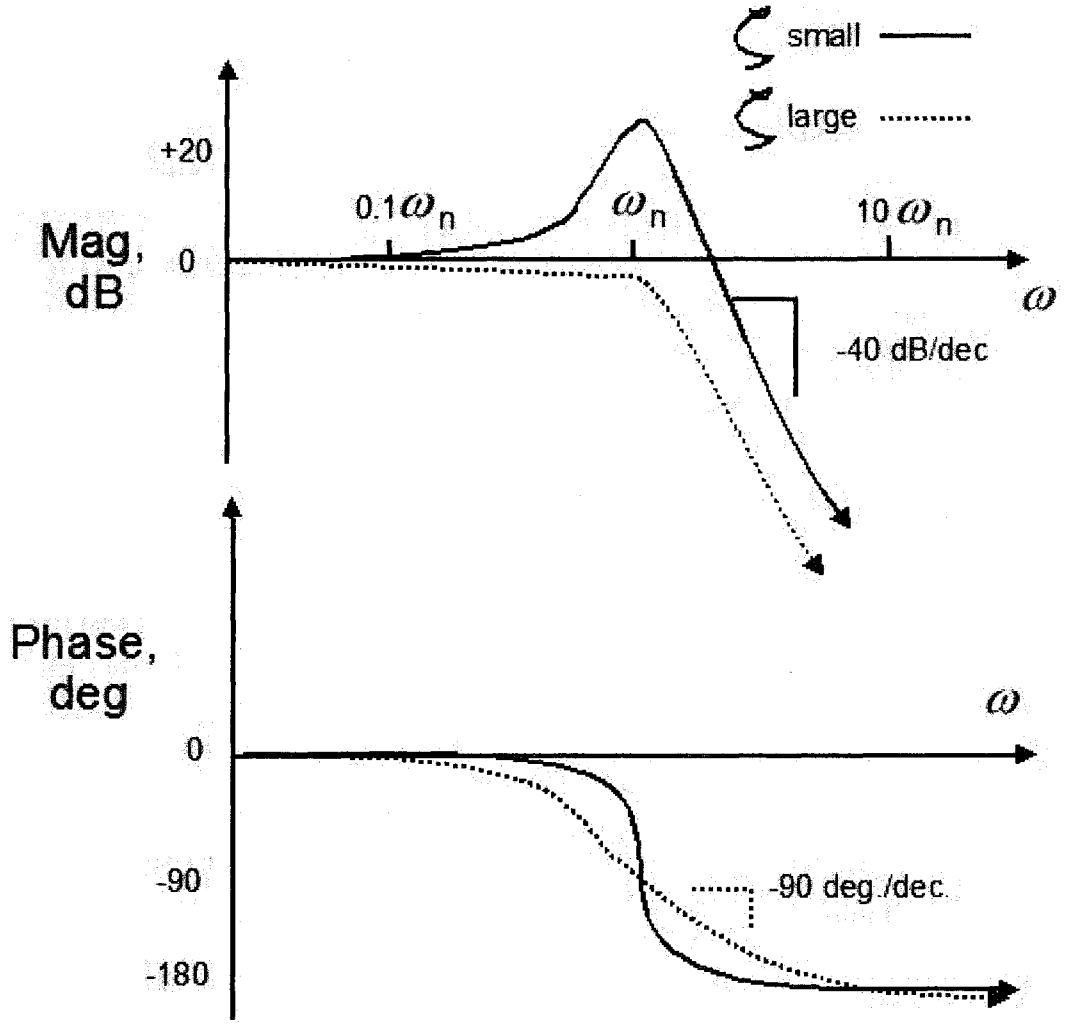


Fig. 2.12. Bode plot of stable oscillatory poles

Unstable Oscillatory Poles: Stable poles of the form $\frac{\omega_n^2}{\left(s^2 - 2\zeta\omega_n s + \omega_n^2\right)}$ will exhibit a low frequency magnitude of approximately 1. If the damping, ζ, is less than about 0.1, there will be a sharp increase of +180 deg. at $\omega \approx \omega_n$. If the damping, ζ, is closer to 1, then there will be a steady increase of -90 deg./dec. at one tenth of the natural frequency.

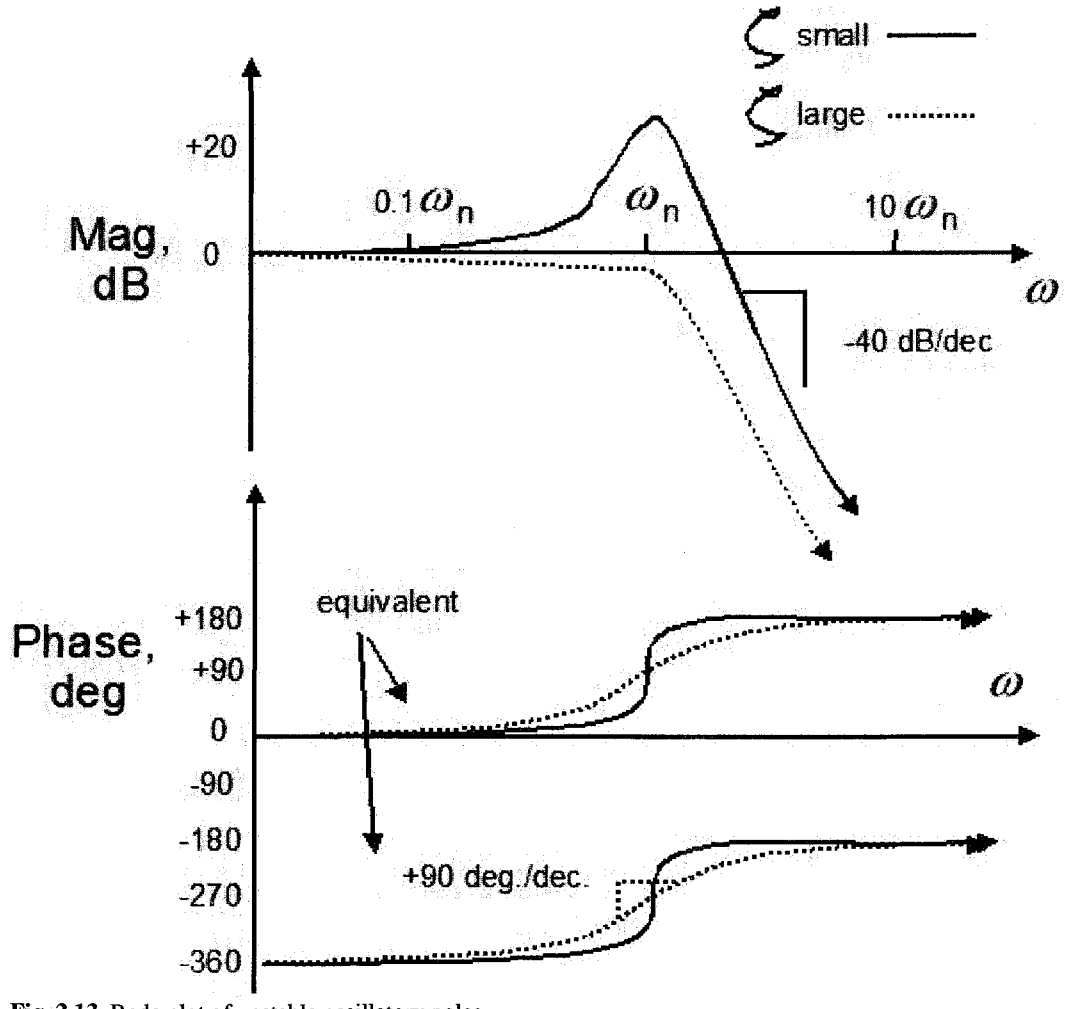


Fig. 2.13. Bode plot of unstable oscillatory poles

2.14.4 Oscillatory Zeros

Stable Oscillatory Zeros: Stable poles of the form $\frac{\left(s^2+2\zeta\omega_n s+\omega_n^2\right)}{\omega_n^2}$ will exhibit a low frequency magnitude of approximately 1. If the damping, ζ, is less than about 0.1, there will be a sharp decrease of -180 deg. at $\omega \approx \omega_n$. If the damping, ζ, is closer to 1, then there will be a steady decrease of -90 deg./dec. at one tenth of the natural frequency.

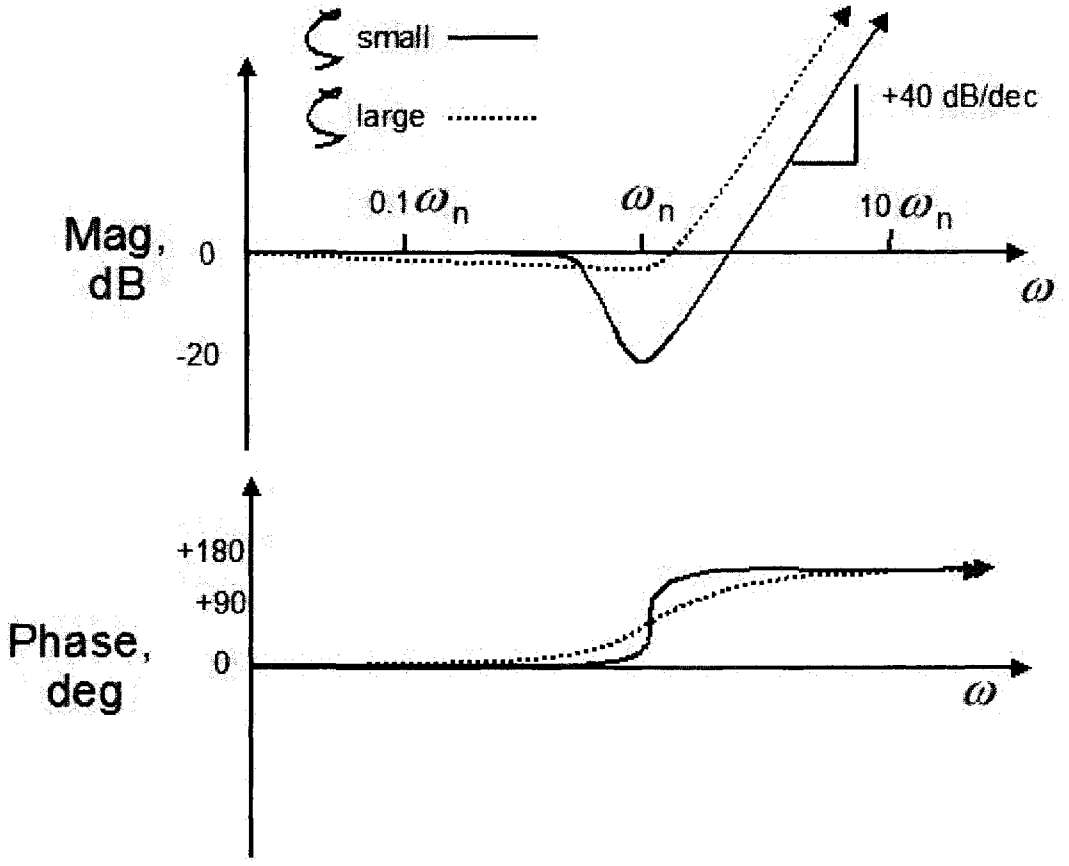


Fig. 2.14. Bode plot of a pair of stable oscillatory zeros

Unstable Oscillatory Zeros: Stable poles of the form $\frac{\left(s^2 - 2\zeta\omega_n s + \omega_n^2\right)}{\omega_n^2}$ will exhibit a low frequency magnitude of approximately 1. If the damping, ζ, is less than about 0.1, there will be a sharp decrease of -180 deg. at $\omega \approx \omega_n$. If the damping, ζ, is closer to 1, then there will be a steady decrease of -90 deg./dec. at one tenth of the natural frequency.

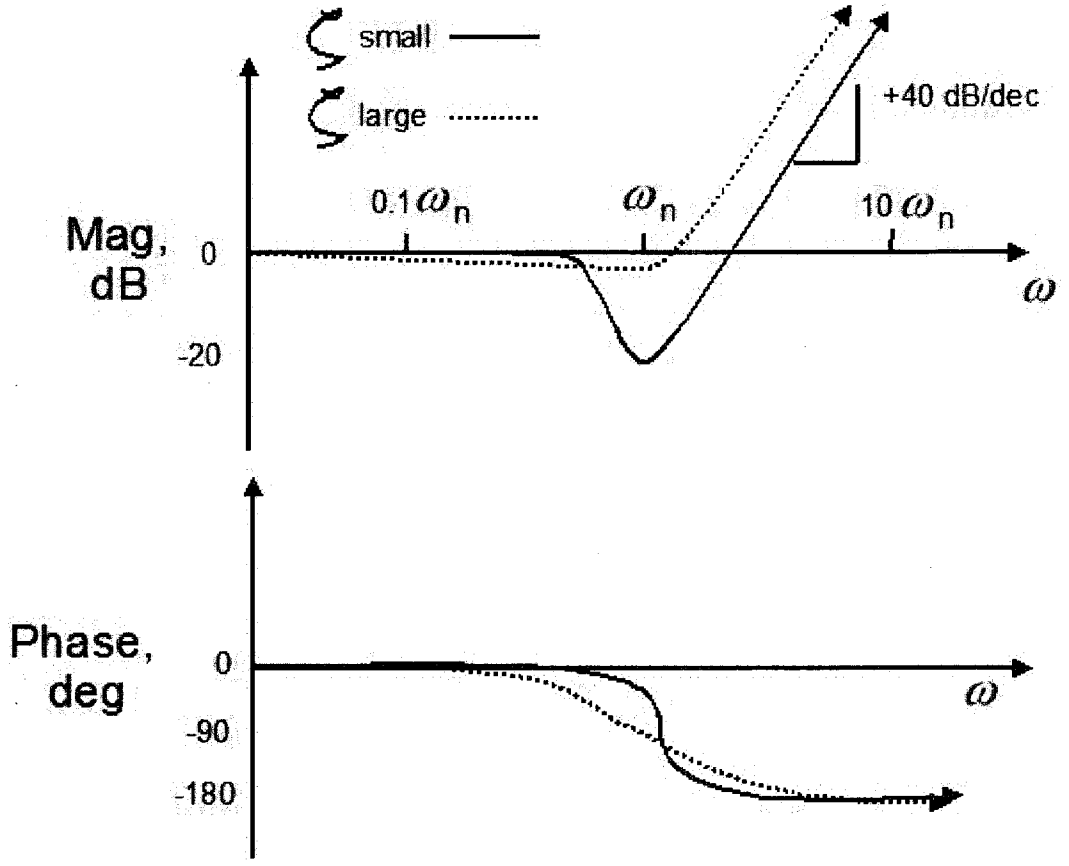


Fig. 2.15. Bode plot of a pair of unstable zeros

2.14.5 Superposition Principle

With multiple poles and zeros, the phase angle of the poles and zeros are added together. The magnitude of the poles and zeros are multiplied by each other. If the magnitude plot is logarithmic, then the magnitudes add with each other in dB [2.1,2.2].

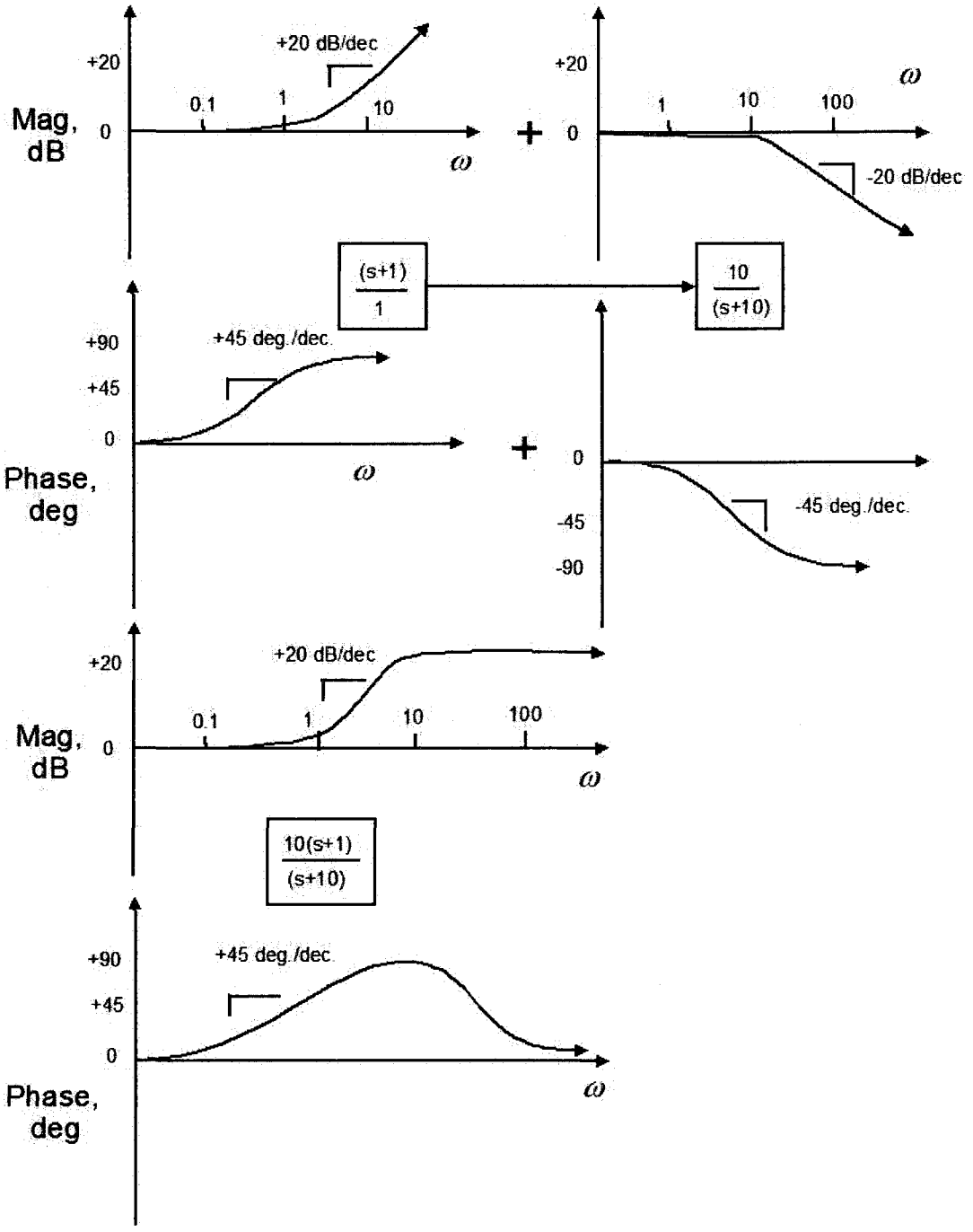


Fig. 2.16. Superposition principle for approximate bode plots

2.15 Loop Shaping Concept

Suppose that a loop shape, $L(s)$, has been achieved by a design of a compensator, $C(s)$, with unity feedback. The idea is to design the compensator, $C(s)$, such that there mostly a first-order like response in a certain region called the *crossover region* where $L(s) \approx 1$ or 0 dB.

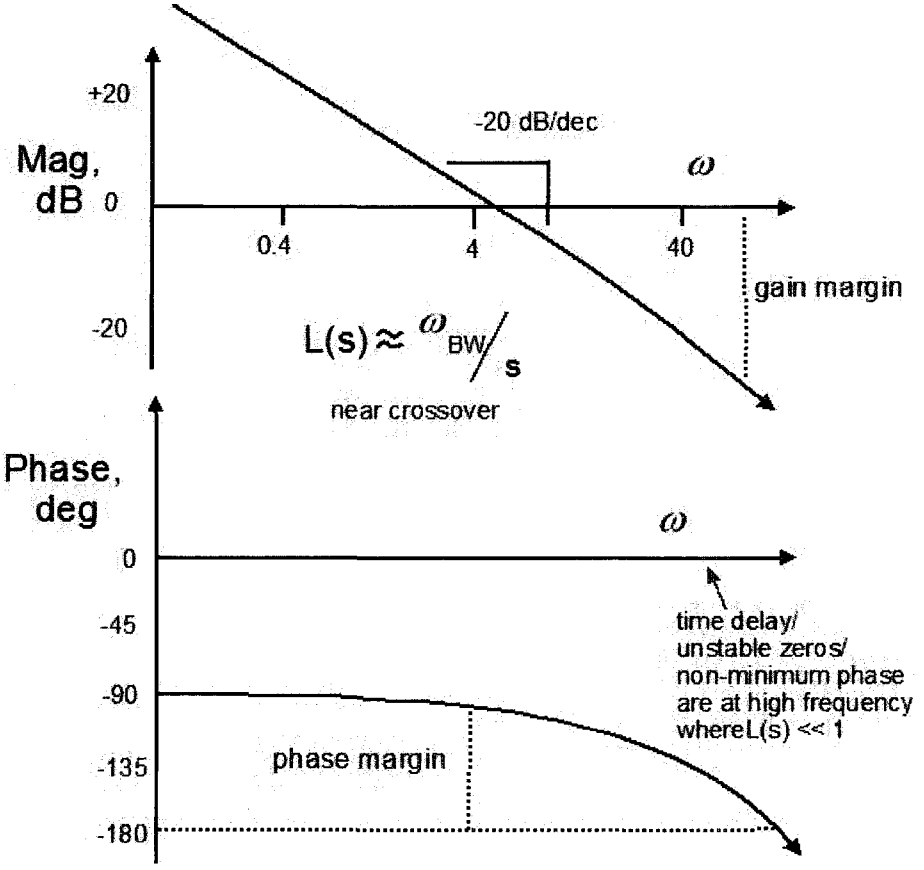


Fig. 2.17. The loop-shaping idea is that one tries to make the plant and controller look like an integrator near crossover.

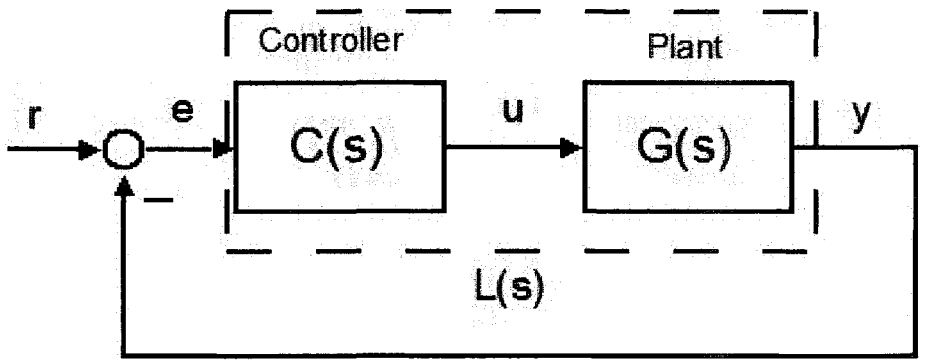


Fig. 2.18. The loop-shape, L(s), is the controller and plant in series.

This means that the bode plot follows a -20dB/dec. response over the *crossover region.*

Example 2.5

$$G(s)=\frac{-3(s+0.2)(s-40)}{(s^2-0.2s+1)(s+0.6)(s+40)}$$

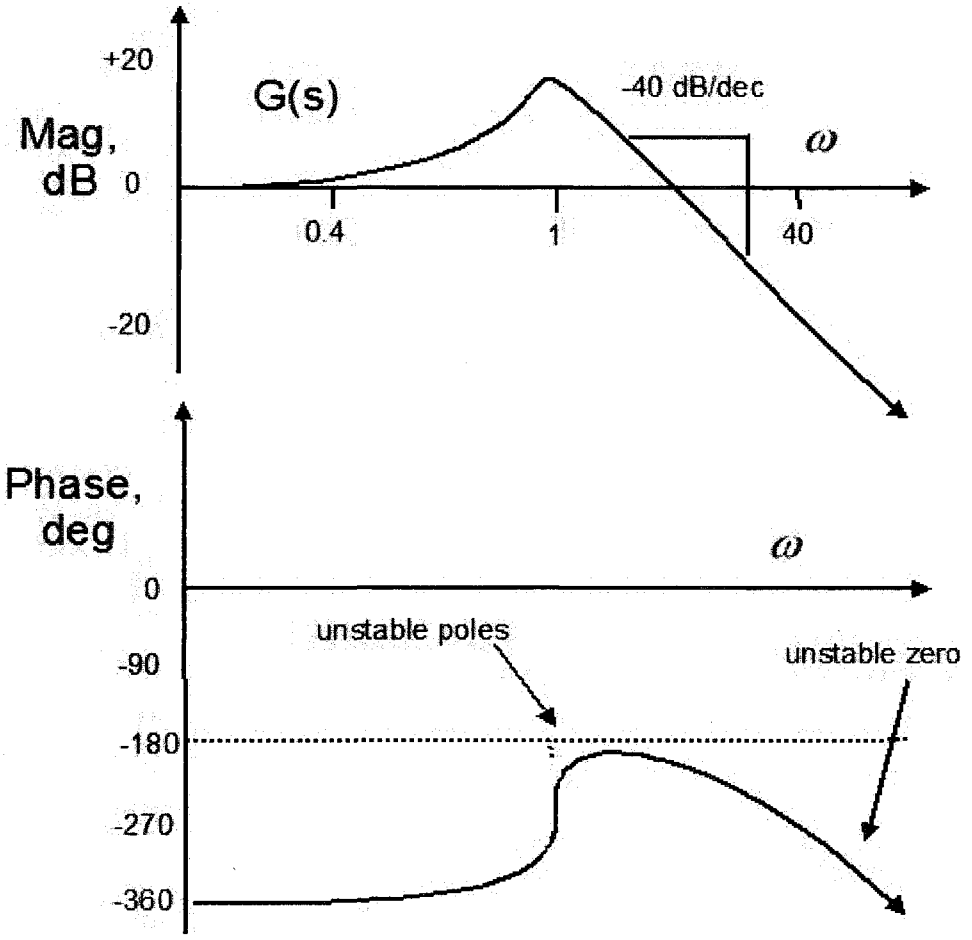


Fig. 2.19. Open loop bode plot

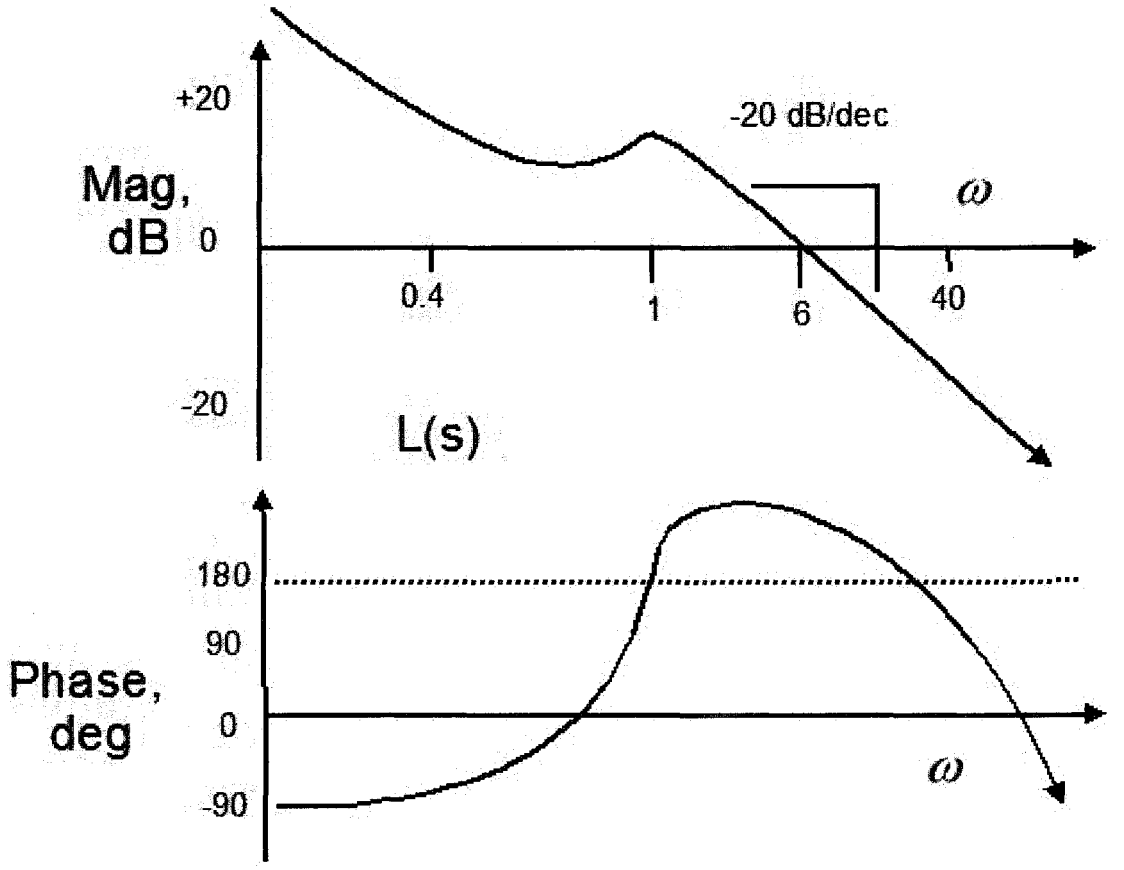


Fig. 2.20. Total loopshape has an integrator-like response near crossover at 6 rad./sec. The bandwidth of the closed loop system is slightly greater than 6 rad./sec., but the crossover frequency is usually fairly close to the bandwidth.

Now, to do the compensator design, we make stable pole-zero cancellations of the $\frac{s+0.2}{s+0.6}$ portion. We include an integrator. We also want to put a pair of stable zeros in the frequency of 1 rad./sec. to offset the influence of the unstable poles. However, we want to lead the unstable zeros slightly, so we put a pair of zeros at 0.5 rad./sec. into the compensator. At this point the compensator has 1 more zero than poles. This would result in an improper transfer function, so we have to put a lag into the compensator at high frequency, this is done at 40 rad./sec. We also decide to cross over at about 6 rad./sec. Hence, the gain of 6 appears in the compensator below. The gain of 3 appears in the denominator of C(s) to cancel the gain of 3 in the numerator of G(s).

$$C(s)=\frac{6(s+0.5)^2(s+0.6)}{3s(s+0.2)}\frac{40}{(s+40)}$$

2.16 Closed Loop Stability with Bode Plot and Nyquist Plot

Suppose that a loop shape, $L(s)$, has been achieved by a design of a compensator, $C(s)$ with unity feedback as in Fig. 2.18.

$$G(s)C(s) = L(s) \tag{2.24}$$

We want to analyze stability from the Nyquist Plot of $L(s)$, and we will use the Bode plot of $L(s)$ to help us.

2.16.1 D-Contour Regions

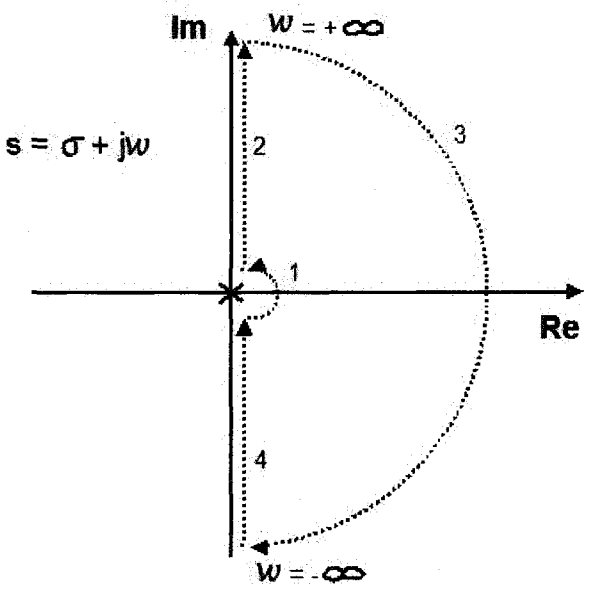


Fig. 2.20. Contour regions for a loop shape with integral action, i.e. there is at least one s in the denominator of the loop shape.

The D-contour regions to follow for a nyquist plot are summarized as follows:

D-Contour region 1: This is only necessary for loop shapes with integral action. It has an infinitely small radius, ε. The contour follows $s = \varepsilon^{j\phi}$ with $-\pi/2 \le \phi \le \pi/2$.

D-Contour region 2: This basically follows $s = j\omega$ from $0 \le \omega \le +\infty$. Mapping this contour onto a Nyquist plot of $L(s)$ can be obtained directly from a Bode plot.

D-Contour region 3: This follows a radius of infinite frequency. If the relative degree of $L(s)$ is 1 or greater (i.e. more poles than zeros), then $L(\infty) = 0$. For this case, when mapping onto a Nyquist plot, it will go to the origin.

D-Contour region 4: This follows $s = j\omega$ from $-\infty \le \omega \le 0$ and the mapping should be the mirror image the mapping obtained from contour region 2.

2.16.2 Mapping a Nyquist Plot

Region 1:

Avoid this step and move onto region 2 if there are no integrators in $L(s)$. This will sweep an arc of infinite radius.

$s = \varepsilon^{j\phi}$ and $-\pi/2 \le \phi \le \pi/2$, where ε is an arbitrarily small constant.

$L(s) = \varepsilon^{-n} e^{j\theta}$ and $\theta = -(n\phi + m\pi + p\pi)$

Where

n = number of integrators in $L(s)$,

m = number of real poles and zeros in the RHP of the s-plane,

p = 0 if the gain of $L(s)$, or 1 if the gain of $L(s)$ is negative.

Region 2:

This can be obtained from bode plot of $L(s)$ from $0 \le \omega \le +\infty$.

Region 3:

More poles than zeros: If the relative degree of $L(s)$ is 1 or greater (i.e. more poles than zeros), then $L(\infty) = 0$. For this case, when mapping onto a Nyquist plot, it will go to the origin.

Equal number of poles and zeros: Take the high frequency gain of the loop shape, it should stay pinned there, $\lim_{s\to\infty} L(s)$, Ex: $\lim_{s\to\infty} \frac{0.1(s+1)(s-1)}{(s+3)(s+4)} = 0.1$.

Region 4:

This should be the mirror image of region 2 reflected by the real axis. This is because ω is negative from $-\infty \le \omega \le 0$ and if $W(j\omega) = a + bj$, then $W(-j\omega) = a - bj$.

2.16.3 Stability from a Nyquist Plot

By looking at the number of counterclockwise encirclements of -1 one can determine whether or not the closed loop system is stable.

N : the number of *clockwise* encirclements of -1.

P: the number of open-loop poles of $L(s)$

Z: the number of closed-loop poles

$Z = N + P,\ N = Z - P$

Example 2.6

$$L(s) = \frac{-10(s-20)}{s(s+20)}$$

Region 1:

n = 1, m = 1, p = 1

$\phi = -\pi/2 \quad \theta = -3\pi/2$

$\phi = 0 \quad \theta = -2\pi$

$\phi = \pi/2 \quad \theta = -5\pi/2$

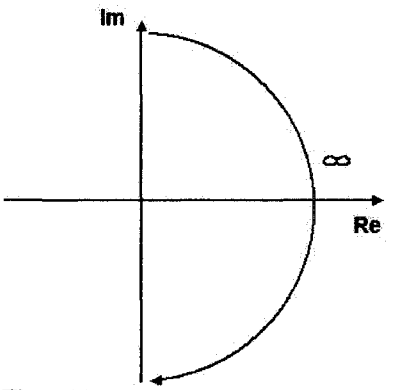
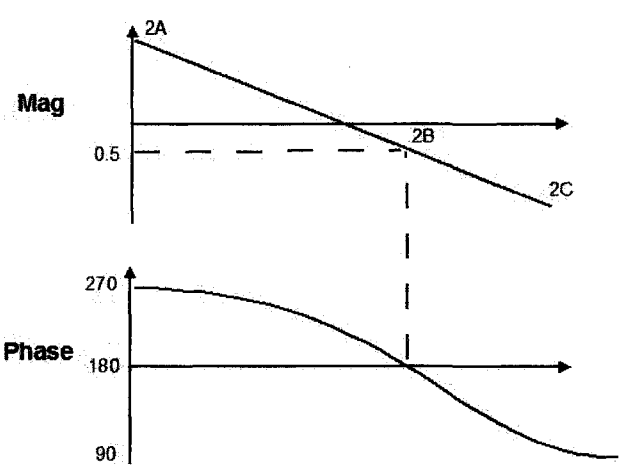


Fig. 2.21. Region 1

Region 2:



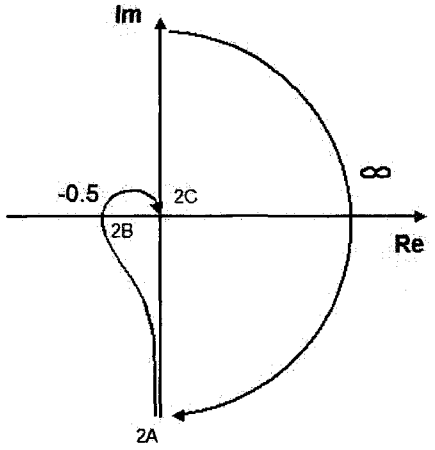


Fig. 2.22. Region 2

Region 3:

The mapping stays at the origin because the relative degree is 1.

Region 4:

The mapping is simply a reflection of region 2.

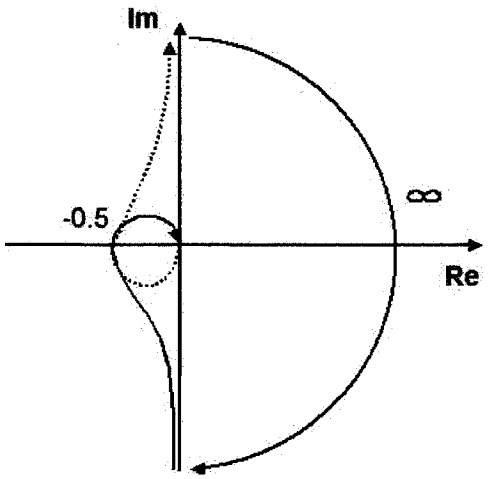


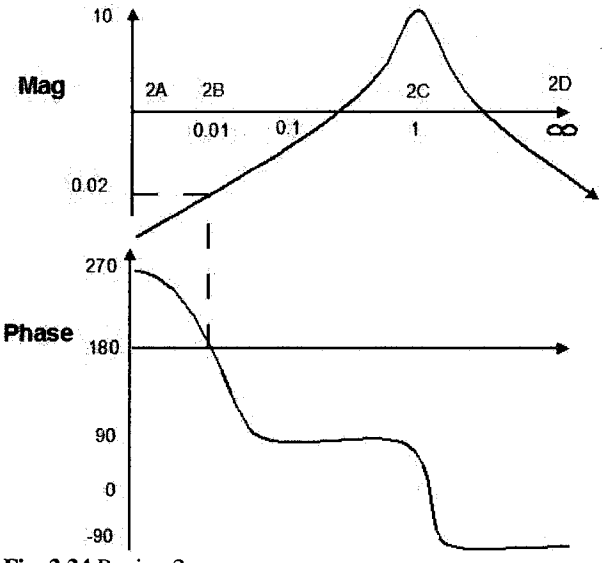
Fig. 2.23, Region 4. N = 0, there are no encirclements of -1, P = 0, open-loop stability, and therefore Z = N + P = 0 and the system is closed-loop stable

Example 2.7

$$L(s)=\frac{2s(s-0.01)}{(s+0.01)(s^2+0.2s+1)}$$

Region 1: Does not need to be done, There are no integrators.

Region 2:



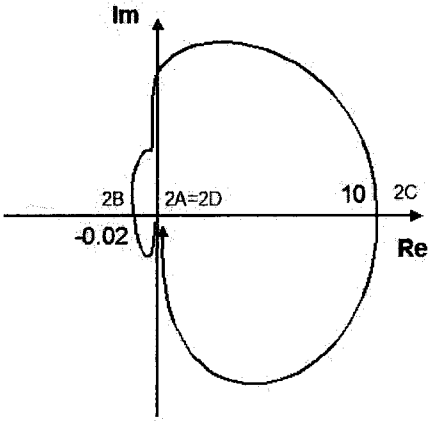


Fig. 2.24 Region 2.

Region 4:

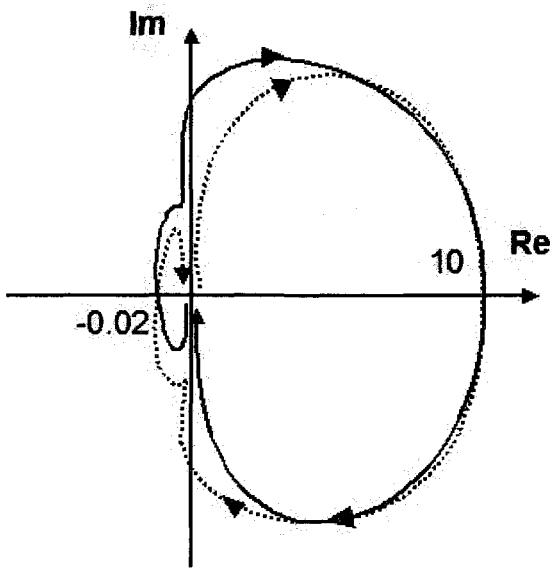


Fig. 2.25 Region 4. N = 0, there are no encirclements of -1, P = 0, open-loop stability, and therefore Z = N + P = 0 and the system is closed-loop stable

Example 2.8

$$L(s)=\frac{2s(s-0.01)}{(s+0.01)(s^2-0.2s+1)}$$

Region 1: Does not need to be done, There are no integrators.

Region 2:

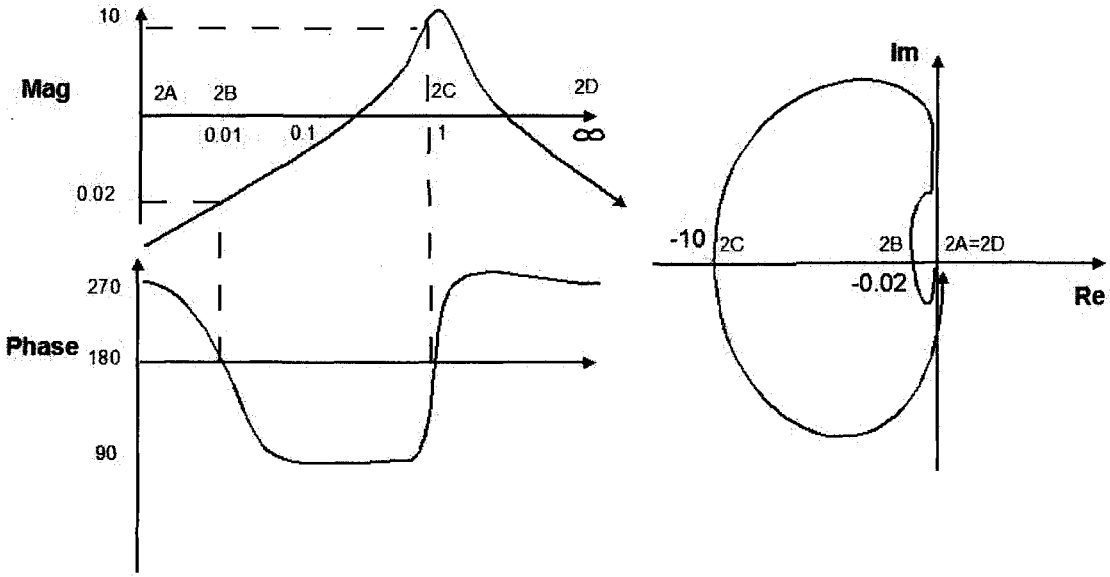


Fig. 2.26 Region 2

Region 4:

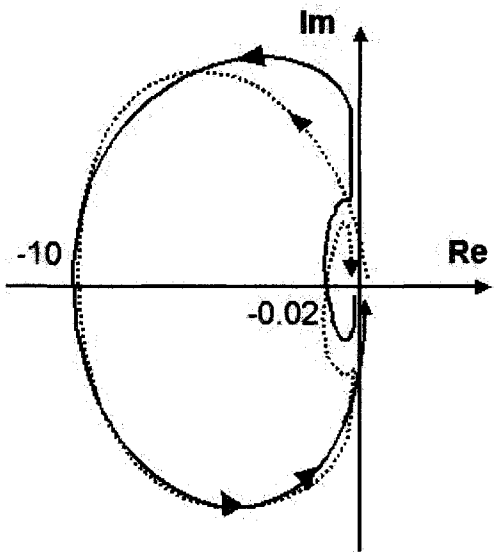


Fig. 2.27 Region 4. N = -2, there are two CCW encirclements of -1, P = 2, open-loop unstable poles, and therefore Z = N + P = 0 and the system is closed-loop stable

Example 2.9

$$L(s)=\frac{2(s+0.1)^2}{s^2(s-0.1)}$$

Region 1:

n = 2, m = 1, p = 0

$$\phi=-\pi/2 \qquad \theta=0$$
$$\phi=-\pi/4 \qquad \theta=-\pi/2$$
$$\phi=0 \qquad \theta=-\pi$$
$$\phi=\pi/4 \qquad \theta=-3\pi/2$$
$$\phi=\pi/2 \qquad \theta=-2\pi$$

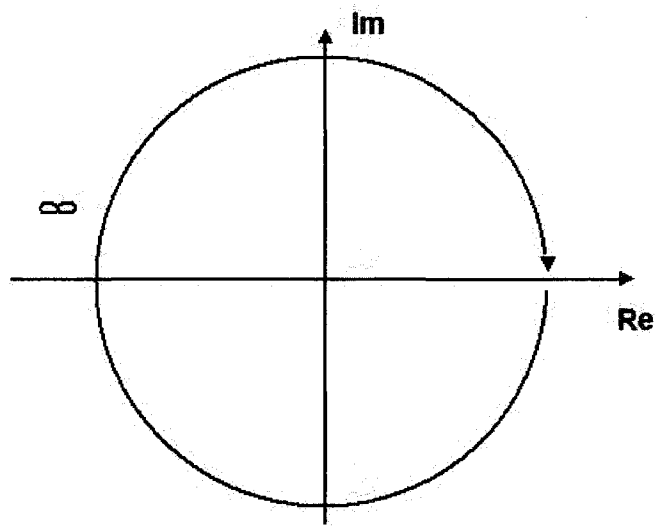


Fig. 2.28 Region 1

Region 2:

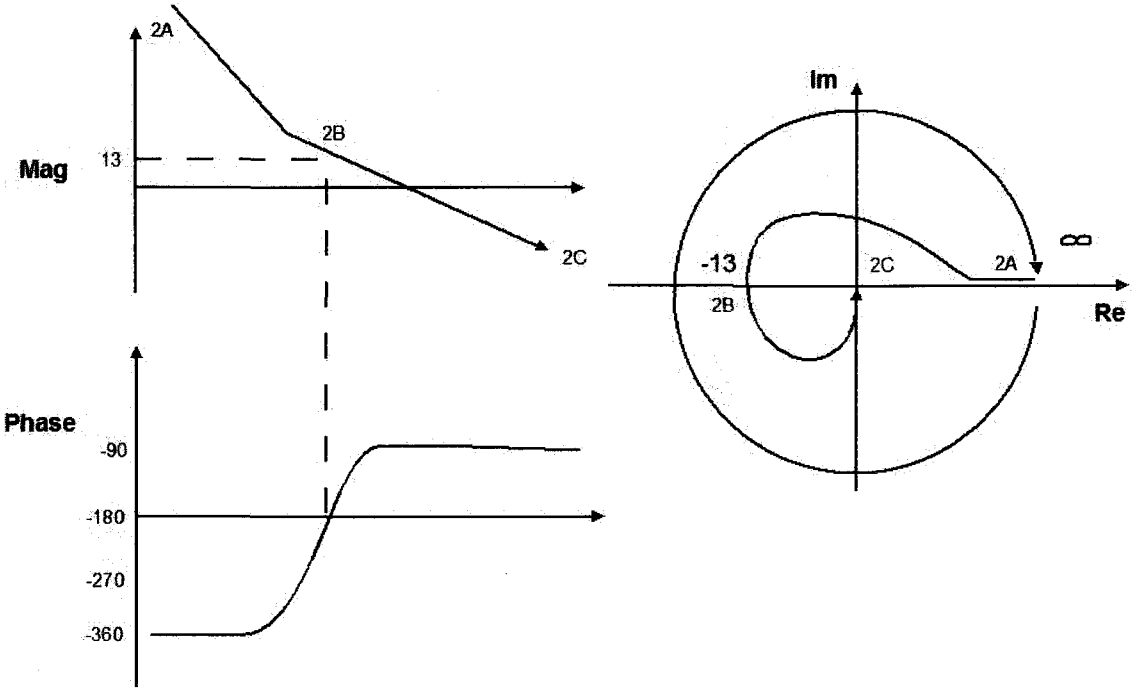


Fig. 2.29 Region 2

Region 4:

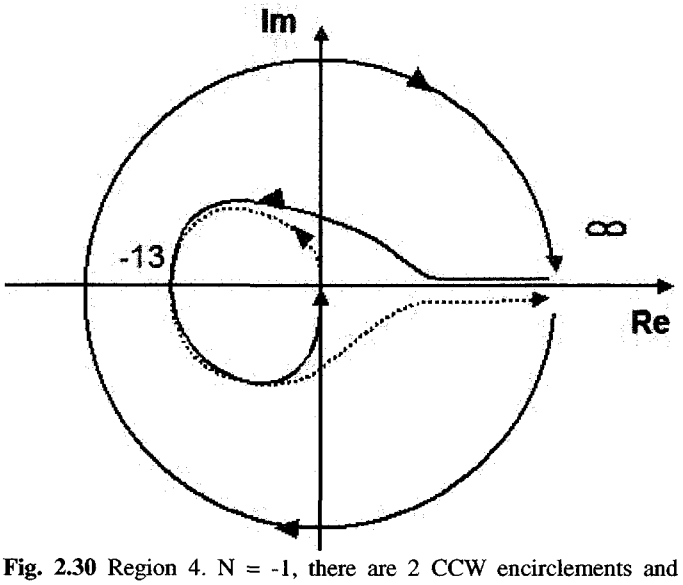


Fig. 2.30 Region 4. N = -1, there are 2 CCW encirclements and 1 CW encirclements of -1, P = 1, open-loop instability, and therefore Z = N + P = 0 and the system is closed-loop stable

2.17 Transmission Zeros of Square MIMO Systems

Transmission zeros for *m x m* square systems can be related to the idea of infinite gain [2.9]. A transmission zero, s_o, would be defined as the poles of closed loop with infinite gain.

$$roots \quad of \quad \boldsymbol{det}(I + kG(s)) = 0 \quad as \quad k \to \infty$$

As the gain, k, becomes large, $\boldsymbol{det}(I + kG(s)) \approx \boldsymbol{det}(kG(s)) = k^m \boldsymbol{det}(G(s))$.Therefore, one would approximate the transmission zeros would by eq. 2.25.

$$\boldsymbol{det}(G(s_o)) = 0 \quad \forall \quad s_o \in TZ \tag{2.25}$$

In eq. 2.25, s_o, is a transmission zero and *TZ* is the set of all transmission zeros of $G(s)$.

Also, a general system matrix, $S_{ys}(s_o)$, has been used to define transmission zeros for systems of any size. If $S_{ys}(s_o)$ loses rank, then the value, s_o, is a transmission zero of the transfer function [2.9,2.3-2.6].

$$S_{ys}(s_o) = \begin{bmatrix} s_o I - A & -B \\ C & D \end{bmatrix} \tag{2.26}$$

If $G(s)$ is square, then $S_{ys}(s_o)$ will be square and one can take the determinant of $S_{ys}(s_o)$ to define where the transmission zeros. Using Schur's determinant formula for a partitioned matrix helps reduce the determinant.

$$\boldsymbol{det}\left(\begin{bmatrix} s_o I - A & -B \\ C & D \end{bmatrix}\right) = \boldsymbol{det}(s_o I - A)\boldsymbol{det}\left(D + C(s_o I - A)^{-1} B\right) \tag{2.27}$$

Using eq. 2.27, an alternative definition of transmission zeros for square systems can be given in eq. 2.28.

$$\boldsymbol{det}(s_o I - A)\boldsymbol{det}(G(s_o)) = 0 \quad \forall \quad s_o \in TZ \tag{2.28}$$

Results from eq. 2.28 and 2.25 will give the same result for transmission zeros as long as there are not open loop poles and zeros in the same location. If there are open loop poles and zeros in the same place, usage of eq. 2.28 is recommended.

The important thing about zeros is that they reflect what will happen to the system dynamics if high gain is used. Plants with unstable zeros will become destabilized if high gain is used near the frequency of those unstable zeros.

Example 2.10

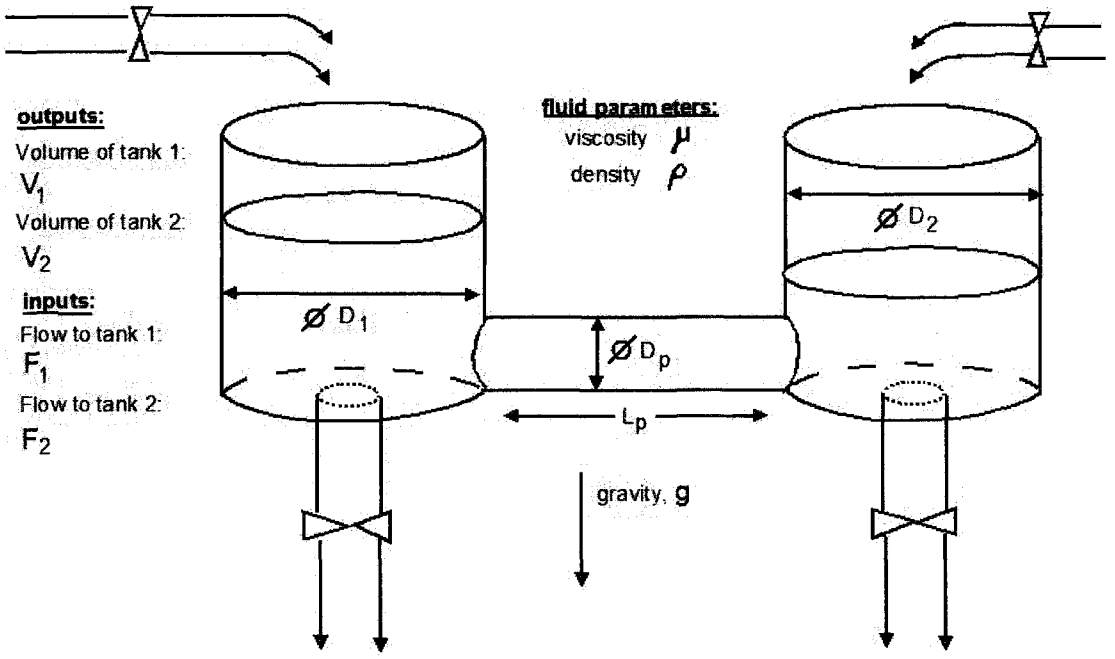


Fig. 2.31. Two tank system with a pipe of diameter D_p and length L_p. The first tank has diameter D_1 and the second tank has diameter D_2. The inputs are volumetric flows to tanks 1 and 2, F_1 and F_2, respectively. The fluid has viscosity, μ, and density, ρ

Assuming the flow is laminar across the pipe, a linear model for the two tank system was derived using a bond graph theory [2.8]. The state space is shown below.

$$I_{pipe}=\frac{\rho L_p}{\pi/4\,D_p^2} \quad R_{pipe}=\frac{32\mu L_p}{D_p^2} \quad C_1=\frac{\pi/4\,D_1^2}{\rho g} \quad C_2=\frac{\pi/4\,D_2^2}{\rho g}$$

$$a_1=\left(I_{pipe}C_1\right)^{-1} \quad a_2=-\left(I_{pipe}C_2\right)^{-1} \quad a_3=-R_{pipe}\left(I_{pipe}\right)^{-1}$$

$$A=\begin{bmatrix}0 & 0 & -1\\ 0 & 0 & 1\\ a_1 & a_2 & a_3\end{bmatrix} \quad B=\begin{bmatrix}1 & 0\\ 0 & 1\\ 0 & 0\end{bmatrix} \quad C=\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}$$

$$G(s)=C(sI-A)^{-1}B=\begin{bmatrix}\dfrac{s(s-a_3)-a_2}{s\left(s^2-a_3s+(a_1-a_2)\right)} & \dfrac{-a_2}{s\left(s^2-a_3s+(a_1-a_2)\right)}\\ \dfrac{a_1}{s\left(s^2-a_3s+(a_1-a_2)\right)} & \dfrac{s(s-a_3)+a_1}{s\left(s^2-a_3s+(a_1-a_2)\right)}\end{bmatrix}$$

Now, to calculate the transmission zeros, one can simply take the determinant of G(s), multiply by the characteristic polynomial, $\det(sI\text{-}A)$, and set it equal to 0.

$$det(s_o I - A)det(G(s_o)) = \frac{s_o(s_o - a_3)(s_o^2 - a_3 s_o + (a_1 - a_2))}{s_o(s_o^{\ 2} - a_3 s_o + (a_1 - a_2))} = 0$$

$$s_o = a_3 \quad TZ = \{a_3\}$$

Hence, we can conclude that the transmission zero of this two tank system is stable and is determined by the parameters of the pipe.

Example 2.11

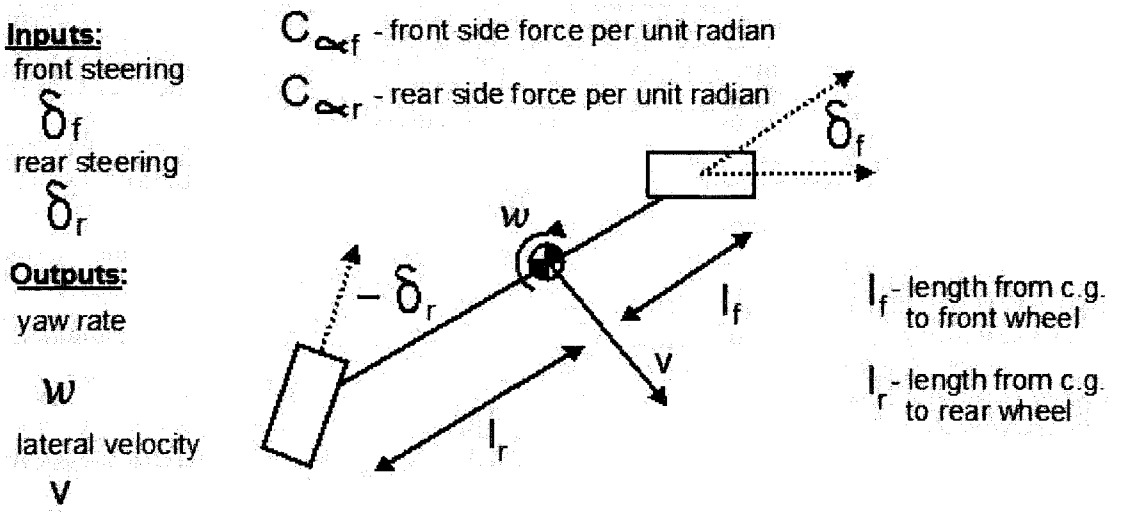


Fig. 2.32 The bicycle model of a vehicle. Inputs are front steering angle δ_f and rear steering angle, δ_r.

$$b_1 = \frac{2C_{af}}{m} \quad b_2 = \frac{2C_{ar}}{m} \quad b_3 = \frac{2C_{af}l_f}{I_{zz}} \quad b_4 = \frac{-2C_{ar}l_r}{I_{zz}}$$

$$a_1 = \frac{-2(C_{af} + C_{ar})}{mU_o} \quad a_2 = -U_o - \frac{-2(C_{af}l_f - C_{ar}l_r)}{mU_o}$$

$$a_3 = \frac{-2(C_{af}l_f - C_{ar}l_r)}{mI_{zz}} \quad a_4 = \frac{-2(C_{af}l_f^2 + C_{ar}l_r^2)}{mI_{zz}}$$

$$A = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \quad B = \begin{bmatrix} b_1 & b_2 \\ b_3 & b_4 \end{bmatrix} \quad C = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$G(s) = C(sI - A)^{-1}B =$$

$$\begin{bmatrix} \frac{a_3 b_1 + (s - a_1)b_3}{(s^2 + (-a_4 - a_1)s + (a_1 a_4 - a_2 a_3))} & \frac{a_3 b_2 + (s - a_1)b_4}{(s^2 + (-a_4 - a_1)s + (a_1 a_4 - a_2 a_3))} \\ \frac{(s - a_4)b_1 + a_2 b_3}{(s^2 + (-a_4 - a_1)s + (a_1 a_4 - a_2 a_3))} & \frac{(s - a_4)b_2 + a_2 b_4}{(s^2 + (-a_4 - a_1)s + (a_1 a_4 - a_2 a_3))} \end{bmatrix}$$

There are actually no transmission zeros in this case.

$$det(s_o I - A)det(G(s_o)) = \frac{(-b_1b_4 + b_2b_3)\left(s_o^{\,2} + (-a_4 - a_1)s_o + (a_1a_4 - a_2a_3)\right)}{\left(s_o^{\,2} + (-a_4 - a_1)s_o + (a_1a_4 - a_2a_3)\right)} = 0$$

$$TZ = \varnothing$$

2.18 Other System Zeros

Zeros of non-square systems can become confusing. Limitations encountered with the unstable zeros of non-square systems may not be conveyed well. Additionally, zero calculation may require the Smith-Mcmillan form [2.4], which is rarely explained well.

Additionally, some authors choose to categorize zeros which can lead to more confusion. Zeros can be categorized into transmission zeros, invariant zeros, blocking zeros, zeros at infinity, input decoupling zeros, and output decoupling zeros. Terms are used with mixed consistency in control systems literature. The different types of zeros will be discussed briefly without too much math.

Remarks:

1. Transmission zeros are the same as invariant zeros if the plant (or system) is completely observable and completely controllable [2.4].
2. A blocking zero makes the entire transfer function zero in all entries, $G(s_o) = 0_{p\times m}$.
3. The input decoupling zeros are also the uncontrollable eigenvalues of the system [2.4].
4. The output decoupling zeros are also the unobservable eigenvalues of the system [2.4].
5. The input-output decoupling zeros are also the hidden modes of the system (i.e. the uncontrollable and unobservable eigenvalues) [2.4].
6. A zero at infinity is synonymous with no zero. It basically means that at infinite gain, one or more poles will be located at $\pm\infty$. Example 2.11 had no zeros, yet it also had 2 zeros at infinity. Theoretically, one could use infinite feedback gain for the theoretical model of the car in example 2.11. In reality, unstable zeros from time delay would manifest themselves at high frequency, making very high gain controllers unstable.
7. Zeros of non-square plants are not always meaningful. Consider the plant in eq. 2.29.

$$G(s) = \begin{bmatrix} \frac{-(s-20)}{(s+20)} & \frac{-2(s-30)}{(s+30)} & \frac{-3(s-40)}{(s+40)} \\ \frac{-3(s-20)}{(s+20)} & \frac{-2(s-30)}{(s+30)} & \frac{-(s-40)}{(s+40)} \end{bmatrix} \tag{2.29}$$

There are unstable SISO-like zeros at s = 20, 30, and 40 which cause the plant to lose column rank. However, none of these zeros cause a loss of row rank. There are therefore no transmission zeros for this plant. This is a problem because eq. 2.29 basically approximates a matrix of input time delays of 0.1, 0.0667, and 0.05 seconds, respectively. A combination of time delays and high gain is always destabilizing in feedback systems. If unstable transmission zeros give an indication of achievable performance, then there is potential for misdirection.

2.19 Zeros are Unaffected by Feedback

This section is intended to note that feedback does not affect transmission zeros at all [2.9]. It is a waste of time to attempt changing system zeros by using feedback. Achievable performance as defined by zeros can only be modified by looking at different outputs, getting more inputs or changing the inputs.

The system matrix, $S_{ys}(s_o)$, below defines the system zeros more generally. If the system matrix loses rank, then s_o is a transmission zero.

$$S_{ys}(s_o) = \begin{bmatrix} s_o I - A & -B \\ C & D \end{bmatrix} \tag{2.30}$$

Lemma 2.19.1 – By Sylvester's Inequality the product of a rank deficient matrix and any matrix is rank deficient.

If one can express the augmented system, $S_{ys}^{aug}(s_o)$, as $S_{ys}(s_o)W$, then one can conclude that there has been no change to system zeros by lemma 2.19.1.

$$S_{ys}^{aug}(s_o) = S_{ys}(s_o)W = \begin{bmatrix} s_o I - A & -B \\ C & D \end{bmatrix} W \tag{2.31}$$

If a system zero makes $S_{ys}^{aug}(s_o)$ rank deficient, then $S_{ys}(s_o)$ is rank deficient as well. Note that the matrix, W, is whatever it is. So a zero of $S_{ys}^{aug}(s_o)$ is a zero of $S_{ys}(s_o)$.

$$zeros \quad of \quad S_{ys}^{aug}(s_o) = S_{ys}(s)W \quad \Leftrightarrow \quad zeros \quad of \quad S_{ys}(s)$$

Lemma 2.19.2 – State feedback does not effect transmission zeros

Proof:

$$S_{ys}^{aug}(s_o)=\begin{bmatrix} s_oI-A+BK & -B \\ C & D \end{bmatrix}=\begin{bmatrix} s_oI-A & -B \\ C & D \end{bmatrix}\begin{bmatrix} I & 0 \\ -K & I \end{bmatrix}$$
$$S_{ys}^{aug}(s_o)=S_{ys}(s_o)W,\quad W=\begin{bmatrix} I & 0 \\ -K & I \end{bmatrix} \tag{2.32}$$

In eq. 2.32, we have successfully arranged, $S_{ys}^{aug}(s_o)$ as $S_{ys}(s_o)W$ and by Lemma 2.19.1 we can say with confidence that state feedback does not effect transmission zeros.

Lemma 2.19.3 – Output feedback does not effect transmission zeros

Proof:

$$S_{ys}^{aug}(s_o)=\begin{bmatrix} s_oI-A+B(I+KD)^{-1}KC & -B \\ C & D \end{bmatrix}=\begin{bmatrix} s_oI-A & -B \\ C & D \end{bmatrix}\begin{bmatrix} I & 0 \\ -(I+KD)^{-1}KC & I \end{bmatrix}$$
$$S_{ys}^{aug}(s_o)=S_{ys}(z_o)W,\quad W=\begin{bmatrix} I & 0 \\ -(I+KD)^{-1}KC & I \end{bmatrix} \tag{2.33}$$

Therefore, in eq. 2.33, we have successfully arranged, $S_{ys}^{aug}(s_o)$ as $S_{ys}(s_o)W$ and by Lemma 2.19.1 we can say with confidence that state feedback does not effect transmission zeros.

2.20 Transmission Zeros of a Biproper System are Poles of the Inverse System

It is true that for any biproper transfer function, *G(s)*, the poles of $G(s)^{-1}$ are the transmission zeros of *G(s)*. The system matrix for a biproper (square and invertible) transfer function is described below.

$$S_{ys}(s_o)=\begin{bmatrix} s_oI-A & -B \\ C & D \end{bmatrix} \tag{2.34}$$

By Lemma 2.19.1, we know that the zeros of $S_{ys}(s_o)W$ are the zeros of $S_{ys}(s_o)$. Now, by some manipulations, we can also express the original system matrix $S_{ys}^{aug}(s_o)$ as $S_{ys}(s_o)W$. This is shown in eq. 2.35.

$$S_{ys}^{aug}(s_o)=\begin{bmatrix} s_oI-A & -B \\ C & D \end{bmatrix}\begin{bmatrix} I & 0 \\ 0 & D^{-1} \end{bmatrix}\begin{bmatrix} I & 0 \\ -C & I \end{bmatrix}=S_{ys}(s_o)W$$
$$S_{ys}^{aug}(s_o)=\begin{bmatrix} s_oI-A+BD^{-1}C & -BD^{-1} \\ 0 & I \end{bmatrix} \tag{2.35}$$

It can be seen that $S_{ys}^{aug}(s_o)$ will lose rank with the roots of $s_oI-A+BD^{-1}C$, which are incidentally the poles of the inverse system.

For completeness, we will very quickly derive the inverse dynamics in eq. 2.36 through eq. 2.39.

$$\dot{x}=Ax+Bu \quad y=Cx+Du \tag{2.36}$$

The inverse dynamics will have *u* as an output and *y* as an input, so we switch eq. 2.36 around in eq. 2.37.

$$u=(D)^{-1}(y-Cx) \tag{2.37}$$

We insert the relationship of eq. 2.37 into eq. 2.36 and arrive at the state space of the inverse dynamics in eq. 2.38.

$$\begin{aligned} \dot{x} &= Ax-B(D)^{-1}Cx+B(D)^{-1}y \\ u &= -(D)^{-1}Cx+(D)^{-1}y \end{aligned} \tag{2.38}$$

For completeness, we re-write eq. 2.38 in the laplace domain in eq. 2.39.

$$G^{-1}(s)=-D^{-1}C\left(sI-A+BD^{-1}C\right)^{-1}BD^{-1}+D^{-1} \tag{2.39}$$

Now we have shown that poles of $G^{-1}(s)$ are $s_oI-A+BD^{-1}C$ will also reduce the rank of eq. 2.35.

$$zeros \quad of \quad G(s) \quad \Leftrightarrow \quad poles \quad of \quad G^{-1}(s)$$

2.21 Zero Directions

Although there are multiple equivalent expressions of zero directions, a state space representation will be given [2.6]. Suppose that a given system zero has a value of s_o, then one can say that the following solution to the system matrix S_{ys} will give a solution to a non-zero state zero direction, $\bar{x}_o$, and a non-zero input zero direction, $\bar{u}_o$.

$$\begin{bmatrix} s_o I - A & -B \\ C & D \end{bmatrix} \begin{bmatrix} \bar{x}_o \\ \bar{u}_o \end{bmatrix} = 0 \tag{2.40}$$

To solve for these directions, one would solve for the right null space of the system matrix S_{ys}, evaluated at s_o. One can also rearrange eq. 2.40 in a more familiar state space form of eq. 2.41.

$$\begin{aligned} s_o \bar{x}_o &= A\bar{x}_o + B\bar{u}_o \\ y &= C\bar{x}_o + D\bar{u}_o = 0 \end{aligned} \tag{2.41}$$

In the time domain, this would mean that if the initial state were the state zero direction, $x(0) = \bar{x}_o$, and if the input were $u(t) = \bar{u}_o e^{s_o t}$, then $x(t) = \bar{x}_o e^{s_o t}$ and the output would be equal to 0 for all time.

Example 2.12:

We will find the zero directions of $s_o = a_3$, from the two tank apparatus of example 2.10.

$$S_{ys}(a_3) = \begin{bmatrix} a_3 & 0 & 1 & -1 & 0 \\ 0 & a_3 & -1 & 0 & -1 \\ -a_1 & -a_2 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \bar{x}_o \\ \bar{u}_o \end{bmatrix} = 0 \tag{2.42}$$

$$\bar{x}_o = \begin{bmatrix} 0 & 0 & c \end{bmatrix}^T \quad \bar{u}_o = \begin{bmatrix} c & -c \end{bmatrix}^T \quad c \in \Re^1$$

2.22 Sensitivity Functions for Unity Feedback

Suppose that a loop shape, $L(s)$, has been achieved by the design of a compensator, $C(s)$ with

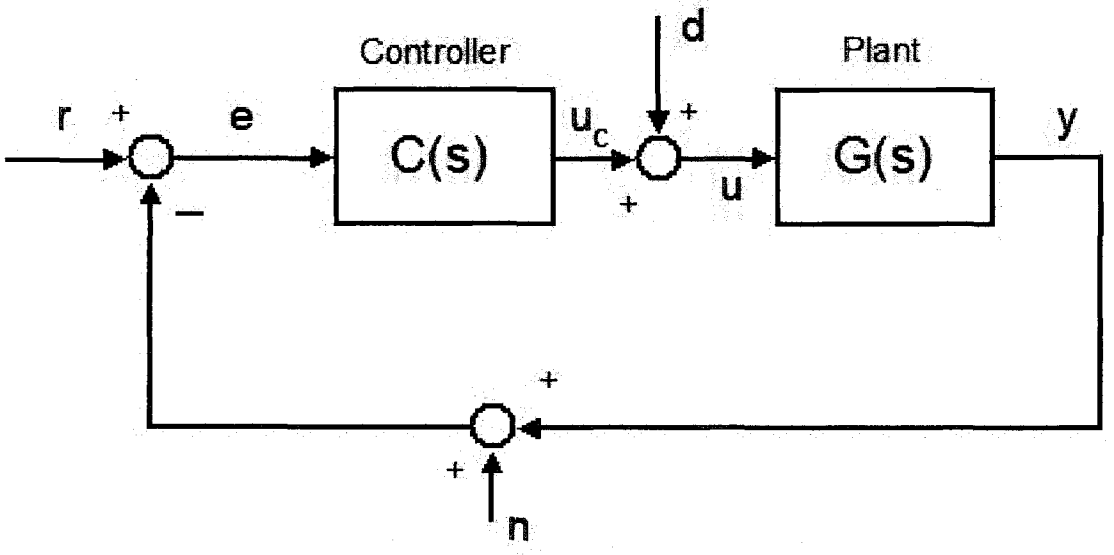


Fig. 2.33. Unitary feedback loop with compensator, $C(s)$, and plant , $G(s)$. There is output, *y*, reference command, *r*, tracking error, *e*, additive noise, *n*, input disturbance, *d*, and input from compensator, u_c.

unity feedback.

2.20.1 Output Based Sensitivity and Complimentary Sensitivity

We introduce the Sensitivity, $S(s)$, Complimentary Sensitivity, $T(s)$, and input-disturbance sensitivity, $S_d(s)$. We also introduce the loop-shape, given by $L(s)$.

$$\begin{aligned} L(s) &= G(s)C(s) \\ S(s) &= (I + G(s)C(s))^{-1} = (I + L(s))^{-1} \\ T(s) &= (I + L(s))^{-1} L(s) = L(s)(I + L(s))^{-1} \\ S_d(s) &= (I + L(s))^{-1} G(s) = G(s)(I + C(s)G(s))^{-1} \end{aligned} \tag{2.43}$$

These sensitivity transfer functions represent the following transfer functions in the Laplace domain.

$$\begin{bmatrix} y \\ e \end{bmatrix} = \begin{bmatrix} T(s) & S_d(s) & -T(s) \\ S(s) & S_d(s) & -S(s) \end{bmatrix} \begin{bmatrix} r \\ d \\ n \end{bmatrix} \tag{2.44}$$

Also, one should note that the Sensitivity and Complimentary Sensitivity should always sum to 1

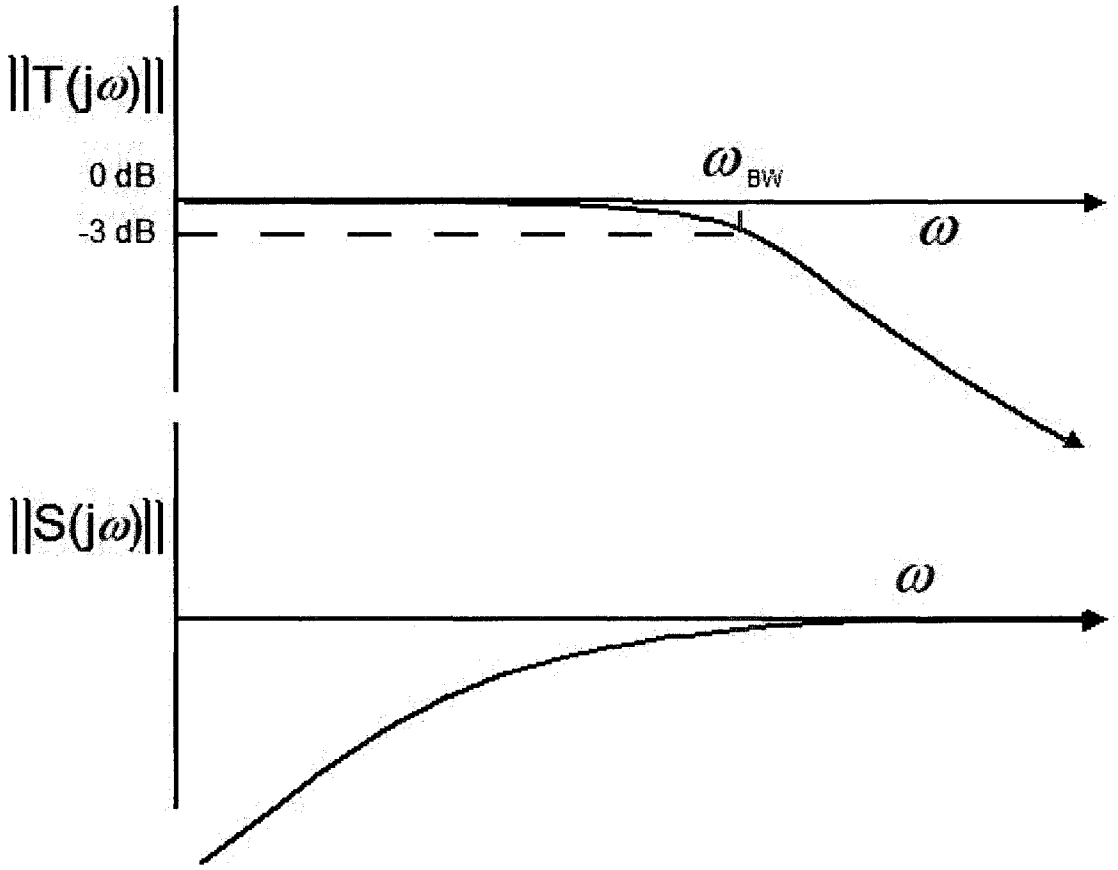


Fig. 2.34. Desirable frequency domain plots of Sensitivity, $S(s)$, and Complimentary Sensitivity, $T(s)$. Oftentimes, a critical number in determining closed loop performance is the bandwidth of the Co-Sensitivity, ω_{BW}. A high bandwidth would mean a very aggressive response with a large amount disturbance rejection, $S(j\omega) << 1 \quad \omega << \omega_{BW}$. Using high bandwidth also engenders risk of instability due to unknown high frequency dynamics such as fast time delays. A low bandwidth would mean a more gentle response with less disturbance rejection and lower tracking performance, $T(j\omega) \approx 1 \quad \omega < \omega_{BW}$ and $T(j\omega) << 1 \quad \omega >> \omega_{BW}$. Low bandwidth or small gain lessens the risk of instability due to unknown high frequency dynamics.

or the identity matrix, I.

$$T(s) + S(s) = I \tag{2.45}$$

A first-order-like ideal Sensitivity and Complimentary Sensitivity are shown in Fig. 2.34. With integral tracking, it is common to expect that $T(0) = I$, and $S(0) = 0$.

2.20.2 Input Based Sensitivity and Complimentary Sensitivity

It would be a very unusual instance where input based sensitivity functions are used for design specifications. Only for some cases of robustness analysis do input-based sensitivity functions become of interest [2.10]. Most control designs do not take these functions into account.

We introduce the Sensitivity at input, $S_I(s)$, Co-Sensitivity at input, $T_I(s)$, and noise sensitivity at input, $S_{nI}(s)$. We also introduce the loop-shape at input, given by $L_I(s)$.

$$\begin{aligned} L_I(s) &= C(s)G(s) \\ S_I(s) &= (I + C(s)G(s))^{-1} = (I + L_I(s))^{-1} \\ T_I(s) &= (I + L_I(s))^{-1} L_I(s) = L_I(s)(I + L_I(s))^{-1} \\ S_{nI}(s) &= -(I + L_I(s))^{-1} C(s) = -C(s)(I + G(s)C(s))^{-1} \end{aligned} \tag{2.46}$$

These sensitivity transfer functions represent the following transfer functions in the Laplace domain.

$$\begin{bmatrix} u_c \\ u \end{bmatrix} = \begin{bmatrix} -S_{nI}(s) & -T_I(s) & S_{nI}(s) \\ -S_{nI}(s) & S_I(s) & S_{nI}(s) \end{bmatrix} \begin{bmatrix} r \\ d \\ n \end{bmatrix} \tag{2.47}$$

Also, one should note that the Sensitivity at input and Co-Sensitivity at input should always sum to 1 or the identity matrix, I.

$$T_I(s) + S_I(s) = I \tag{2.48}$$

2.23 Q Parameterization

The Q parameterization, also called the Youla parameterization [2.11] is a convenient way of parameterizing a class of all stabilizing controllers, $C(s)$, for a given plant, $G(s)$. It can also show whether or not impossible specifications of $T(s)$ have been chosen.

The Q parameterization is introduced in eq. 2.49.

$$T(s) = G(s)Q(s) \tag{2.49}$$

Thus we can also say that the following is true in eq. 2.50.

$$\begin{aligned} Q(s) &= G^{-1}(s)T(s) = C(s)(I + G(s)C(s))^{-1} \\ S(s) &= I - G(s)Q(s) \\ Q(s) &= -S_{nI}(s) \end{aligned} \tag{2.50}$$

Now, suppose we want an expression for $C(s)$ in terms of $Q(s)$ and $G(s)$. We use the following process outlined in eq.'s 2.51 through 2.53.

$$S(s)^{-1} = (I - G(s)C(s))^{-1} = I + G(s)C(s) \tag{2.51}$$

Now we use the matrix inversion lemma to expand $S(s)^{-1}$ in eq. 2.52.

$$S(s)^{-1} = (I - G(s)Q(s))^{-1} = I + G(s)(I - Q(s)G(s))^{-1}Q(s) \tag{2.52}$$

Equating eq. 2.51 and eq. 2.52, we can see that the following is true in eq. 2.53.

$$C(s) = (I - Q(s)G(s))^{-1}Q(s) = Q(s)(I - G(s)Q(s))^{-1} \tag{2.53}$$

Thus we can use the following block diagram in Fig. 2.35 with confidence.

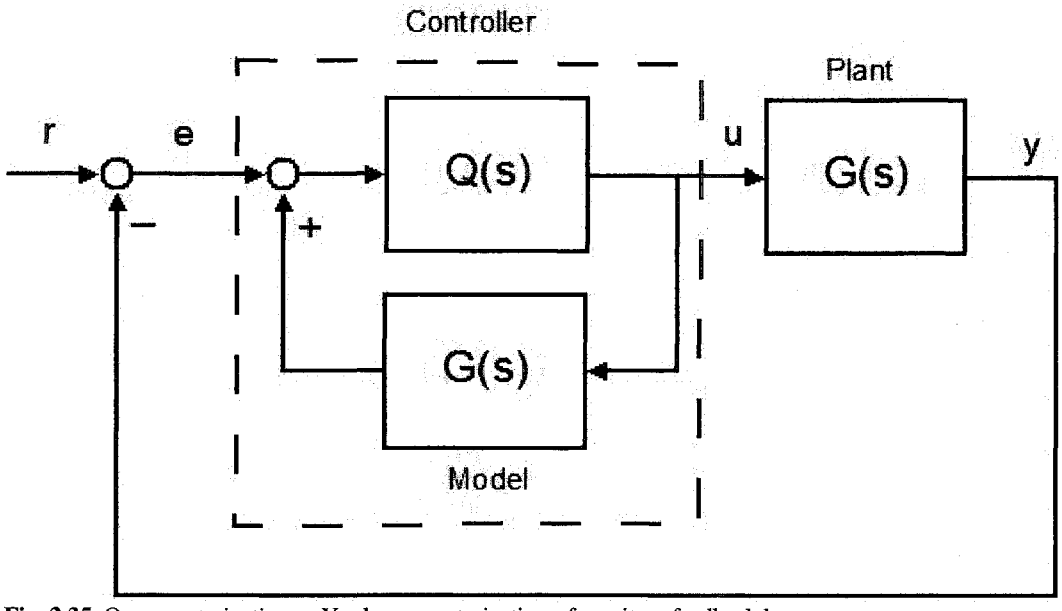


Fig. 2.35. Q parameterization or Youla parameterization of a unitary feedback loop.

If $Q(s)$ is internally stable, then the closed loop system is guaranteed to be stable. Also, if $Q(s)$ is unstable, then the closed loop system is guaranteed to be unstable. Therefore, if one generates a unreasonable specification of $T(s)$, one can show that if $Q(s)=G(s)^{-1}T(s)$ is unstable, then no stabilizing controller exists that satisfies $T(s)$.

The closed loop is stable if and only if Q(s) is stable

Choose $T(s)$

$Q(s)=G(s)^{-1}T(s)$, if $Q(s)$ is stable,
then $T(s)$ is possible with $G(s)$

$Q(s)=G(s)^{-1}T(s)$, if $Q(s)$ is unstable,
then $T(s)$ is impossible with $G(s)$
redesign $T(s)$ or $G(s)$

2.24 State Feedback and Observer Design from State Space

Suppose that an observer, L_o, has been designed and also that a state feedback gain, K_x, has

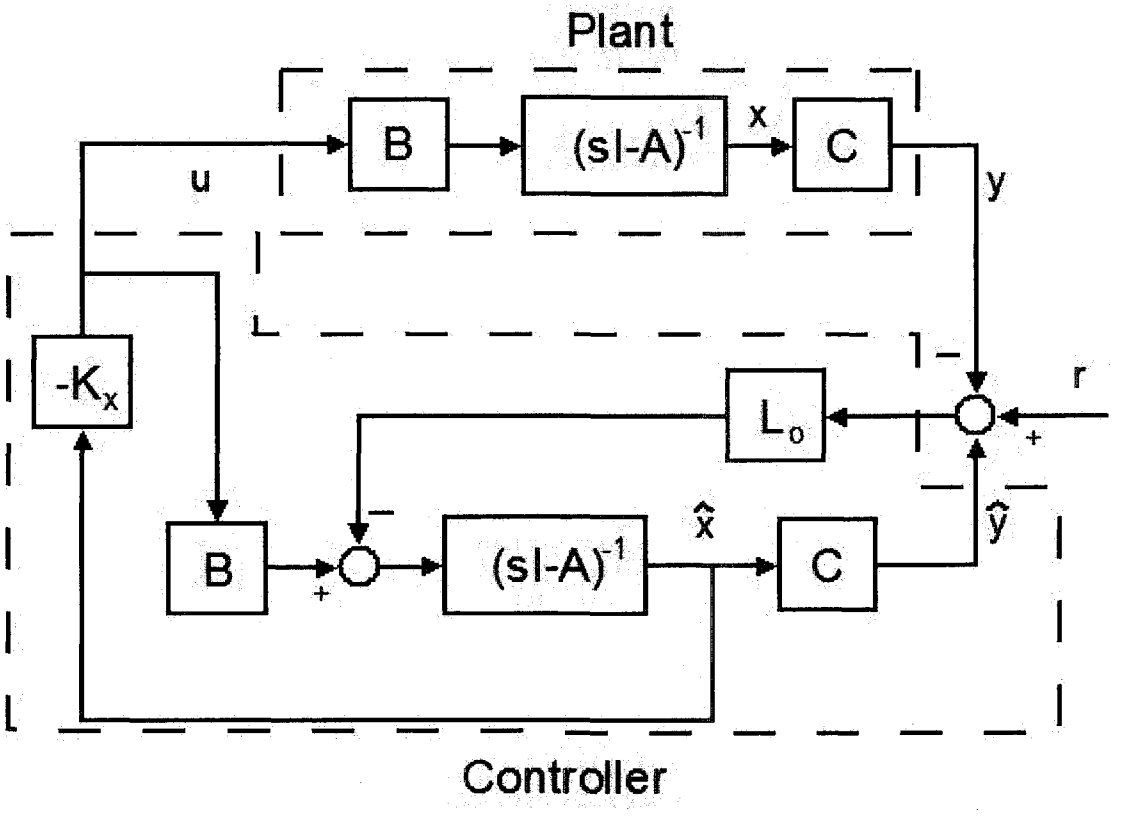


Fig. 2.36. Generic observer-state feedback control loop.

been designed. One can use the following block diagram for the controller in Fig. 2.36.

2.24.1 Separation Principle

The separation principle is the idea that one can design a state feedback gain, K_x, separately from a observer, L_o. Given a state space model which is highly accurate across a wide range of frequencies, it is possible that such a design strategy will work. The separation principle does not necessarily work very well with an uncertain model. It is possible to carelessly choose L_o and K_x in such a poor combination that very small gain and phase margins result. Nonetheless, the separation principle is mathematically sound given that the state space matrices (A, B, C) of the plant are highly accurate.

We write out the state space form of the controller and plant with the reference, r = 0, as follows in eq. 2.54. The state vector is x, and the estimated state vector is $\hat{x}$.

$$\begin{aligned} \dot{x} &= Ax - BK_x\hat{x} \\ \dot{\hat{x}} &= L_oCx + A\hat{x} - L_oC\hat{x} - BK_x\hat{x} \end{aligned} \tag{2.54}$$

Collecting like terms, we can express eq. 2.54 in a more compact state space form in eq. 2.55, where A_{cl} represents the closed loop A matrix.

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \dot{\hat{x}} \end{bmatrix} &= \begin{bmatrix} A & -BK_x \\ L_oC & A - L_oC - BK_x \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{\hat{x}} \end{bmatrix} \\ A_{cl} &= \begin{bmatrix} A & -BK_x \\ L_oC & A - L_oC - BK_x \end{bmatrix} \end{aligned} \tag{2.55}$$

The next step is to arrange the closed loop A_{cl} into the triangular form of A_{clt}, with the transformation matrix T_x.

$$\begin{aligned} A_{clt} &= T_x^{-1}A_{cl}T_x = \begin{bmatrix} I & 0 \\ I & -I \end{bmatrix} \begin{bmatrix} A & -BK_x \\ L_oC & A - L_oC - BK_x \end{bmatrix} \begin{bmatrix} I & 0 \\ I & -I \end{bmatrix} \\ &= \begin{bmatrix} A - BK_x & -BK_x \\ 0 & A - L_oC \end{bmatrix} \end{aligned} \tag{2.56}$$

Now it is accurate to equate the poles or eigenvalues of A_{cl} and A_{clt} as follows in eq. 2.57.

$$\begin{aligned} &\boldsymbol{det}(sI - A_{clt}) = \boldsymbol{det}\left(T_x^{-1}(sI - A_{cl})T_x\right) = \\ &\boldsymbol{det}\left(T_x^{-1}\right)\boldsymbol{det}(sI - A_{cl})\boldsymbol{det}(T_x) = \boldsymbol{det}(sI - A_{cl}) \end{aligned} \tag{2.57}$$

Using Schur's determinant formula, one can separate the eigenvalues or poles of A_{clt} into two distinct sets as seen in eq. 2.58.

$$\begin{aligned} &\boldsymbol{det}(sI - A_{cl}) = \boldsymbol{det}(sI - A_{clt}) = \boldsymbol{det}\left(\begin{bmatrix} sI - A + BK_x & BK_x \\ 0 & sI - A + L_oC \end{bmatrix}\right) \\ &= \boldsymbol{det}(sI - A + BK_x)\boldsymbol{det}\left(sI - A + L_oC - 0(sI - A + BK_x)^{-1}BK\right) \\ &= \boldsymbol{det}(sI - A + BK_x)\boldsymbol{det}(sI - A + L_oC) \end{aligned} \tag{2.58}$$

From eq. 2.58, there are two distinct sets of poles. One set of poles is determined by the pole assignment of the observer as: $\boldsymbol{det}(sI - A + L_oC)$. The other set of poles is determined by the pole assignment of the state feedback as: $\boldsymbol{det}(sI - A + K_xC)$. Assuming a high level accuracy of the state space, it is mathematically accurate to use the separation principle.

Table. 2.5. Design by separation principle for observer and state feedback

Design Step	
1	Design stable state feedback gain K_x so that eigenvalues are in the LHP at desirable locations. $\boldsymbol{Re}\left((\lambda[A - BK_x])_i\right) < 0 \quad \forall \quad i = 1 \ldots n$
2	Design observer, L_o, so that eigenvalues are significantly faster than the poles of the state feedback system. $\boldsymbol{Re}\left((\lambda[A - L_oC])_i\right) << \boldsymbol{Re}\left((\lambda[A - BK_x])_i\right) \quad \forall \quad i = 1 \ldots n$

2.24.2 State Feedback Design by Pole Placement for SISO systems

One can easily design a state feedback gain, K_x, that places poles in desired locations by decomposing the SISO system into the controller canonical form and exploiting some simple equalities.

$$G(s) = \frac{g_o\left(s^r + c_{r-1}s^{r-1} + c_{r-2}s^{r-2} + \ldots + c_1s + c_o\right)}{s^n + a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + \ldots + a_1s + a_o} = \frac{g_oG_N(s)}{G_D(s)} \tag{2.59}$$

We let the plant, $G(s)$ be factored into a numerator polynomial, $G_N(s)$ and a denominator polynomial $G_D(s)$, where all a_n, c_r, and g_o are scalars. The plant is considered strictly proper such that the integers, r and n will fulfill the following inequality: $r < n$.

Given that this is true in eq. 2.59, the controller canonical form can be listed in eq. 2.60.

$$A=\begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \dots & 1 \\ -a_{n-1} & -a_{n-2} & -a_{n-3} & \dots & -a_o \end{bmatrix} \quad B=\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ g_o \end{bmatrix} \tag{2.60}$$

The state feedback controller can be listed in row vector form as

$$K_x=\begin{bmatrix} k_{n-1} & k_{n-2} & \dots & k_o \end{bmatrix}.$$

Performing simple matrix multiplication, we can list the closed loop denominator, $G_{Dcl}(s)$, as follows in eq. 2.61.

$$G_{Dcl}(s)=s^n+(a_{n-1}-g_o k_{n-1})s^{n-1}+(a_{n-2}-g_o k_{n-2})s^{n-2}+\ldots+k_o a_o \tag{2.61}$$

2.24.3 Eigenstructure Assignment by State Feedback

One can design a state feedback gain, K_x, that not only places eigenvalues in certain locations, but also changes eigenvectors as well [2.12]. We suppose that we start with a set of desired closed loop eigenvalues, λ_i^{cl}, such that the following is true in eq. 2.62.

$$\lambda_i^{cl}=(\lambda[A_{cl}])_i=(\lambda[A-BK_x])_i\neq(\lambda[A])_i \quad \forall \quad i=1\ldots n \tag{2.62}$$

Furthermore we also have a set of defined closed loop eigenvectors, V^{cl}, defined as follows in eq. 2.63.

$$\begin{aligned} V^{cl}&=\begin{bmatrix} V_{\bullet,1}^{cl} & V_{\bullet,2}^{cl} & \dots & V_{\bullet,n}^{cl} \end{bmatrix} \\ (\lambda_i^{cl}I-A_{cl})V_{\bullet,i}^{cl}&=0 \quad \forall \quad i=1\ldots n \end{aligned} \tag{2.63}$$

Manipulating eq. 2.63 with K_x, we have the following expression in eq. 2.64.

$$\begin{aligned} (\lambda_i^{cl}I-A_{cl})V_{\bullet,i}^{cl}&=0 \\ (\lambda_i^{cl}I-A)V_{\bullet,i}^{cl}&=BK_xV_{\bullet,i}^{cl} \\ (\lambda_i^{cl}I-A)V_{\bullet,i}^{cl}&=BW_{\bullet,i} \quad K_x=V^{-1}W \end{aligned} \tag{2.64}$$

Now, *achievable eigenvectors* are defined as follows in eq. 2.65.

$$V_{\bullet,i}^{cl} = N_{\bullet,\bullet,i} W_{\bullet,i} \quad \forall \quad i = 1 \ldots n$$
$$N_{\bullet,\bullet,i} = \left(\lambda_i^{cl} I - A\right)^{-1} B \tag{2.65}$$

The columns of each $W_{\bullet,i}$ and $V_{\bullet,i}^{cl}$ can be real or complex numbers, $W_{\bullet,i} \in \mathbf{C}^{m\times 1}$ $V_{\bullet,i}^{cl} \in \mathbf{C}^{n\times 1}$. There is also a linear operator $N_{\bullet,\bullet,i} \in \mathbf{C}^{n\times m}$

The next step is to choose $W_{\bullet,i}$ such that $V_{\bullet,i}^{cl}$ is approximately desirable. This is shown in eq. 2.66, where two eigenvectors $V_{\bullet,1}^{cl}$ and $V_{\bullet,2}^{cl}$ are decoupled with regard to the first three states. The dc values in eq. 2.66 refer to "don't care" entries of the eigenvector.

$$V_{\bullet,1}^{cl} = \begin{bmatrix} 0 \\ dc \\ 0 \\ dc \end{bmatrix} \quad V_{\bullet,2}^{cl} = \begin{bmatrix} dc \\ 0 \\ dc \\ 0 \end{bmatrix} \quad etc. \tag{2.66}$$

The process of making individual entries of $V_{\bullet,1}^{cl}$ zero is a challenging task by hand, but an optimization subroutine can usually iteratively guess $W_{\bullet,i}$ to find an approximate solution. We decide on a masking matrix, $M_{\bullet,\bullet,i}^{ask}$ in the following manner in eq. 2.67.

$$M_{\bullet,\bullet,1}^{ask} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad M_{\bullet,\bullet,2}^{ask} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad etc. \tag{2.67}$$

Next we want to minimize the following expression in eq. 2.68.

$$W_{\bullet,i} = \min_{W} \left(\left\| M_{\bullet,\bullet,i}^{ask} N_{\bullet,\bullet,i} W_{\bullet,i} \right\| \Big/ \left\| W_{\bullet,i} \right\| \right) \tag{2.68}$$

A MATLAB script shown below gives an example of this can be solved.

Example 2.13 – MATLAB code for eigenstructure assignment

```
%Flight Condition: NT-33 mach 0.4 SL
%
Uo=447; bank=0/180*pi; Ysda=0; Lda=12.6;
cbank=cos(bank); Yv=-.181; Nda=.165; Ysdr=.0503;
Lbeta=-8.02; Nbeta=2.71; Ldr=1.57; Ndr=-3.50;
Lp=-2.15; Np=-0.0512;
Lr=0.320; Nr=-0.291;
%
Alat=[Yv 0 -1 32.2/Uo*cbank;Lbeta Lp Lr 0;...
    Nbeta Np Nr 0;0 1 0 0];
Blat=[Ysda Ysdr;Lda Ldr;Nda Ndr;0 0];
Clat=[0 0 0 1;0 0 1 0;Uo*Yv 0 0 32.2*cbank];
Dlat=[zeros(3,2);Ysda*Uo  Uo*Ysdr;zeros(1,2)];
%
A=Alat;B=Blat;
%state vector is [beta p r phi]
%input vector is [da dr]
%
%Solve roll placement, want r = 0 and beta = 0
Mask=[1 0 0 0;0 0 1 0];
roll_root=-4;  %closed loop eigenvalue on real axis
Nroll=inv(roll_root*eye(4)-A)*B;
%initial guess solution
Xo=null(Nroll(1,:));
%optimization to get approximate roll vector
X=fminsearch(@(X) ...
    norm(Mask*Nroll*X)/norm(Nroll*X),Xo);
Vcl_roll=Nroll*X/norm(X);
W_roll=X/norm(X);
%Solve dutch roll placement, want p = 0 and phi = 0
Mask=[0 1 0 0;0 0 0 1];
dutch_root=-sin(.4)*1.74+cos(.4)*1.74j;
Ndutch=inv(dutch_root*eye(4)-A)*B;
%initial guess solution
Xo=[1 1]';
%optimization to get approximate dutch roll
X=fminsearch(@(X) ...
    norm(Mask*Ndutch*X)/norm(Ndutch*X),Xo);
Vcl_dutch=Ndutch*X/norm(X);
W_dutch=X/norm(X);
%Solve spiral root placement, don't care about eigenvector
spiral_root=-.01;
Nspiral=inv(spiral_root*eye(4)-A)*B;
%random guess
Xo=[1 1]';
Vcl_spiral=Nspiral*Xo/norm(Xo);
W_spiral=Xo/norm(Xo);
%Final solution
Kx=[W_roll W_spiral W_dutch conj(W_dutch)]*...
    inv([Vcl_roll Vcl_spiral Vcl_dutch conj(Vcl_dutch)]);
Kx=real(Kx);
[Vcl,Dcl]=eig(A-B*Kx); %get eigenvalues and eigenvectors
```

Remarks:

1. The code above generates the following state feedback gains:

$$K_x = \begin{bmatrix} 0.3029 & -0.2288 & -0.2958 & -0.3423 \\ 0.6666 & -0.1066 & 0.2340 & -0.3396 \end{bmatrix}$$, which are strange. This is because the transient specifications are strange. Most often, adding more damping to the dutch roll mode naturally adds more roll into closed loop dutch roll motion. Assigning less damping to the dutch roll mode will make the gains more reasonable.

2. Eigenstructure assignment has the potential for creating poorly conditioned eigenvectors, choose eigenvectors and eigenvalues wisely.

3. Alternatively, one could use the MATLAB command "place( )" which conveniently chooses eigenvectors for which the condition number of the eigenvectors is minimized [2.13].

2.24.4 Observer Design

One can design an observer in the same way that one designs a state feedback gain matrix. Only one uses the transpose of the A matrix and the transpose of the C matrix.

2.24.5 Optimal Control Techniques

An optimal state feedback such as the Linear Quadratic Regulator (LQR) can be used together with an observer such as a Kalman filter. This design strategy is often called a Linear Quadratic Gaussian (LQG) design. One may have problems if the observer design is too slow. A Loop Transfer Recovery (LTR) basically recovers the original robustness of the state feedback by using a very fast observer. The LTR design procedure is explicitly guaranteed to work only for plants which are square and minimum phase [2.14]. Even with a working optimal control law, these control systems can produce large amounts of overshoot over 50% for a simple step response. Typically, feed-forward elements are needed to counteract this overshoot. Details on designing these feed-forward elements can be found in Ref. [2.15].

2.25 Coprime Factorizations

Coprime factorizations, also called matrix fraction decompositions, are sometimes useful in control theory and controller design [2.16,2.17,2.3]. However, they are not absolutely essential.

2.25.1 Right Coprime Factorizations

Suppose there is a transfer function , $G(s)$. We want to find a decomposition such that $G(s)=G_N(s)G_D(s)^{-1}$, where $G_N(s)$ is a "numerator", and $G_D(s)$ would be a "denominator".

It is shown that if the transfer function $G(s)$ is stabilizable (all unobservable modes are stable), then one can obtain a right coprime factorization by the use of a state-feedback regulator [2.16]. The result in [2.16] only covers the strictly proper case. This section will add the proper case.

Assume first that $G(s)=C(sI-A)^{-1}B+D$ of dimension $p\times m$, with p outputs and m inputs. Also assume that there is a state feedback law with external input, w, such that $u=-Kx+w$. Then with this feedback law with external input, the state space can be written below.

$$\begin{aligned}\dot{x}&=(A-BK)x+Bw\\ y&=(C-DK)x+Dw\\ w&=u+Kx\end{aligned} \tag{2.69}$$

We now write the following transfer function from w to u as $u(s)=G_D(s)w(s)$. From the state space representation in eq. 2.69, $G_D(s)$ can be written as in eq. 2.70.

$$\begin{aligned}u(s)&=G_D(s)w(s)\\ G_D(s)&=-K(sI-A+BK)^{-1}B+I\end{aligned} \tag{2.70}$$

To obtain the inverse dynamics, one would use the state space representation below to obtain $G_D(s)^{-1}$.

$$\begin{aligned}\dot{x}&=Ax+Bu\\ w&=Kx+u\\ G_D(s)^{-1}&=K(sI-A)^{-1}B+I\end{aligned} \tag{2.71}$$

Now, we need to get the following transfer function from w to y as $y(s)=G_N(s)w(s)$. From the state space representation in eq. 2.69, $G_N(s)$ can be written as in eq. 2.72.

$$G_N(s) = (C - DK)(sI - A + BK)^{-1} B + D \tag{2.72}$$

Finally, to obtain the coprime form results in a simple algebraic substitution below in eq. 2.73.

$$\begin{aligned} w(s) &= G_D(s)^{-1} u(s) \\ y(s) &= G(s)u(s) = G_N(s)w(s) = G_N(s)G_D(s)^{-1} u(s) \\ \therefore G(s) &= G_N(s)G_D(s)^{-1} \end{aligned} \tag{2.73}$$

It may seem that this method may fail if $C = DK$. However, it works fine and one arrives at exactly $G(s) = C(sI - A)^{-1} B + D$ even when $C = DK$.

2.25.2 Left Coprime Factorizations

Suppose there is a transfer function, $G(s)$. We want to find a decomposition such that $G(s) = \overline{G}_D(s)^{-1} \overline{G}_N(s)$, where $\overline{G}_N(s)$ is a "numerator", and $\overline{G}_D(s)$ would be a "denominator".

It is shown that if the transfer function $G(s)$ is detectable (all unstable modes are observable), then one can obtain a right coprime factorization by the use of a state-feedback regulator [2.16].

The result in [2.16] only covers the strictly proper case. This section will add the proper case.

Assume first that $G(s) = C(sI - A)^{-1} B + D$ of dimension $p \times m$, with p outputs and m inputs. Also assume that there is a observer with inputs, u, and y, such that eq. 2.74 is fulfilled, where v would be the observer error.

$$\begin{aligned} \dot{\hat{x}} &= (A - L_o C)\hat{x} + (B - L_o D)u + L_o y \\ v &= y - C\hat{x} - Du \end{aligned} \tag{2.74}$$

Assuming zero initial conditions, we now write the following transfer functions in the Laplace domain from u and y to v as $v(s) = \overline{G}_D(s)y(s) + J(s)u(s)$. From the state space representation in eq. 2.74, $G_D(s)$ can be written as in eq. 2.75.

$$\begin{aligned} v(s) &= \overline{G}_D(s)y(s) + J(s)u(s) \\ J(s) &= -C(sI - A + L_o C)^{-1}(B - L_o D) - D \\ \overline{G}_D(s) &= -C(sI - A + L_o C)^{-1} L_o + I \end{aligned} \tag{2.75}$$

We now propose that $u(s)=G^{-1}(s)y(s)=\left[\overline{G}_D(s)^{-1}\overline{G}_N(s)\right]^{-1}y(s)=\overline{G}_N(s)^{-1}\overline{G}_D(s)y(s)$.

If the observer had an initial estimate error of 0 and was designed with a perfect model, then $v(s)=0$.

$$\begin{aligned} v(s)&=\overline{G}_D y(s)+J(s)\overline{G}_N(s)^{-1}\overline{G}_D(s)y(s)=0 \\ \overline{G}_D(s)y(s)&=-J(s)\overline{G}_N(s)^{-1}\overline{G}_D(s)y(s) \\ \therefore \overline{G}_N(s)&=-J(s)=C(sI-A+L_oC)^{-1}(B-L_oD)+D \end{aligned} \tag{2.76}$$

Table 2.6 summarizes the left and right coprime factorizations based upon an observer design L_o and state feedback K.

Table 2.6 Summary of left and right coprime factorizations

Right Coprime Factorization $G(s)=C(sI-A)^{-1}B+D=G_N(s)G_D(s)^{-1}$	Choose any stable state feedback K
$G_D(s)$	$-K(sI-A+BK)^{-1}B+I$
$G_N(s)$	$(C-DK)(sI-A+BK)^{-1}B+D$
Left Coprime Factorization $G(s)=C(sI-A)^{-1}B+D=\overline{G}_D(s)^{-1}\overline{G}_N(s)$	Choose any stable observer L_o
$\overline{G}_D(s)$	$-C(sI-A+L_oC)^{-1}L_o+I$
$\overline{G}_N(s)$	$C(sI-A+L_oC)^{-1}(B-L_oD)+D$

2.25.5 Asymptotic Tracking of Constant References

Suppose there is a closed loop shape as in Fig. 2.34, $L(s)$. We want to investigate whether or not it can track a constant reference. We will use a right coprime factorization of the loop shape $L(s)$ and look at the steady state. An alternative interpretation of this concept can be found in Ref. [2.5].

As below in eq. 2.77, a fundamental assumption is that the loop shape can be factored into $L(s)=L'_N(s)\frac{1}{s}L_D(s)^{-1}$, where the left coprime numerator has an integrator, $L_N(s)=L'_N(s)\frac{1}{s}$. Note that in order for this to be possible $L(s)$ will always be square with dimension $p\times p$. Also, we assume that $\boldsymbol{det}\left(L'_N(0)L_D(0)^{-1}\right)\neq 0$, i.e. there is no transmission zero of $L(s)$ at $s=0$.

$$
\begin{aligned}
&e(s) = S(s)r(s) \quad r(s) = r_o \\
&e(0) = S(0)r_o \\
&L(s) = L_N'(s)\frac{1}{s}L_D(s)^{-1} \quad S(s) = \left(I + L_N'(s)\frac{1}{s}L_D(s)^{-1}\right)^{-1}
\end{aligned}
\tag{2.77}
$$

In eq. 2.78, we expand out the sensitivity function, $S(s)$, using the matrix inversion lemma. We notice from eq. 2.78 that the steady state error, $e(0)$, is zero.

$$
\begin{aligned}
&S(s) = I - L_N'(s)\left(sI + L_D(s)^{-1}L_N'(s)\right)^{-1}L_D(s)^{-1} \\
&S(0) = I - L_N'(0)\left(L_D(0)^{-1}L_N'(0)\right)^{-1}L_D(0)^{-1} = I - I = 0 \\
&\therefore e(0) = 0
\end{aligned}
\tag{2.78}
$$

Remarks:

1. Suppose that $L(s) = \frac{1}{s}\begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}$. No coprime factorization of the form $L(s) \neq L_N'(s)\frac{1}{s}L_D(s)^{-1}$ can exist. Odds are that the plant, $G(s)$, is functionally uncontrollable.

2. The same procedure can be followed through for ramp references where, $r(s) = \frac{1}{s}r_o$ and $L(s) = \overline{L}_D(s)^{-1}\frac{1}{s^2}$. From the matrix inversion lemma, $S(s) = \overline{L}_D s^2 - \overline{L}_D s^2\left(I + \overline{L}_D s^2\right)^{-1}\overline{L}_D s^2$. Thus, $e(s) = \left(\overline{L}_D s - \overline{L}_D s^2\left(I + \overline{L}_D s^2\right)\overline{L}_D s\right)r_o$ and $e(0) = 0$.

3. With integral tracking the dc gain of the Co-Sensitivity is identity, $T(0) = I$. $T(s) + S(s) = I$, $T(0) + S(0) = T(0) + 0 = T(0) = I$.

4. In many cases, when $G(0) \in \Re^{m \times m}$ and $\boldsymbol{det}(G(0)) \neq 0$, using m integrators when designing $C(s)$ will fulfill asymptotic tracking.

2.25.3 Closed Loop Stability from Coprime Factorizations

Note the following relation is true below.

$$
zeros \quad of \quad (I + G(s)C(s)) \quad \Leftrightarrow \quad poles \quad of \quad (I + G(s)C(s))^{-1}
$$

Suppose there are right coprime factorizations for the controller, $C(s)=C_N(s)C_D(s)^{-1}$, and left coprime factorizations for the plant, $G(s)=\overline{G}_D(s)^{-1}\overline{G}_N(s)$. We know that the closed loop will be stable if and only if the sensitivity $S(s)$ is stable.

$$\begin{aligned} S(s)&=(I+G(s)C(s))^{-1} \\ S(s)^{-1}&=I+\overline{G}_D(s)^{-1}\overline{G}_N(s)C_N(s)C_D(s)^{-1} \\ \overline{G}_D(s)S(s)^{-1}C_D(s)&=\overline{G}_D(s)C_D(s)+\overline{G}_N(s)C_N(s) \\ S(s)&=C_D(s)\left(\overline{G}_D(s)C_D(s)+\overline{G}_N(s)C_N(s)\right)^{-1}\overline{G}_D(s) \end{aligned} \tag{2.79}$$

Provided that $C_D(s)$ and $\overline{G}_D(s)$ are stable, then one can check to make sure that the roots of $\boldsymbol{det}\left(\overline{G}_D(s)C_D(s)+\overline{G}_N(s)C_N(s)\right)=0$ are stable.

We can get the reverse statement for input sensitivity, $S_I(s)=(I+C(s)G(s))^{-1}$. Suppose there are left coprime factorizations for the controller, $C(s)=\overline{C}_D(s)^{-1}\overline{C}_N(s)$, and right coprime factorizations for the plant, $G(s)=G_N(s)G_D(s)^{-1}$.

$$\begin{aligned} S_I(s)&=(I+C(s)G(s))^{-1} \\ S_I(s)^{-1}&=I+\overline{C}_D(s)^{-1}\overline{C}_N(s)G_N(s)G_D(s)^{-1} \\ \overline{C}_D(s)S_I(s)^{-1}G_D(s)&=\overline{C}_D(s)G_D(s)+\overline{C}_N(s)G_N(s) \\ S_I(s)&=G_D(s)\left(\overline{C}_D(s)G_D(s)+\overline{C}_N(s)G_N(s)\right)^{-1}\overline{C}_D(s) \end{aligned} \tag{2.80}$$

Provided that $\overline{C}_D(s)$ and $G_D(s)$ are stable, then one can check to make sure that the roots of $\boldsymbol{det}\left(\overline{C}_D(s)G_D(s)+\overline{C}_N(s)G_N(s)\right)=0$ are stable.

2.25.5 Observer-State Feedback Controller Representation

Assuming that a transfer function has left and right coprime factorizations as;

$Y(s)=\overline{Y}_D(s)^{-1}\overline{Y}_N(s)=Y_N(s)Y_D(s)^{-1}$, there exist coprime factorizations

$W(s)=\overline{W}_D(s)^{-1}\overline{W}_N(s)=W_N(s)W_D(s)$ that fulfill the following *generalized Bezout identity* which is shown in eq. 2.81. The author has not had much luck using these Bezout identities for control synthesis of unstable plants, however they might work for stable and minimum phase plants.

$$\begin{bmatrix} \overline{W}_D(s) & \overline{W}_N(s) \\ -\overline{Y}_N(s) & \overline{Y}_D(s) \end{bmatrix} \begin{bmatrix} Y_N(s) & -W_N(s) \\ Y_D(s) & W_D(s) \end{bmatrix} = \begin{bmatrix} Y_D(s) & -W_N(s) \\ Y_N(s) & W_D(s) \end{bmatrix} \begin{bmatrix} \overline{W}_D(s) & \overline{W}_N(s) \\ -\overline{Y}_N(s) & \overline{Y}_D(s) \end{bmatrix} = I \tag{2.81}$$

It can be shown [2.16] that if one substitutes in $G(s)$ and $C(s)$, one arrives at the following expressions in eq. 2.82. Once again, these identities are not essential for control synthesis.

$$\begin{aligned} \overline{G}_D(s)C_D(s) + \overline{G}_N(s)C_N(s) &= I \\ \overline{C}_D(s)G_D(s) + \overline{C}_N(s)G_N(s) &= I \end{aligned} \tag{2.82}$$

For a strictly proper plant, $G(s) = C(sI - A)^{-1}B$, an acceptable solution to eq. 2.82 defines the controller, $C(s)$, as shown in eq. 2.83.

$$\begin{aligned} C_N(s) &= K(sI - A + BK)^{-1}L_o \\ C_D(s) &= I + C(sI - A + BK)^{-1}L_o \\ \overline{C}_N(s) &= K(sI - A + L_oC)^{-1}L_o \\ \overline{C}_D(s) &= I + K(sI - A + L_oC)^{-1}B \end{aligned} \tag{2.83}$$

Collecting these terms together from eq. 2.83 gives the following realization for the controller in eq. 2.84.

$$C(s) = K(sI - A + BK + L_oC)^{-1}L_o \tag{2.84}$$

Remarks:

1. From eq. 2.84, we can see that *any* stabilizing and strictly proper controller has a realization which consists of a full-state observer, L_o, and state feedback design, K.

2. Given $C(s)$, the full-state observer, L_o, and the state feedback design, K are not unique.

   Consider a plant with, $A = 0 \quad B = 1.99 \quad C = 1$. The controller will be $C(s) = \dfrac{1}{(s+20)}$.

   Acceptable realizations for the controller can be achieved with $L_o = 19.9$ and $K = 1/19.9$ or $L_o = 0.1$ and $K = 10$. The observer and state feedback should exist in pairs, though it may be difficult to figure out how many pairs may exist for large systems.

3. Many control design techniques are available. Some may claim that one design technique is flawed and that another design technique is superior. There is some unfairness associated with these comparisons because there should be multiple ways to design the same controller. One person may use a bode plot and another person may use a state feedback-observer method. The two people could independently arrive at the same controller. Given the same tracking requirements, one should ideally arrive at very similar controllers independent of the design method. However, it is fair to compare which method is more easily designed to meet control specifications. If control specifications occur in the frequency domain, a natural choice would be to use frequency domain methods such as Nyquist and Bode plots. However, in the event that one wants to minimize a given cost function in the time domain, an optimal control method would be easier to use instead of a frequency domain method.

2.26 The Small Gain Theorem

Suppose that the plant, $G(s)$, is known to be stable. Also, assume that the controller, $C(s)$, is stable as well. If the following is true of the loop-shape below, then the closed feedback loop is guaranteed to be stable [2.18].

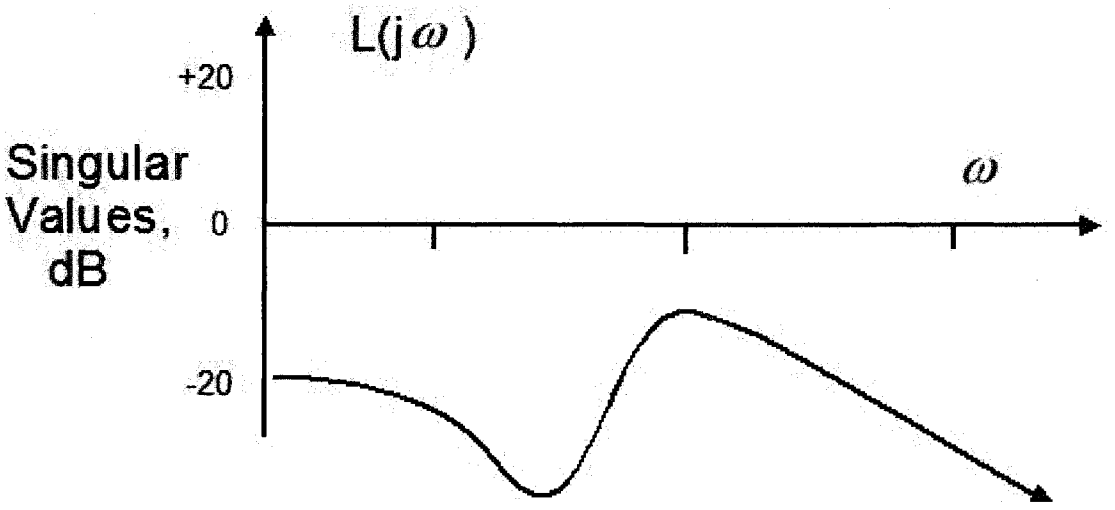


Fig. 2.36. If all of the singular values of the loopshape are below 1 and the plant is open loop stable, then closed loop stability is assured.

$$\begin{aligned} &\textit{if} \quad L(s) \quad \textit{is} \quad \textit{stable} \quad \textit{and} \\ &\textit{if} \quad \|L(j\omega)\| < 1 \quad \forall \quad \omega \\ &\textit{then} \quad \textit{closed} \quad \textit{loop} \quad \textit{is} \quad \textit{stable} \end{aligned}$$

The small gain theorem is valid with any matrix norm, though the infinity norm, which is the maximum singular value across all frequencies, is most commonly used.

The problem with the singular value is that it is a sufficient but not necessary condition for stability. One cannot use the singular value to show instability. One *cannot* use the small gain theorem to arrive at a gain margin for MIMO systems.

$$\begin{aligned} &\textit{if} \quad \|L(j\omega)\| > 1 \quad \textit{anywhere} \\ &\textit{then} \quad \textit{closed} \quad \textit{loop} \quad \textit{stability} \quad \textit{is} \quad \textit{unknown}, \\ &\textit{must} \quad \textit{check} \quad \textit{eigenvalues} \end{aligned}$$

2.27 Parameterizations for Robustness Investigation

Oftentimes, the plant, $G(s)$, has some uncertainty associated with it. It is often of interest to make sure that the closed loop is stable under this uncertainty. Additionally, one may also want to look into the performance behavior under uncertainty. This section covers several parametrizations for investigation into the robustness of closed loop systems.

We suppose that a loop shape, $L(s)$, has been achieved by a design of a compensator, $C(s)$ with unity feedback.

2.27.1 Closed Loop Stability with Uncertainty

Suppose that a closed loop with uncertainty can be decomposed into the $M\Delta$ positive feedback structure of Fig. 2.37. The parameterization of M varies upon the type of uncertainty under consideration.

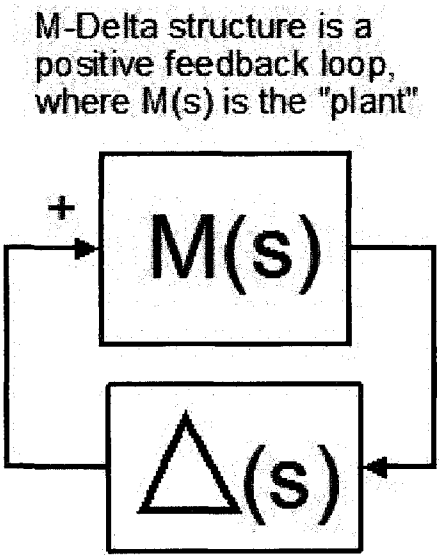


Fig. 2.37. Closed loop stability of the total system can be investigated with the $M\Delta$ positive feedback structure. M should contain the nominal plant dynamics, $G(s)$, and the controller $C(s)$. $\Delta(s)$ contains the dynamics which are considered to be uncertain.

M should contain the nominal plant dynamics, $G(s)$, and the controller $C(s)$. Delta, $\Delta(s)$, contains the dynamics which are considered to be uncertain.

We want to know whether or not $M\Delta$ positive feedback structure will be stable under uncertainty. The system will be stable under uncertainty if the poles of the sensitivity function, $(I - M(s)\Delta(s))^{-1}$, are stable. It is shown in section 2.20 that the zeros of a biproper (square and invertible) transfer function are the poles of the inverse of that transfer function.

$$zeros \quad of \quad (I - M(s)\Delta(s)) \quad \Leftrightarrow \quad poles \quad of \quad (I - M(s)\Delta(s))^{-1}$$

One thing which can be done is to collect known data for delta, $\Delta(s)$, and compute the poles of the $M\Delta$ sensitivity or the zeros of the $M\Delta$ return difference.

Another thing which can be done is to get a worst case gain margin by multiple loop closures. One would want to design $\Delta(s)$ to destabilize the system.

Structured Singular Value (SSV) analysis, often called μ analysis, is devoted to finding $\Delta(s)$ such that the following condition in eq. 2.85 is met.

$$\boldsymbol{det}(I - M(s)\Delta(s)) = 0 \tag{2.85}$$

If eq. 2.85 were true then the closed loop system would be neutrally stable, at a particular value of s. The computationally difficult problem statement for μ analysis is to find $\Delta(s)$ such that eq. 2.85 is satisfied.

$$find \quad \Delta(s) \quad and \quad s \quad s.t. \quad \boldsymbol{det}(I - M(s)\Delta(s)) = 0$$

This is a numerically difficult task if $\Delta(s)$ could be anything at all. Therefore, the assumption for stability analysis is that $\Delta(s)$ has a prescribed structure. Frequently, $\Delta(s)$ is accurately considered to be purely diagonal for input and output uncertainty. When $\Delta(s)$ is diagonal, one can use decentralized techniques to arrive at independent gain margins. This is available with the MATLAB loopmargin( ) subroutine.

However, if one wants to consider interactions between multiple loops, one can consider the example of such a procedure is listed below.

If $\Delta(s)=\begin{bmatrix} \Delta_{11}(s) & 0 & \cdots & 0 \\ 0 & \Delta_{22}(s) & \cdots & \vdots \\ \vdots & \cdots & \ddots & 0 \\ 0 & \cdots & 0 & \Delta_{mm}(s) \end{bmatrix}$:

Step 1.1: Look at gain margin of $M_{11}(s)\Delta_{11}(s)$. Close loop of $M_{11}(s)\Delta_{11}(s)$ with $\Delta_{11}(s)$ at its upper limit with positive feedback.

Step 1.2: Look at gain margin of second loop closure, $M_{22}(s)\Delta_{22}(s)\big|_{M11\to\Delta11}$. Close loop of $M_{22}(s)\Delta_{22}(s)\big|_{M11\to\Delta11}$ with $\Delta_{22}(s)$ at its upper limit with positive feedback.

Step 1.3: Look at gain margin of third loop closure, $M_{33}(s)\Delta_{33}(s)\big|_{M11\to\Delta11,M22\to\Delta22}$. Close loop of $M_{33}(s)\Delta_{33}(s)\big|_{M11\to\Delta11,M22\to\Delta22}$ with $\Delta_{33}(s)$ at its upper limit with positive feedback.

Etc....

Step 1.m: Look at gain margin of mth loop closure,

$M_{mm}(s)\Delta_{mm}(s)\big|_{M11\to\Delta11,M22\to\Delta22,\ldots M(m-1)(m-1)\to\Delta(m-1)(m-1)}$. Close loop of

$M_{mm}(s)\Delta_{mm}(s)\big|_{M11\to\Delta11,M22\to\Delta22,\ldots M(m-1)(m-1)\to\Delta(m-1)(m-1)}$ with $\Delta_{33}(s)$ at its upper limit with positive feedback.

One should note that the first loop closure of $M_{11}(s)\Delta_{11}(s)$ was done at its upper limit, there is the unlikely possibility that the first loop closure acted had a stabilizing effect for

the second loop closure. Therefore, the analysis must be repeated with the first loop closed with a low uncertainty.

Step 2.1: Look at gain margin of $M_{11}(s)\Delta_{11}(s)$. Close loop of $M_{11}(s)\Delta_{11}(s)$ with $\Delta_{11}(s)$ at its lower limit with positive feedback.

Step 2.2: Look at gain margin of second loop closure, $M_{22}(s)\Delta_{22}(s)\big|_{M11\to\Delta11}$. Close loop of $M_{22}(s)\Delta_{22}(s)\big|_{M11\to\Delta11}$ with $\Delta_{22}(s)$ at its upper limit with positive feedback.

Step 2.3: Look at gain margin of third loop closure, $M_{33}(s)\Delta_{33}(s)\big|_{M11\to\Delta11,M22\to\Delta22}$. Close loop of $M_{33}(s)\Delta_{33}(s)\big|_{M11\to\Delta11,M22\to\Delta22}$ with $\Delta_{33}(s)$ at its upper limit with positive feedback.

Etc....

Step 2.m: Look at gain margin of mth loop closure,

$M_{mm}(s)\Delta_{mm}(s)\big|_{M11\to\Delta11,M22\to\Delta22,\ldots M(m-1)(m-1)\to\Delta(m-1)(m-1)}$. Close loop of

$M_{33}(s)\Delta_{33}(s)\big|_{M11\to\Delta11,M22\to\Delta22}$ with $\Delta_{33}(s)$ at its upper limit with positive feedback.

Now one would want to consider the second loop closure, what effect would it have if it a small value of $\Delta_{22}(s)$ were used? One would have to repeat the loop closure process again. This procedure can become very exhaustive. Taking only upper and lower limits of the $\Delta_{ii}(s)$ values would result in a total of 2^m permutations of loop closures!

Finding true stability margins for large, complicated and uncertain MIMO systems require exponentially large and complicated calculations. The fact that the number of calculations has an exponential distribution should act as a strong deterrent for designing large, complicated and highly uncertain MIMO systems

Arbitrary norm bounded uncertainties can be shown to be stable via the small gain theorem. However, one should note that this often leads to overly conservative estimations of stability margins.

small gain theorem: if $\bar{\sigma}(M(s)\Delta(s))<1 \Rightarrow$ **stable**
if $\bar{\sigma}(M(s)\Delta(s))>1 \Rightarrow$ **stability unknown**

Alternatively, one can *very cautiously* use μ analysis software to perform the extensive search to find upper bounds on allowable $\Delta(s)$.

2.27.2 Additive Uncertainty

We adopt the following notation for the parameterization of additive uncertainty.

$$G^p(s): \textbf{perturbed plant}$$
$$G(s): \textbf{nominal plant}$$
$$\Delta_A(s) \equiv G^p(s) - G(s): \textbf{additive uncertainty}$$

Although it is not absolutely necessary, one can decompose $\Delta_A(s)$ into a gain matrix, w_A, and a complex valued portion, $U_A(s)$, $\Delta_A(s) = w_A U_A(s)$. Hence, the $M\Delta$ structure for additive uncertainty can be written in eq. 2.86.

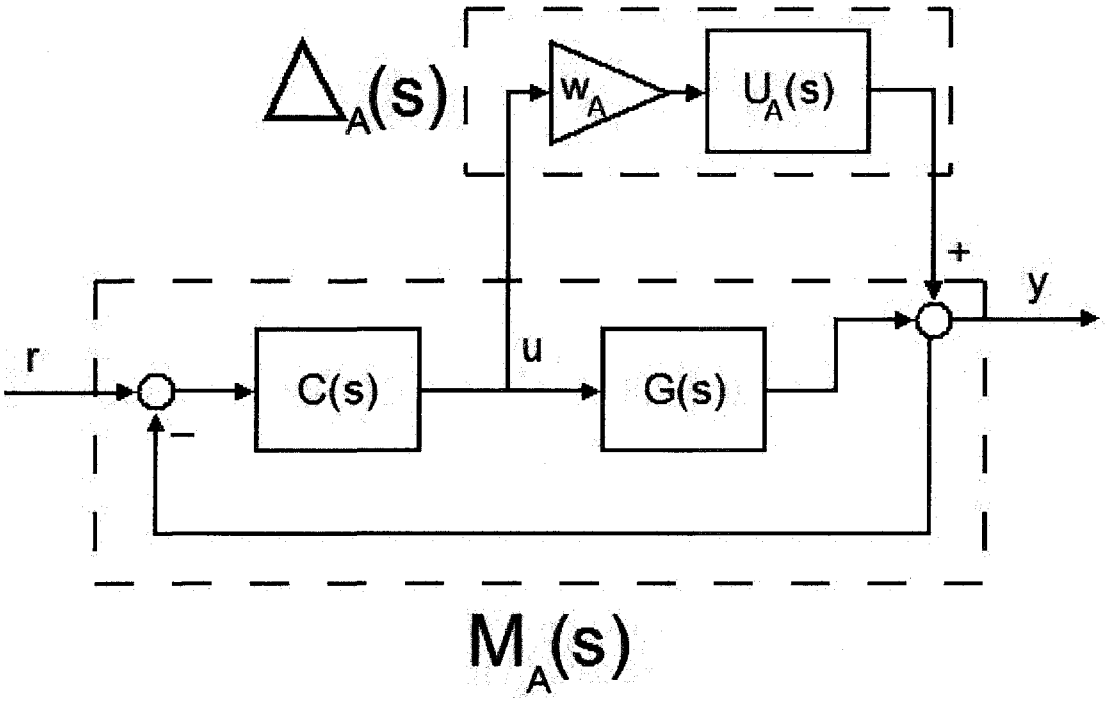


Fig. 2.37. Unitary feedback loop with compensator, $C(s)$, and plant , $G(s)$. There is output, *y*, reference command, *r*, tracking error, *e*, additive noise, *n*, input disturbance, *d*, and input from compensator, u_c.

$$\begin{aligned} M_A(s) &= -(I + C(s)G(s))^{-1} C(s) = -C(s)(I + G(s)C(s))^{-1} = S_{nI}(s) \\ \Delta_A(s) &= G^p(s) - G(s) \end{aligned} \tag{2.86}$$

With this parameterization, one can see whether elements of $\Delta_A(s)$ cause instability of the closed loop system.

Example 2.14 - Additive Uncertainty

Consider the following plant, $G(s)$, and perturbed plants, $G^{p1}(s)$ and $G^{p2}(s)$, below.

$$G(s)=\begin{bmatrix} \dfrac{(s+8)}{4(s+1)(s+2)} & 0 \\ 0 & \dfrac{2(s+10)}{10(s+1)(s+2)} \end{bmatrix}$$

$$G^{p1}(s)=\begin{bmatrix} \dfrac{(s+8)}{4(s+1)(s+2)} & \dfrac{(s+8)}{10(s+1)(s+2)} \\ \dfrac{10(s+10)}{(s+1)(s+2)} & \dfrac{2(s+10)}{10(s+1)(s+2)} \end{bmatrix}$$

$$G^{p2}(s)=\begin{bmatrix} \dfrac{(s+8)}{4(s+1)(s+2)} & \dfrac{-(s+8)}{10(s+1)(s+2)} \\ \dfrac{-10(s+10)}{(s+1)(s+2)} & \dfrac{2(s+10)}{10(s+1)(s+2)} \end{bmatrix}$$

Thus, we can say that additive uncertainties $\Delta_{Ap1}(s)$ and $\Delta_{Ap2}(s)$ are as follows.

$$\Delta_{Ap1}(s)=\begin{bmatrix} 0 & \dfrac{(s+8)}{10(s+1)(s+2)} \\ \dfrac{10(s+10)}{(s+1)(s+2)} & 0 \end{bmatrix} \quad \Delta_{Ap2}(s)=\begin{bmatrix} 0 & \dfrac{-(s+8)}{10(s+1)(s+2)} \\ \dfrac{-10(s+10)}{(s+1)(s+2)} & 0 \end{bmatrix}$$

Suppose that the following decentralized controller was designed for $G(s)$ as follows. This controller was designed to get good disturbance rejection from the first output to the second output. This was achieved by making the first loop have a bandwidth of 1 rad./sec. where the second loop had a bandwidth of 5 rad./sec.

$$C(s)=\begin{bmatrix} \dfrac{4(s+1)(s+2)}{s(s+8)} & 0 \\ 0 & \dfrac{25(s+1)(s+2)}{s(s+10)} \end{bmatrix}$$

$M_A(s)$ can be calculated below.

$$M_A(s)=\begin{bmatrix}\dfrac{-4(s+2)}{(s+8)} & 0\\ 0 & \dfrac{-25(s+2)(s+1)}{(s+10)(s+5)}\end{bmatrix}$$

From direct computation of $(sI-M(s)\Delta(s))^{-1}$, it can be easily shown that the closed loop is unstable with the above uncertainty. But suppose we want to find out a kind of off diagonal gain margin. We decide to take the following procedure to discover allowable uncertainties.

Step 1.1: Look at gain margin of $M_{A12}(s)\Delta_{A21}(s)$ or $M_{A22}(s)\Delta_{A21}(s)$, this should be infinite since the $M\Delta$ is exactly triangular at this point. Close loop of $M_{A12}(s)\Delta_{A21}(s)$ with $\Delta_{A21}(s)$ at its upper limit with positive feedback.

$$\Delta_{A21}(s)=\frac{10(s+10)}{(s+1)(s+2)}$$

$$M_A(s)\big|_{y1\to u2}=\begin{bmatrix}\dfrac{-4(s+2)}{(s+8)} & 0\\ \dfrac{1000(s+2)}{(s+8)(s+5)} & \dfrac{-25(s+2)(s+1)}{(s+5)(s+10)}\end{bmatrix} \quad \text{with} \quad \Delta_{A21}(s)$$

Step 1.2: Look at gain margin of $\left(M_A(s)\big|_{y1\to u2}\right)_{21}\Delta_{A12}(s)$ in positive feedback. Basically, we take $\Delta_{A12}(s)$ and close the loop of $y2\to u1$ to arrive at $M_A(s)\big|_{y1\to u2,y2\to u1}$. Then we see what kind of allowable values of $\Delta_{A12}(s)$ would make this loop closure stable. So, this means generating a root locus from the following equation below.

$$\Delta_{A12}(s)=\frac{w_{A12}(s+8)}{(s+1)(s+2)}$$

$$1-\left(M_A(s)\big|_{y1\to u2}\right)_{21}\Delta_{A12}(s)=0$$

$$\text{Root locus: } 1-w_{A12}\frac{1000(s+2)}{(s+8)(s+5)}\frac{(s+8)}{(s+1)(s+2)}=0$$

From this we can clearly see that a positive value of w_{A12} would lead to instability. However if w_{A12} were negative we would get an infinite gain margin. Thus we can say the following below.

$$\text{With} \quad \Delta_{A21}(s) = \frac{10(s+10)}{(s+1)(s+2)} \quad \text{and} \quad \Delta_{A12}(s) = \frac{w_{A12}(s+8)}{(s+1)(s+2)}$$

$$-\infty < w_{A12} < 0.005 \quad \text{for} \quad \text{stability}$$

The MATLAB code to achieve this loop-by-loop analysis is shown below.

Example 2.14 – MATLAB code for additive M-delta analysis

```
G=[zpk(-8,[-1 -2],0.25) 0;
    0 zpk(-10,[-1 -2],0.2)];
C=[zpk([-1 -2],[-8 0],4) 0;
    0 zpk([-1 -2],[-10 0],5*5)];
MA=minreal(-C*inv(eye(2)+G*C));
deltap1=[0 zpk(-8,[-1 -2],0.1);
    zpk(-10,[-1 -2],10) 0];
deltap2=[0 zpk(-8,[-1 -2],-0.1);
    zpk(-10,[-1 -2],-10) 0];

Mdeltap1=minreal(feedback(MA,deltap1,+1)); %unstable
Mdeltap2=minreal(feedback(MA,deltap2,+1)); %unstable

%step 1.1 close first loop in positive feedback
MAstep11=feedback(MA,[0 0;deltap1(2,1) 0],+1);

%step 1.2 think about allowable gains for delta(1,2)

%w_A12 is positive
rlocus(-MAstep11(2,1)*zpk(-8,[-1 -2],1))

%w_A21 is negative
%rlocus(MAstep11(2,1)*zpk(-8,[-1 -2],1))
```

Performing similar steps with $\Delta_{A21}(s) = \frac{-10(s+10)}{(s+1)(s+2)}$ gives the following result.

$$\text{With} \quad \Delta_{A21}(s) = \frac{-10(s+10)}{(s+1)(s+2)} \quad \text{and} \quad \Delta_{A12}(s) = \frac{w_{A12}(s+8)}{(s+1)(s+2)}$$

$$-0.005 < w_{A12} < \infty \quad \text{for} \quad \text{stability}$$

In this case, it appears as though the sign of the off diagonal terms $G_{21}^{p}(s)$ and $G_{12}^{p}(s)$ should not share similar sign.

One should be careful not to try to draw too many conclusions or generalize from this example. This plant is stable and is of minimum phase because all transmission zeros are stable. Non-minimum phase plants with unstable zeros are very likely to exhibit a very different behavior.

2.27.3 Multiplicative Output Uncertainty

We adopt the following notation for the parameterization for multiplicative output uncertainty.

$$G^p(s): \textbf{perturbed plant}$$
$$G(s): \textbf{nominal plant}$$
$$\Delta_o(s) \equiv G^p(s)G(s)^{-1} - I : \textbf{multiplicative output uncertainty}$$

Although it is not absolutely necessary, one can decompose $\Delta_o(s)$ into a gain matrix, w_o, and a complex valued portion, $U_o(s)$, $\Delta_o(s) = w_o U_o(s)$. Hence, the $M\Delta$ structure for additive uncertainty can be written in eq. 2.87.

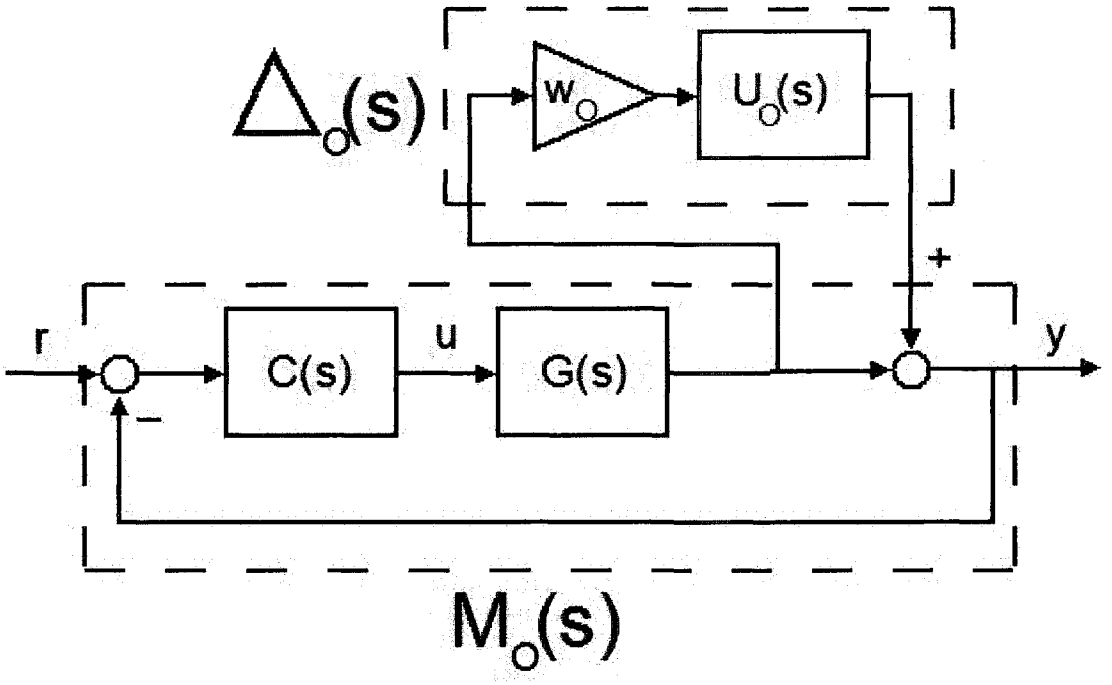


Fig. 2.38. Unitary feedback loop with compensator, $C(s)$, and plant , $G(s)$. There is output, *y*, reference command, *r*, tracking error, *e*, additive noise, *n*, input disturbance, *d*, and input from compensator, u_c.

$$M_o(s) = -(I + G(s)C(s))^{-1} G(s)C(s) = -T(s)$$
$$\Delta_o(s) = G^p(s)G(s)^{-1} - I \qquad (2.87)$$
$$\textbf{often} \quad \Delta_o(s) = \textbf{diag}(\Delta_{11}(s) \quad \Delta_{22}(s) \quad \ldots \quad \Delta_{pp}(s))$$

With this parameterization, one can see whether elements of $\Delta_o(s)$ cause instability of the closed loop system.

Example 2.15 - Multiplicative Output Uncertainty

Consider the following plant, $G(s)$, and the following $\Delta_o(s)$ below.

$$G(s) = \begin{bmatrix} \dfrac{(s+8)}{4(s+1)(s+2)} & 0 \\ 0 & \dfrac{2(s+10)}{10(s+1)(s+2)} \end{bmatrix}$$

$$\Delta_o(s) = \begin{bmatrix} \dfrac{-5s(s-40)}{(s+100)(s+40)} & 0 \\ 0 & \dfrac{-5s(s-40)}{(s+100)(s+40)} \end{bmatrix}$$

The idea behind $\Delta_o(s)$ is that it is a representation of sensor delay and high frequency uncertainty. Suppose that the following decentralized controller, $C(s)$, was designed below.

$$C(s) = \begin{bmatrix} \dfrac{4(s+1)(s+2)}{s(s+8)} & 0 \\ 0 & \dfrac{25(s+1)(s+2)}{s(s+10)} \end{bmatrix}$$

Due the high frequency nature of this sensor uncertainty, we can use the small gain theorem to justify stability of the closed loop.

$$\text{small gain theorem: if } \bar{\sigma}(-T(s)\Delta_o(s)) < 1 \Rightarrow \text{stable}$$

Shown to the below is a singular value plot of $\bar{\sigma}(-T(j\omega)\Delta_o(j\omega))$ to show that stability is not a threat to the closed loop.

The singular value plot to the below can be generated with the MATLAB subroutine "sigma( )".

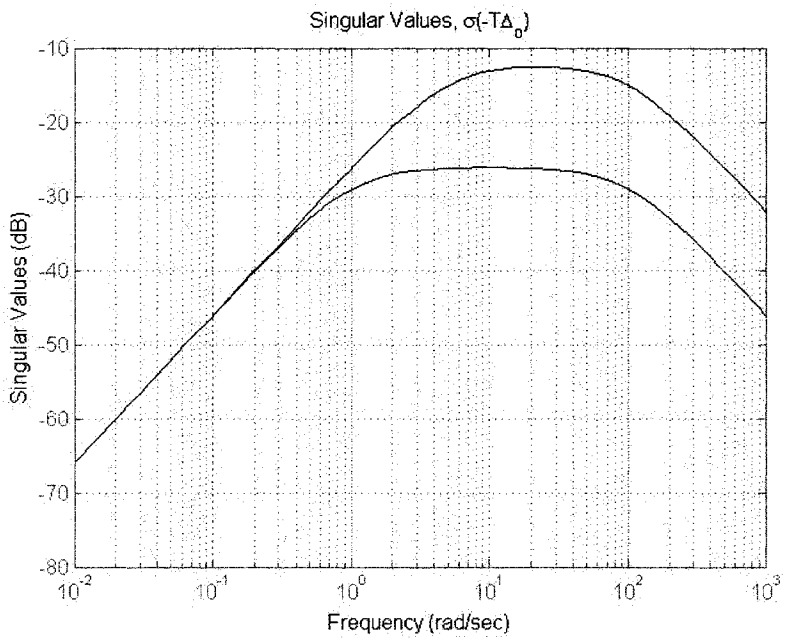


Fig. 2.39. We can correctly conclude that the closed loop system is safely stable with the output uncertainty because the singular values of $\sigma(-T(j\omega)\Delta_o(j\omega))$ are both less than one.

One should note that if $\overline{\sigma}(-T(j\omega)\Delta_o(j\omega))>1$, then one could not conclude that the closed loop is unstable.

2.27.4 Multiplicative Input Uncertainty

We adopt the following notation for the parameterization for multiplicative input uncertainty.

$$G^p(s): \textbf{perturbed plant}$$
$$G(s): \textbf{nominal plant}$$
$$\Delta_I(s) \equiv G(s)^{-1}G^p(s) - I : \textbf{multiplicative output uncertainty}$$

Although it is not absolutely necessary, one can decompose $\Delta_I(s)$ into a gain matrix, w_o, and a complex valued portion, $U_I(s)$, $\Delta_I(s) = w_I U_I(s)$. Hence, the $M\Delta$ structure for additive uncertainty can be written in eq. 2.88.

$$\begin{aligned} &M_I(s) = -(I + C(s)G(s))^{-1}C(s)G(s) = -T_I(s) \\ &\Delta_I(s) = G(s)^{-1}G^p(s) - I \\ &\textbf{often} \quad \Delta_I(s) = \textbf{diag}(\Delta_{11}(s) \quad \Delta_{22}(s) \quad \ldots \quad \Delta_{mm}(s)) \end{aligned} \tag{2.88}$$

With this parameterization, one can see whether elements of $\Delta_I(s)$ cause instability of the closed loop system.

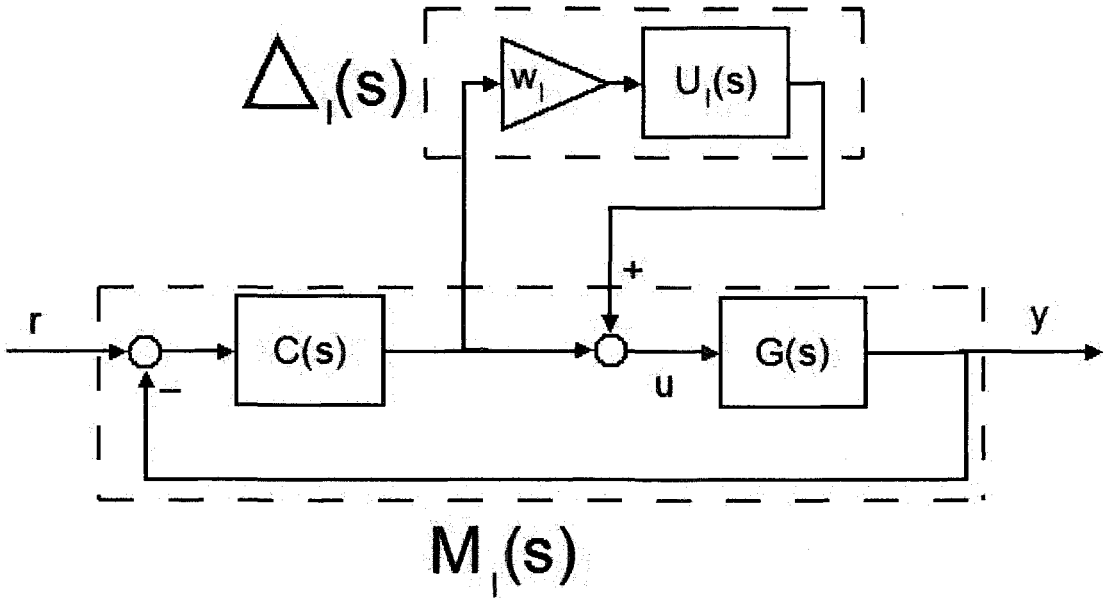


Fig. 2.40. Unitary feedback loop with compensator, $C(s)$, and plant , $G(s)$. There is output, *y*, reference command, *r*, tracking error, *e*, additive noise, *n*, input disturbance, *d*, and input from compensator, u_c.

Example 2.16 - Multiplicative Input Uncertainty

We will compute individual gain margins with multiple loop closures of a diagonal $\Delta_I(s)$ for a decentralized lateral control system. The aircraft of concern will be the Navion aircraft at a speed of Mach 0.158 at Sea-Level (SL) [2.19]. Though the control design is decentralized, it is unconventional in that integral action is pursued in both the bank angle channel and the yaw rate channel.

The decentralized controller is listed below.

$$C(s)=\begin{bmatrix} \dfrac{7(s+0.009)(s+8.4)}{s(s+40)} & 0 \\ 0 & \dfrac{-0.5(s^2+1.2s+3.97)}{s(s+0.27)} \end{bmatrix}$$

A first order pade' approximation of $\dfrac{-(s-20)}{(s+20)}$ will be used to represent a 0.1 second time delay in actuator dynamics. Allowable upper limits of w_{11} and w_{22} will be analyzed with $\Delta_I(s)$ listed below.

$$\Delta_I(s)=\begin{bmatrix} w_{11} & 0 \\ 0 & w_{22} \end{bmatrix}$$

The dynamics for the Navion with inputs of aileron and rudder and outputs of ϕ and r are listed below.

$$G(s)=\begin{bmatrix} 28.984(s^2+0.998s+4.562) & 2.193(s-7.9)(s+3.8) \\ -0.2218(s+54.08)(s-1.25)(s+1.54) & -4.6(s+8.61)(s^2+0.1413s+0.2939) \end{bmatrix}\cdots$$
$$\frac{-(s-20)}{(s+0.00876)(s+8.435)(s^2+0.9735s+5.688)(s+20)}$$

The process for finding upper limits on w_{11} and w_{22} for stability are shown below. This process is based upon multiple loop closures.

1. Set $w_{11}=0$, calculate $w_{22} \le w_{22max}$ based upon gain margin of $[T_I(s)\Delta_I(s)]_{2,2}$
2. Set $w_{22}=0$, calculate $w_{11} \le w_{11max}$ based upon gain margin of $[T_I(s)\Delta_I(s)]_{1,1}$
3. Close the $[M\Delta]_{1,1}$ loop with a gain of w_{11}. Observe the gain margin of the $[M\Delta]_{2,2}$ loop; the gain margin corresponds to the maximum w_{22} for the particular value of w_{11}. Repeat this process by progressively increasing w_{11} inside the interval $0 \le w_{11} \le w_{11max}$.
4. Close the $[M\Delta]_{2,2}$ loop with a gain of w_{22}. Observe the gain margin of the $[M\Delta]_{1,1}$ loop; the gain margin corresponds to the maximum w_{11} for the particular value of w_{22}. Repeat this process by progressively increasing w_{22} inside the interval $0 \le w_{22} \le w_{22max}$.

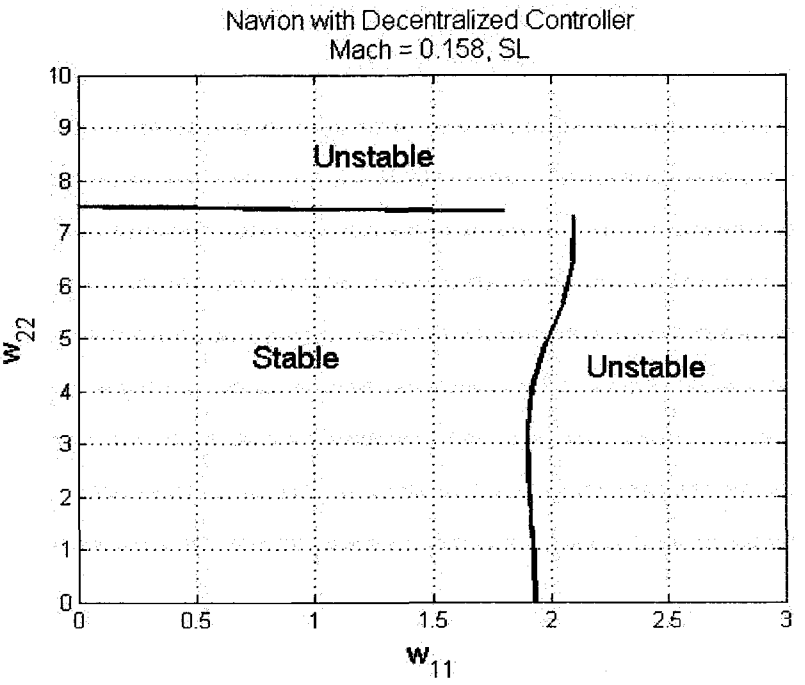


Fig. 2.41. Notice that that this shape is mostly box-like, which indicates a very small interaction between loops

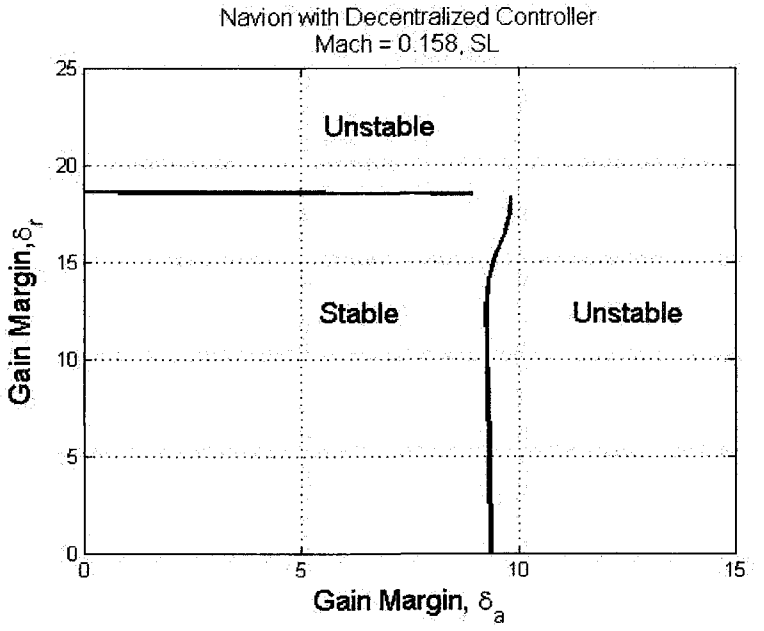


Fig. 2.42. The independent SISO gain margins are largely unaffected by compound variations in aileron and rudder effectiveness, units are in dB.

Results from this analysis are shown in Fig. 2.41 and Fig. 2.42. Notice that the region of stability looks very much like a box. This means that the individual gain margins of each loop are a good indicator of the stability margins of the total system. Similar results have been reported in Ref. [2.20]. Most fixed wing aircraft in most flight conditions exhibit diagonally dominant dynamics at frequencies above 1 to 4 rad./sec. There are some exceptions such as the HL-10 and F-104 [2.21]. However, controllers for these aircraft can be designed with cross-feed elements such as an Aileron-to-Rudder Interconnect (ARI) or Rudder-to-Aileron Interconnect (RAI) to reduce the interaction between loops at higher frequencies above 1 to 4 rad./sec.

Example 2.16 – MATLAB code for Input Uncertainty Example

```
clc
clear all

%lateral Navion at SL mach 0.158
%Calculate ff precomp with phi and r as outputs

Uo=176;
bank=0/180*pi;
cbank=cos(bank);
Yv=-.2543;
Lbeta=-15.982;
Nbeta=4.495;
Lp=-8.402;
Np=-0.3498;
Lr=2.193;
Nr=-0.7605;

Ysda=0;
Lda=28.984;
Nda=-0.2218;
Ysdr=.0708;
Ldr=2.193;
Ndr=-4.597;

Alat=[Yv 0 -1 32.2/Uo*cbank;Lbeta Lp Lr 0;
    Nbeta Np Nr 0;0 1 0 0];
Blat=[Ysda Ysdr;Lda Ldr;Nda Ndr;0 0];
Clat=[0 0 0 1;0 0 1 0;Uo*Yv 0 -Uo*0 32.2*cbank];
Dlat=[zeros(2,2);Ysda*Uo    Ysdr*Uo];

syslat=ss(Alat,Blat,[Clat(1,:);Clat(2,:)],zeros(2));

C11=zpk([-0.009 -8.4],[0 -40],7);
C22=zpk([-0.6+1.9j -0.6-1.9j],[0 -0.27],-0.5);

tdelay=zpk([20],[-20],-1);
Ti=feedback([C11 0;0 C22]*syslat*tdelay,eye(2));

w11=linspace(0,1.8,10); %step 1
w22=linspace(0,7.3,10); %step 2
for k=1:10
    temp=feedback(Ti,[w11(k) 0;0 0]); %close Mdelta11 loop
    w22m(k) = margin(temp(2,2)); %get w22 margin
    g11dB(k)=20*log10(w11(k)+1); %convert to gain margin
    g22mdB(k)=20*log10(w22m(k)+1);
    temp=feedback(Ti,[0 0;0 w22(k)]); %close Mdelta22 loop
    w11m(k) = margin(temp(1,1)); %get w11 margin
    g11mdB(k)=20*log10(w11m(k)+1); %convert to gain margin
    g22dB(k)=20*log10(w22(k)+1);
end
```

2.28 Bumpless Controller Transfer

Before discussing bumpless controller transfer, we must establish the concept of a bump.

Imagine that there are two controllers which are just gains $k_1 = 5$ and $k_2 = 6$. Suppose that we want to switch controllers at t = 3 sec. The tracking error is e(3) = 1 at 3 seconds. Initially, we were using controller 1, $k_1 = 5$, therefore the input is u(3$^-$) = 5. However, if we switch to the second controller $k_2 = 6$ at t = 3 sec, then u(3$^+$) = 6. Thus, there would be a bump with a magnitude bump = u(3$^+$) - u(3$^-$) = 6-5 = 1.

If the change in controllers is small and the bump is small, then one would not need to do a bumpless controller transfer. However, if the bump is large enough to be problematic, then one might consider implementing a bumpless transfer.

There are many possible ways to do a bumpless controller transfer. In our previous example, one could simply use a control law of $u(t) = k_2 e(t) + (\text{bump}) e^{-0.5(t-3)}$ for $t > 3$ sec. One could also slowly adjust the gain from 5 to 6.

2.28.1 Hanus Bumpless Controller Transfer for Biproper Controllers

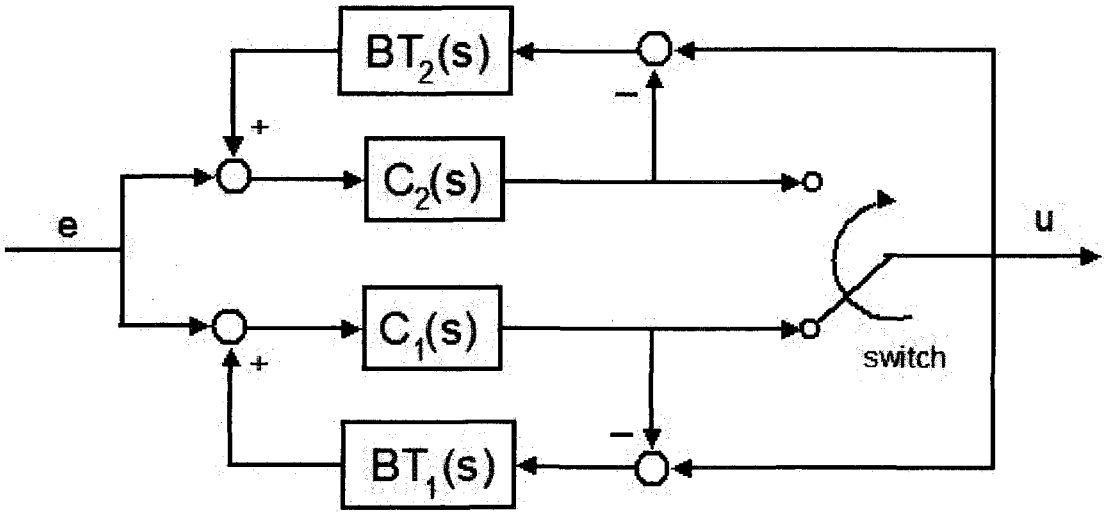


Fig. 2.43. Generic observer-state feedback control loop.

For square and invertible controllers, i.e. biproper controllers, the Hanus bumpless transfer [2.22] technique can be quite useful. It is primarily concerned with controllers which are stable and have stable transmission zeros (i.e. minimum phase). Many controllers fit this description.

The idea is that we design a bumpless transfer function, $BT_1(s)$, for controller 1, $C_1(s)$, and a bumpless transfer function $BT_2(s)$ for $C_2(s)$. Notice that the bumpless transfer functions are in a feedback loop with the controllers. Therefore, we design these bumpless transfer functions as though they are feedback controllers for controllers.

To design a bumpless transfer function, one thinks of it as a controller of a controller. As long as the controller is stable and stably invertible, then we can simply invert the controller to get our bumpless transfer function. However, because this is a feedback loop, things can become destabilized by delay, so we add a filter, $F(s)$ as well.

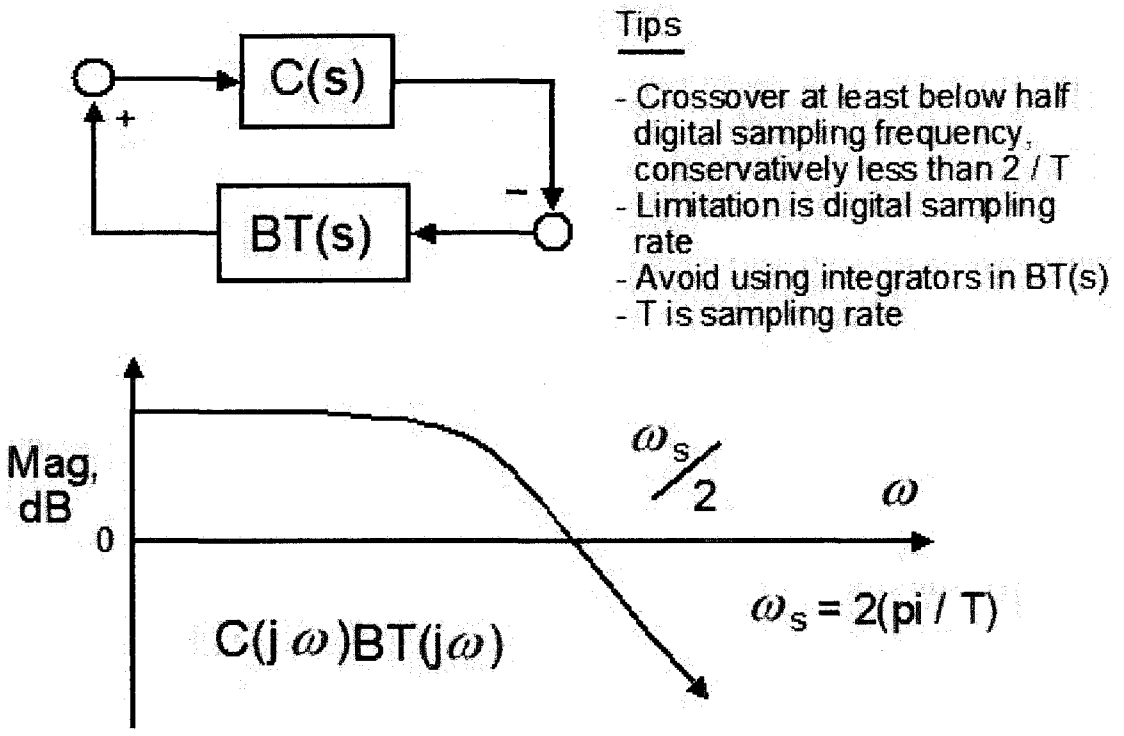


Fig. 2.44. Loop-shapes for designing bumpless controller transfers

$$BT(s) = C(s)^{-1} F(s) \tag{2.89}$$

An example of this would be a PI controller, $C(s) = \frac{2(s+1)}{s}$, and then our bumpless transfer function would be $BT(s) = \frac{s}{2(s+1)} \frac{8}{(s+1)}$, where $F(s) = \frac{8}{(s+1)}$. This bumpless transfer function would achieve a 80 to 90% to bumps in u(t) slower than 1 rad./sec. We hope that this is an adequate

reduction. Using $F(s)=\frac{8}{(s+1)}$ is set conservatively for digital sampling rates of 0.1 seconds, which is a popular digital sampling rate for aircraft control.

2.29 Practical Limitations of SISO Control

Control is limited. The open loop dynamics of the plant mostly dictate what is reasonably achievable and what is not.

2.29.1 Stabilizability

If you performed the kalman decomposition, described in section 2.11, and all of the hidden modes were stable, then the plant is stabilizable. All of the unstable modes must be completely controllable and completely observable.

Although mathematically precise, the concept of stabilizability from state space descriptions can be misleading on what is reasonably achievable for a physical system.

2.29.2 Strongly Stabilizable Plants

A strongly stabilizable plant is a plant which does not require an unstable controller to stabilize it. The following examples illustrate the idea of strong stabilizability.

$$G(s)=\frac{-(s-10)}{(s+10)(s-11)}$$

This plant has a minimal state space realization given below.

$$A=\begin{bmatrix}0 & 1\\ 110 & 1\end{bmatrix}\quad B=\begin{bmatrix}0\\ 1\end{bmatrix}\quad C=\begin{bmatrix}10 & -1\end{bmatrix}$$

There are no hidden modes of this system. The unstable mode at s = 11 is controllable and observable, therefore it is stabilizable. However, it is not strongly stabilizable.

We can come up with the following state feedback gain matrix, $K_x=\begin{bmatrix}111 & 3\end{bmatrix}$ to place poles a pair of poles at well damped locations. We can also use an observer $L_o=\begin{bmatrix}-95 & -1034\end{bmatrix}^T$. The observer based controller is listed below.

$$C(s)=K_x(sI-A+BK_x+L_oC)L_o=\frac{-13647(s+9.935)}{(s^2-82s+1.234x10^4)}$$

Notice that there is a pair of unstable poles in the denominator. Therefore, this is not a strongly stabilizable plant.

Now suppose we changed the plant so that it was less unstable. And the unstable pole was at lower frequency than the unstable zero.

$$G(s)=\frac{-(s-10)}{(s+10)(s-9)}$$

We can use the following controller below to stabilize the plant. This controller is just a PD controller with a high frequency filter.

$$C(s)=\frac{100(s+10)}{(s+100)}$$

It is important not to have more zeros than poles in a controller. Otherwise, the controller would be a very high frequency amplifier for noise and high frequency dynamics.

2.29.3 Robustly Stabilizable Plants

There really is no precise definition of a robustly stabilizable plant, but we define a robustly stabilizable plant as a plant that can be stabilized by a single controller when considering a reasonable amount of modeling uncertainty or variation of the plant parameters.

Suppose that the plant varied between $G^p(s)$ and $G(s)$.

$$G^p(s)=\frac{-0.25(s-10)}{(s+10)(s-9)} \quad G(s)=\frac{-(s-10)}{(s+10)(s-9)}$$

The controller, $C(s)=\frac{100(s+10)}{(s+100)}$, stabilizes $G(s)$, but not $G^p(s)$. The controller, $C^p(s)=\frac{400(s+10)}{(s+100)}$, stabilizes $G^p(s)$, but not $G(s)$.

There is pretty much no controller that strongly stabilizes both $G(s)$, but not $G^p(s)$ so we say that this plant is not robustly stabilizable.

Suppose that the plant were made to be even less unstable so that the unstable pole was at a much lower frequency than previously.

$$G(s)=\frac{-(s-10)}{(s+10)(s-1)} \quad G^p(s)=\frac{-0.25(s-10)}{(s+10)(s-1)}$$

Now, this is a much more reasonable plant, where the unstable zero is ten times faster than the unstable pole. We find that the following controller, $C(s)=6$ can stabilize both $G(s)$, but not $G^p(s)$. Therefore, we say that the redesigned plant with an unstable pole at s = 1 and the unstable zero which was ten times faster at s = 10 is robustly stabilizable.

For plants with unstable poles and zeros *on the real axis*, it is helpful to have unstable poles 6 to 10 times slower than any unstable zeros.

This is not at all a requirement for plants with unstable poles and zeros *near the* $j\omega$ *axis*. Consider the following plant below.

$$G(s)=\frac{(s^2-0.1s+1)}{(s^2-0.2s+9)}\frac{10}{(s+10)} \quad G^p(s)=\frac{(s^2-0.1s+1)}{(s^2-0.2s+9)}\frac{5}{(s+10)}$$

There is an oscillatory pair of zeros at 1 rad./sec. and a pair of oscillatory poles at 3 rad./sec. Yet, it is very easy to get a robustly stabilizing controller for this plant. Consider the following integral controller, $C(s)=\frac{3}{s}$. This stabilizes both plants. With unstable poles and zeros *near the* $j\omega$ *axis*, it is reasonable to have oscillatory poles faster than oscillatory zeros.

2.29.4 Performance Limitations

Not only are there limitations upon what can practically be stabilized, but there are also limitations upon how quickly things can move. Consider a plant with a time delay, like a faucet. From the time you turn the knob until the time water starts coming out, there is a very short time delay. Water has to travel from the pipes to the faucet head. This is a transport delay. Additionally, there is a sensor delay. Humans have delays in awareness of their environment. It is not possible for a person to recognize that too much water is flowing until they hear it, see it or touch it. Also, it is not possible to make the water move any faster by turning the faucet knob.

2.29.5 Performance Limitations for Minimum Phase and Stable Plants of Relative Degree 2

Bode's sensitivity integral relationship is shown below. This only applies to stable plants of relative degree 2 with stable zeros (i.e. minimum phase).

$$\int_0^\infty ln|S(j\omega)|d\omega = 0 \tag{2.90}$$

This means that a reduction in sensitivity at one frequency will be met by an increase in sensitivity at another frequency. Gunter Stein [2.23] related this to digging a ditch. If you start digging in one area, then the dirt must pile up somewhere else. This concept was called "conservation of sensitivity dirt". Another analogy is that of a waterbed [2.10]. If you sit down on one side of the bed, water is displaced and the other side rises up.

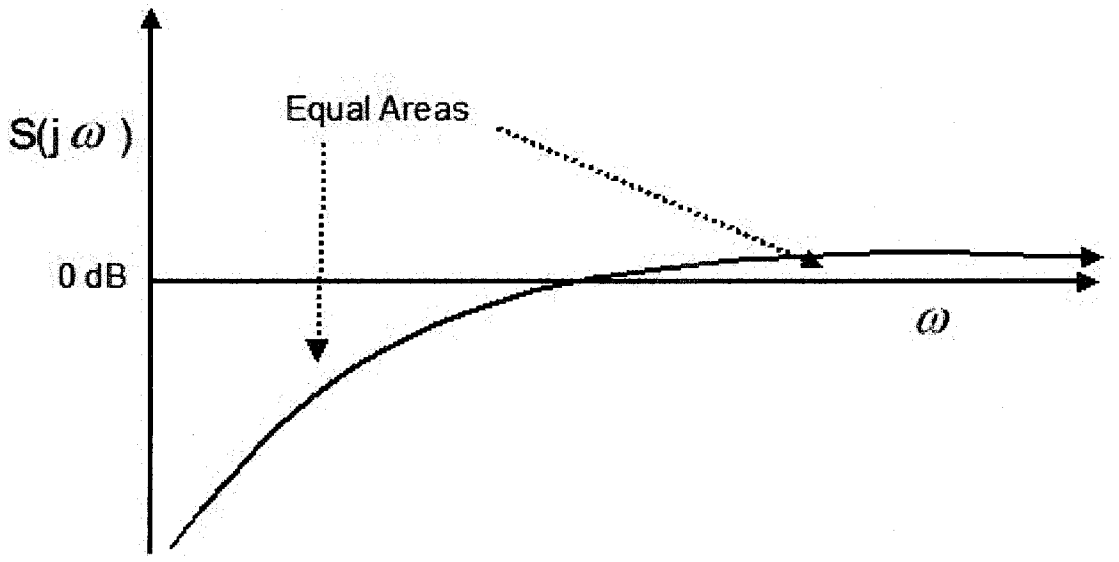


Fig. 2.45. Bode's equal area criterion

2.29.6 Performance Limitations for Plants with a Time Delay

Things simply cannot move any faster than the speed of light. If physics is correct, then there would be no way to get from one place to another instantaneously. If one were able to travel instantaneously from one place to another in a feedback loop, one would have achieved motion control with an infinite bandwidth. Obviously, this is not possible.

Feedback loops must respect time delays. To remain stable, a feedback loop must cross below the 0 dB at a frequency less than 2/(time delay). The cross over frequency is usually pretty close to the bandwidth, ω_{BW}, of the closed loop. This is illustrated in the figure below.

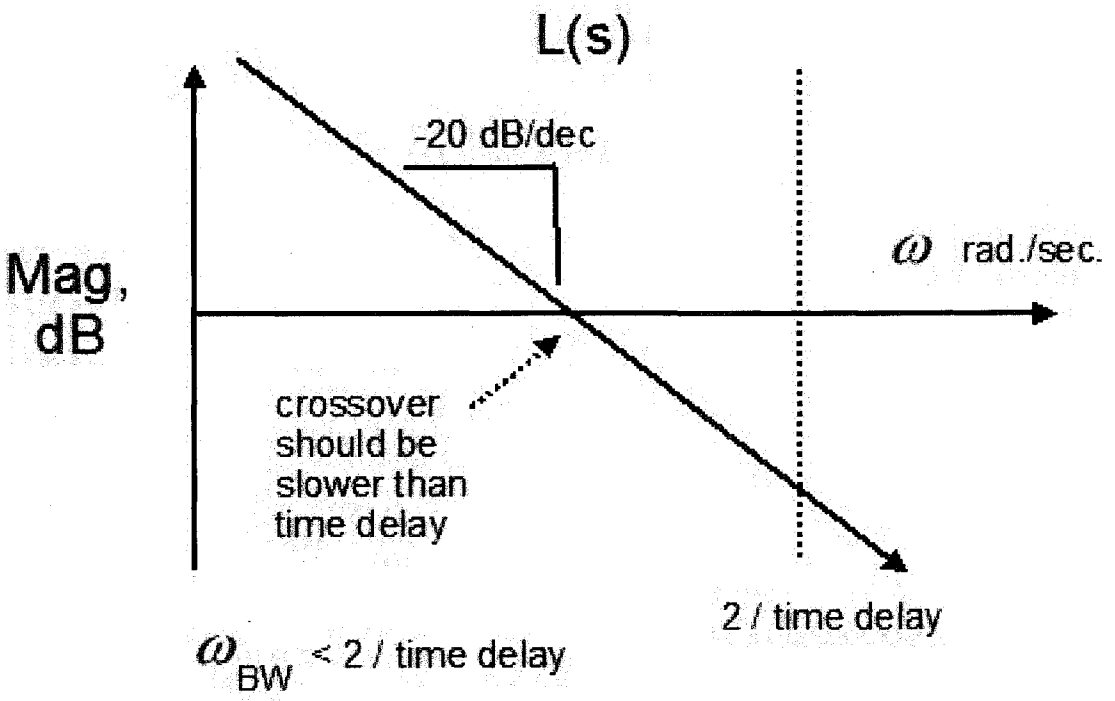


Fig. 2.46. One must crossover at a frequency slower that twice the inverse of the time delay. The time delay would be in units of seconds.

A time delay can be reasonably approximated by an unstable zero and an stable pole pair using a first order Pade' approximation below, where *td* would represent the time delay in seconds.

$$e^{-(td)s} \approx \frac{-\left(s-\frac{2}{td}\right)}{\left(s+\frac{2}{td}\right)} \tag{2.91}$$

Bear in mind that this is only the first order approximation. In truth, there would be thousands of fast unstable zeros for a true time delay. One can use the MATLAB functions "delayss" and "pade" to get higher order linear approximations to time delays in the continuous domain.

As long as one is aware of the time delays present in the system, one can remove them from the plant to make it easier for designing the controller. After the controller has been designed, then one would include these time delays for stability analysis. Many of the control designs in this work are done this way.

2.29.7 Performance Limitations for Stable Plants with a Single Unstable Zero and Unstable Poles

A renewed interest in extending Bode's sensitivity integrals to plants with unstable poles and zeros developed in the 1980's [1.6]. Deriving these expressions is quite difficult and involves Cauchy integrals for which the author is unfamiliar. There are some variations interpretations of these more recent results on bode sensitivity functions [1.8]. Nonetheless, we will summarize these results and try to communicate the meaning from Refs. [1.6,2.10].

We suppose there are N_p unstable poles at $s_{p1}^{RHP}, s_{p2}^{RHP}, \ldots, s_{pNp}^{RHP}$. Also the plant is non-minimum phase and has one unstable zero at s_{o1}^{RHP}. We also denote the conjugate transpose of s_{pk}^{RHP} as $\left(s_{pk}^{RHP}\right)^*$. A stabilizing controller must satisfy the following sensitivity integral below.

$$\int_0^{\infty} \ln\left|S(j\omega)\right| w\left(s_{o1}^{RHP}, \omega\right) d\omega = \pi \ln \prod_{k=1}^{Np} \left| \frac{s_{pk}^{RHP} + s_{o1}^{RHP}}{\left(s_{pk}^{RHP}\right)^* - s_{o1}^{RHP}} \right| \tag{2.92}$$

Note that if s_{o1}^{RHP} is near $\left(s_{pk}^{RHP}\right)^*$, then the sensitivity function can become very, very big, indicating that it is quite difficult plant to stabilize.

If $Im\left\{s_{o1}^{RHP}\right\} = 0$ and the unstable zero is on the real axis such that $s_{o1}^{RHP} = a$, then the weighting function is described by eq. 2.93.

$$w(a,\omega) = \frac{2}{a} \frac{1}{1 + \left(\omega/a\right)^2}, \quad \omega >> a \quad w(a,\omega) \approx 0 \tag{2.93}$$

With no unstable poles, $N_p = 0$, the following approximation can be made of the sensitivity function.

$$\int_0^{a} \ln\left|S(j\omega)\right| d\omega \approx 0 \tag{2.94}$$

This means that the "conservation of sensitivity dirt" is pretty much limited to a band of frequency less than the unstable zero at $s_{o1}^{RHP} = a$.

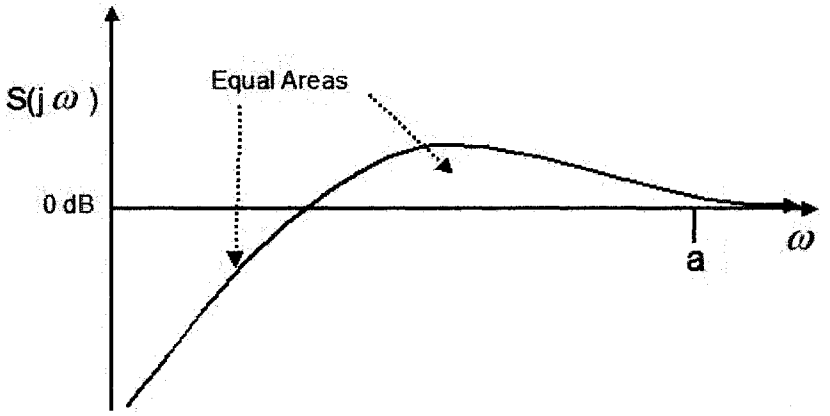


Fig. 2.47. One must crossover at a frequency slower that twice the inverse of the time delay. The time delay would be in units of seconds.

For oscillatory zeros such that $s_{o1}^{RHP} = a \pm jb$, the weighting function is described below.

$$w(a,b,\omega) = \frac{a}{a^2 + (b-\omega)^2} + \frac{a}{a^2 + (b+\omega)^2} \tag{2.95}$$

2.30 Practical Limitations of MIMO Control

Control limitations of Multi-Input-Multi-Output (MIMO) systems may become difficult because there is sometimes *directionality* involved in MIMO systems.

Example 2.17

Consider the following biproper plant with an unstable pole located at s = 1 and an unstable transmission zero at s = 1.

$$G(s) = \begin{bmatrix} \frac{s-1}{s+1} & \frac{-(s+1)}{s-1} \\ \frac{s-1}{s+1} & \frac{(s+1)}{s-1} \end{bmatrix}$$

This can be stabilized by the following regulator, $H(s)$, given below. The closed loop eigenvalues would have values at s = -1 and -0.6.

$$H(s) = \begin{bmatrix} 0 & 0 \\ -2 & 2 \end{bmatrix}$$

For a SISO system with an unstable pole-zero pair at s = 1, there would be an unstable hidden mode according to the Kalman decomposition. Thus, it would not be stabilizable if it were SISO.

However, this is a MIMO system and the unstable pole and zero at s = 1 do not share the same direction. The Kalman decomposition, covered in section 2.11, would indicate that this plant is both completely observable and completely controllable. Furthermore, we can say that this plant is *strongly stabilizable* because we designed a stabilizing controller which has no unstable poles.

Sometimes one may want to use the Youla parameterization described in section 2.23 and inverse dynamics described in Chapter 6. This helps one think about generating reasonable tracking specifications for square plants. Note that one can use an interactor matrix, $\varepsilon_L(s)$ to help with inversion. For more details, see Chapter 6.

$$T(s)=G(s)Q(s) \quad is \quad stable$$
$$iff \quad Q(s)=T(s)G(s)^{-1}=\varepsilon_L(s)T(s)[\varepsilon_L(s)G(s)]^{-1} \quad is \quad stable$$

The unstable zeros, s_{ok}^{RHP}, of a square plant $G(s)$ should not be cancelled in the closed loop.

$$if \quad \boldsymbol{det}[G(s_{ok}^{RHP})]=0 \quad then \quad \boldsymbol{det}[T(s_{ok}^{RHP})]=0 \quad \forall k$$

If one is using inverse based control for a plant with unstable zeros, then one can check to make sure that there are no unstable poles in the dynamics of the controller.

We continue with inverse based control using the numerical example above. We assign $G^{reg}(s)=(I+G(s)H(s))^{-1}G(s)$. Then, we will design a controller, $C^{reg}(s)$, such that a loopshape, $L(s)=G^{reg}(s)C^{reg}(s)$, is met. We must make sure there are no unstable poles in $C^{reg}(s)=G^{reg}(s)^{-1}L(s)$.

The inverse dynamics of $G^{reg}(s)$ are listed below.

$$G^{reg}(s)^{-1}=\begin{bmatrix} \dfrac{0.5(s+1)}{(s-1)} & \dfrac{0.5(s+1)}{(s-1)} \\ \dfrac{-2.5(s+0.6)}{(s+1)} & \dfrac{2.5(s+0.6)}{(s+1)} \end{bmatrix}$$

We can see that some of the entries of $C^{reg}(s) = G^{reg}(s)^{-1} L(s)$ must not have any poles in the denominator of any of the $C^{reg}(s)_{ij}$ polynomials.

$$C^{reg}(s)_{11} = L(s)_{11} \frac{0.5(s+1)}{(s-1)} + L(s)_{21} \frac{0.5(s+1)}{(s-1)}$$
$$C^{reg}(s)_{12} = L(s)_{12} \frac{0.5(s+1)}{(s-1)} + L(s)_{22} \frac{0.5(s+1)}{(s-1)}$$

A selection of $L(s)_{12} = -L(s)_{22}$, $L(s)_{11} = \frac{(s-1)}{something}$, and $L(s)_{21} = \frac{(s-1)}{something \quad else}$ might work. Also, a selection of $L(s)_{11} = -L(s)_{21}$, $L(s)_{22} = \frac{(s-1)}{something}$, and $L(s)_{12} = \frac{(s-1)}{something \quad else}$. What will most likely not work in the feedback loop is $L(s)_{11} = -L(s)_{21}$ and $L(s)_{12} = -L(s)_{22}$.

We decide to try $L(s)_{21} = 0$ and $L(s)_{11} = \frac{-0.25(s-1)}{s(s+1)}$. Also, we try $L(s)_{22} = \frac{3}{s}$ and $L(s)_{12} = -L(s)_{22} = -\frac{3}{s}$. Fortunately, this does work in feedback, after we correct for machine error in MATLAB, we make sure that $S^{reg}(s) = \left(I + G^{reg}(s) C^{reg}(s)\right)^{-1}$ is stable. Our controller for $G^{reg}(s)$ is listed below.

$$C^{reg}(s) = \begin{bmatrix} -0.125 & 0 \\ \frac{0.625(s-1)(s+0.6)}{(s+1)^2} & \frac{15(s+0.6)}{(s+1)} \end{bmatrix} \frac{1}{s}$$

Keep in mind that $G^{reg}(s)$ might not be biproper, so one may have to use an "interactor matrix" $\varepsilon_L(s)$. This interactor matrix is an improper transfer function with several s variables which make the transfer function proper and invertible. The $G^{reg}(s)-C^{reg}(s)$ design procedure is given full detail in Chapter 6, but a block diagram is shown below.

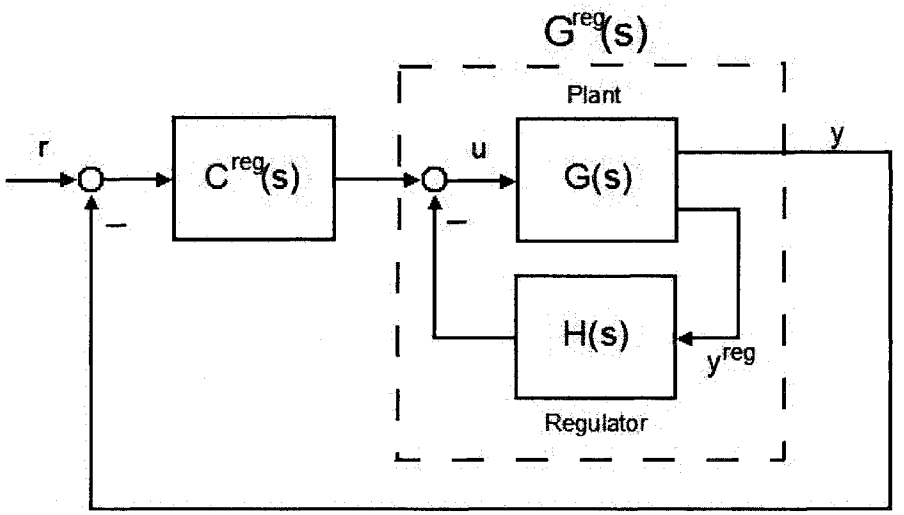


Fig. 2.48. A two-step design, $G^{reg}(s)-C^{reg}(s)$, is shown here. The first step is to design a regulator, *H(s)*, for the plant. The next step is to use a controller, $C^{reg}(s)$, for $G^{reg}(s)$.

Remarks:

1. MIMO control can get strange sometimes because there are *directional* restrictions upon what can be achieved and what cannot be achieved.
2. In this case, there was an unstable zero at s = 1 and we had a bandwidth of 0.25 rad./sec. in the first output, but a bandwidth of 3 rad./sec. in the second output. One would most likely not be

able to achieve this if the transfer function were SISO.

Example 2.17 – MATLAB code for Numerical Example

```
clc
clear all

G=[zpk([1],[-1],1) zpk([-1],[1],-1);
    zpk([1],[-1],1) zpk([-1],[1],1)];
H=[0 0;-2 2];
Greg=feedback(G,H);

% Triangular System
Ltri=[zpk([1],[0 -1],-0.25) zpk([],[0],-3);
 0 zpk([],[0],3)];
Cregtri=minreal(invGreg*Lrand,1e-2);
Cregtri(1,2)=0; %correct for machine error

Ttri=feedback(Greg*Cregtri,eye(2));
```

2.30.1 Considerations for a Class of Square MIMO Plants with Uniform Time Delays

Suppose that there is a plant which can be expressed in a right coprime form, where *td* is a time delay. This time delay is uniformly distributed at the input of the plant.

$$G(s)=G_N(s)G_D(s)^{-1}=\left[e^{(td)s}\right]^{-1}\overline{G}_N(s)=e^{-(td)s}\overline{G}_N(s)\approx\frac{-\left(s-2/td\right)}{\left(s+2/td\right)}\overline{G}_N(s) \qquad (2.96)$$

We also suppose that $G_N(s)$ has is stable and has stable transmission zeros (i.e. stable and minimum phase). Suppose further that the controller is given in the following form below.

$$C(s)=\overline{G}_N(s)^{-1}L(s),\quad L(s)\ \ is\ \ diagonal$$

Provided that the controller was parameterized in the above fashion, and that the plant really did have uniform time delays, then if $\|L(j\omega)\|<1 \quad \forall\omega\geq 2/td$ will most likely stabilize the plant.

Many times, this is a fairly good description of tuning rules and one can look at a singular value plot to show that the closed loop would be stable.

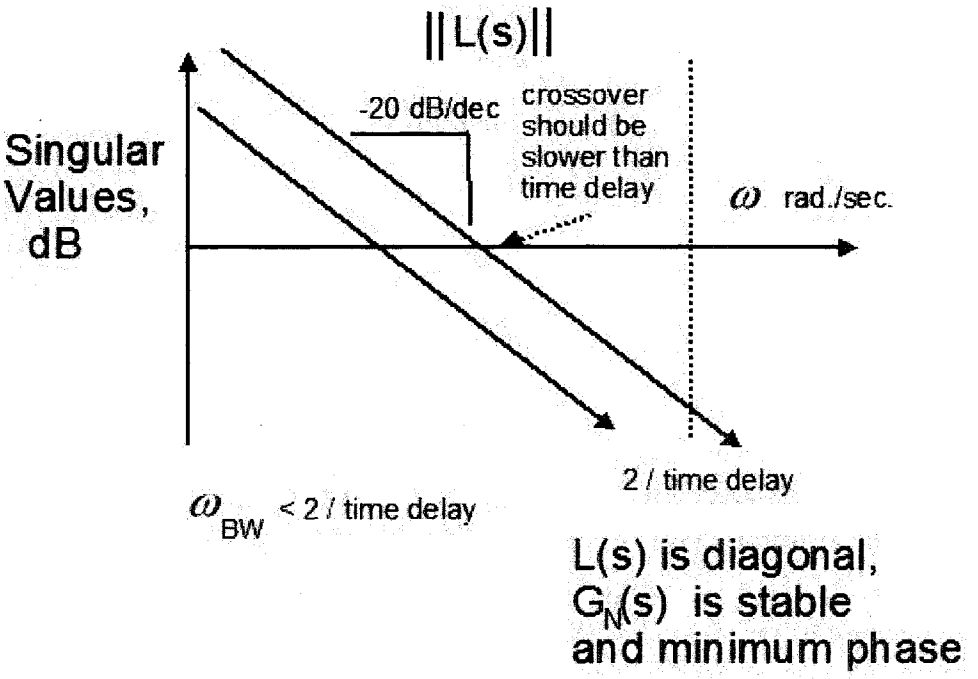


Fig. 2.49. One must crossover at a frequency slower that twice the inverse of the time delay. The time delay would be in units of seconds.

2.30.2 MIMO Sensitivity Functions

There is a whole series of MIMO sensitivity functions involving Blaschke products. These MIMO sensitivity functions usually consist of a Blaschke product and an integral of $\boldsymbol{det}(S(j\omega))$ instead of $S(j\omega)$. This topic is discussed more thoroughly in Refs. [2.10,2.16].

2.31 Classical Squaring Down of a Tall System: Lateral X-15 System

Squaring down of tall systems, i.e. systems with extra outputs, can sometimes be advantageous because one can sometimes place NMP zeros at higher frequencies or get rid of them entirely. This example could also be considered an application of "regulated variables".

The linearized lateral system of an aircraft with outputs of roll rate, *p*, yaw rate, *r*, and lateral acceleration a_y and inputs of aileron, δ_a, and rudder, δ_r can be expressed in the right coprime form in eq. 2.97.

$$G^{tall}(s)=\begin{bmatrix} N^{p}_{\delta a}(s) & N^{p}_{\delta r}(s) \\ N^{r}_{\delta a}(s) & N^{r}_{\delta r}(s) \\ N^{ay}_{\delta a}(s) & N^{ay}_{\delta r}(s) \end{bmatrix}\frac{1}{\Delta_{lat}(s)} \tag{2.97}$$

In this case, we will consider squaring down by mixing the outputs yaw rate, *r*, and lateral acceleration a_y together in the following form of equation 2.98.

$$G^{square}(s)=\left(\begin{bmatrix} N^{p}_{\delta a}(s) & N^{p}_{\delta r}(s) \\ N^{r}_{\delta a}(s) & N^{r}_{\delta r}(s) \end{bmatrix}+x\begin{bmatrix} 0 & 0 \\ N^{ay}_{\delta a}(s) & N^{ay}_{\delta r}(s) \end{bmatrix}\right)\frac{1}{\Delta_{lat}(s)} \tag{2.98}$$

Taking the determinant of $G^{square}(s)$ and setting it equal to 0 gives the equation 2.99, which is in a feedback gain form which can be used to generate a root locus.

$$1+x\frac{\left(N^{p}_{\delta a}N^{ay}_{\delta r}-N^{p}_{\delta r}N^{ay}_{\delta a}\right)}{\left(N^{p}_{\delta a}N^{r}_{\delta r}-N^{p}_{\delta r}N^{ay}_{\delta a}\right)}=0 \tag{2.99}$$

Now, one can draw the root locus as if the "plant", $G^{x}(s)$, were

$G^{x}(s)=\frac{\left(N^{p}_{\delta a}N^{ay}_{\delta r}-N^{p}_{\delta r}N^{ay}_{\delta a}\right)}{\left(N^{p}_{\delta a}N^{r}_{\delta r}-N^{p}_{\delta r}N^{ay}_{\delta a}\right)}$. From there, we can decide upon where to place the transmission zeros of the square system depending upon our choice of *x*.

X-15 at Mach 2.0, 60,000 ft

The state space representation of the vehicle and transfer function representation is listed below.

$$A=\begin{bmatrix} Y_v & 0 & -1 & {}^{g}/_{U_o}\cos\phi_o \\ L'_\beta & L'_p & L'_r & 0 \\ N'_\beta & N'_p & N'_r & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad B=\begin{bmatrix} Y^*_{\delta a} & Y^*_{\delta r} \\ L'_{\delta a} & L'_{\delta r} \\ N'_{\delta a} & N'_{\delta r} \\ 0 & 0 \end{bmatrix} \quad C=\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ U_oY_v & 0 & 0 & g\cos\phi_o \end{bmatrix}$$

$$D=\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ U_oY^*_{\delta a} & U_oY^*_{\delta r} \end{bmatrix}$$

$$y=\begin{bmatrix} p \\ r \\ a_y \end{bmatrix} \quad u=\begin{bmatrix} \delta_a \\ \delta_r \end{bmatrix} \quad x=\begin{bmatrix} \beta \\ p \\ r \\ \Delta\phi \end{bmatrix}$$

U_o	g	Y_v	L'_β	L'_p	L'_r
1936	32.2	-0.127	-2.36	-1.02	0.103
N'_β	N'_p	N'_r	$Y^*_{\delta a}$	$Y^*_{\delta r}$	$L'_{\delta a}$
11.1	-0.00735	-0.196	-0.00498	0.0426	28.7
$L'_{\delta r}$	$N'_{\delta a}$	$N'_{\delta r}$			
4.21	0.993	-2.88			

Table 2.7. Stability derivatives for the X-15 at Mach = 2.0 at 60,000ft.

$$N^p_{\delta a} = 28.7s^3 + 9.384s^2 + 321.6s$$
$$N^p_{\delta r} = 4.21s^3 + 0.9627s^2 + 40.03s$$
$$N^r_{\delta a} = 0.993s^3 + 0.8727s^2 + 0.04537s + 5.338$$
$$N^r_{\delta r} = -2.88s^3 - 2.861s^2 + 0.1061s + 0.6642$$
$$N^{ay}_{\delta a} = -9.641s^4 - 11.72s^3 + 1059s^2 + 272.3s + 1.033x10^4$$
$$N^{ay}_{\delta r} = 82.47s^4 + 100.3s^3 + 359.5s^2 + 222.3s + 1286$$
$$\Delta_{lat} = s^4 + 1.343s^3 + 11.46s^2 + 11.4s - 0.01132$$

Now, if the only outputs were p and r were considered, there would be a very slow unstable transmission zero at $s_o = 0.0372$. So, we will be considering new outputs p and $r + xa_y$ to see what happens. Using the expression in eq. 2.99, one can draw the root locus. Do note that poles of the root locus are the transmission zeros of the new system. For this example of the X-15 from Ref. [2.21],

$G^x(s) = \dfrac{-27.73s^2 - 5.498s - 63.35}{s - 0.03272}$, which is an improper transfer function. Nonetheless, we notice that if x is negative than the unstable transmission zero can be moved into the stable region.

Now, we double check our work, using the "zero( )" command in MATLAB and notice that, in fact, the transmission zero has been moved into the LHP when using the new output of $r - 0.001a_y$.

It is now possible to achieve integral action in both channels, whereas before this would not have been possible due to the low frequency unstable zero.

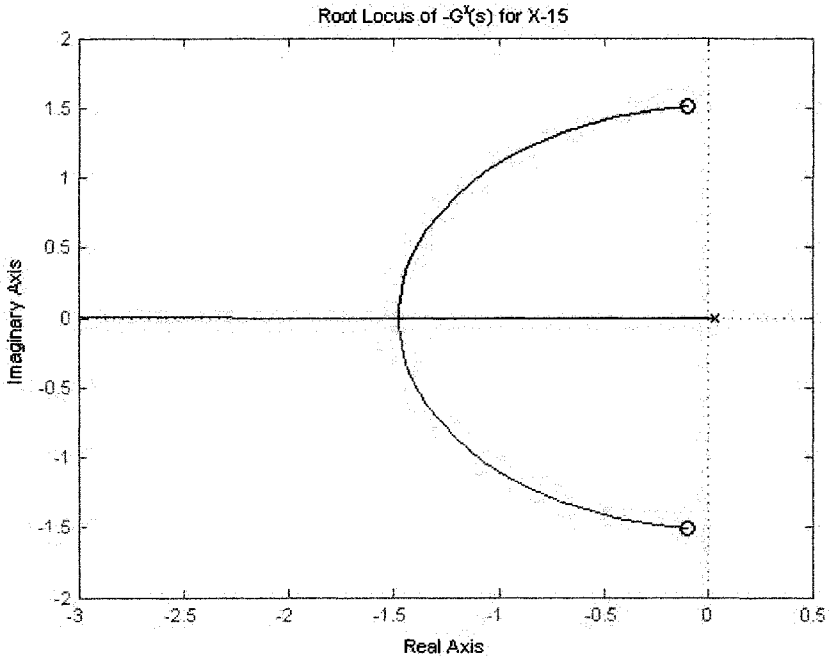


Fig. 2.50 Root locus of $-G^{x}(s)=\dfrac{27.73s^{2}+5.498s+63.35}{s-0.03272}$. This can be used to place the transmission zeros. A value of $x=-0.001$ is enough to place the unstable transmission zero in the LHP.

2.32 Classical Squaring Down of a Fat System: Longitudinal F-18 Carrier Landing

Squaring down of fat systems, i.e. systems with extra inputs, can sometimes be advantageous because one can sometimes place NMP zeros at higher frequencies to achieve better performance. This motivating example shows how one can place NMP zeros at higher frequencies.

Step 1: Mixing Leading Edge Flap and Elevator

The linearized lateral system of an F-18 will be considered in here [2.24]. The outputs are flight path angle, γ, and a "normalized velocity" $\bar{u}=\Delta u/U_{o}$. The inputs consist of elevator, δ_{e}, an engine power lever, δ_{PL}, and a Leading Edge Flap (LEF), δ_{LEF}. This system is expressed in the right coprime form in eq. 2.100.

$$G^{fat}(s)=\begin{bmatrix} N_{\delta e}^{\gamma}(s) & N_{\delta PL}^{\gamma}(s) & N_{\delta LEF}^{\gamma}(s) \\ N_{\delta e}^{\bar{u}}(s) & N_{\delta PL}^{\bar{u}}(s) & N_{\delta LEF}^{\bar{u}}(s) \end{bmatrix}\frac{1}{\Delta_{lon}(s)} \tag{2.100}$$

In this case, we will consider squaring down by mixing the inputs, δ_{e}, and δ_{LEF} together in the following form of equation 2.101.

$$G^{square}(s)=\left(\begin{bmatrix} N^{\gamma}_{\delta e}(s) & N^{\gamma}_{\delta PL}(s) \\ N^{\bar{u}}_{\delta e}(s) & N^{\bar{u}}_{\delta PL}(s) \end{bmatrix}+x\begin{bmatrix} N^{\gamma}_{\delta LEF}(s) & 0 \\ N^{\bar{u}}_{\delta LEF}(s) & 0 \end{bmatrix}\right)\frac{1}{\Delta_{lon}(s)} \tag{2.101}$$

Assuming $G^{square}(s)$ is minimal realization, we can take the determinant of $G^{square}(s)$ and set it equal to 0 in equation 2.103. Notice that eq. 2.103 is in a feedback gain form which can be used to generate a root locus.

$$1+x\frac{\left(N^{\gamma}_{\delta LEF}N^{\bar{u}}_{\delta PL}-N^{\gamma}_{\delta PL}N^{\bar{u}}_{\delta LEF}\right)}{\left(N^{\gamma}_{\delta e}N^{\bar{u}}_{\delta PL}-N^{\gamma}_{\delta PL}N^{\bar{u}}_{\delta e}\right)}=0 \tag{2.103}$$

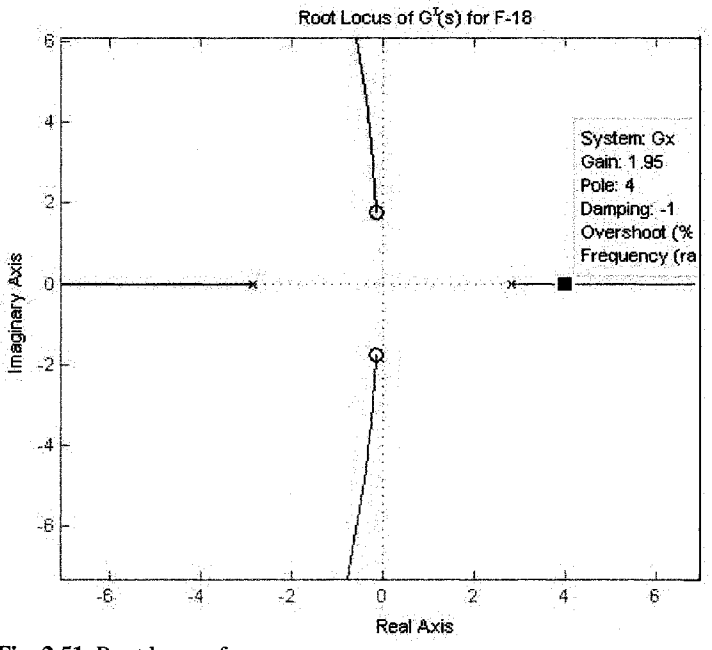


Fig. 2.51. Root locus of $G^{x}(s)=\dfrac{-0.204s^2-0.06169s-0.6385}{s^2+0.03065s-8.036}$. This can be used to place the transmission zeros. Using a value of $x=1.95$ is enough to double the unstable transmission zero. Now, one can use the control of $u_{1/2}=\delta_e+1.95\delta_{LEF}$ to increase the achievable performance!

We can draw the root locus as if the "plant", $G^{x}(s)$, were

$$G^{x}(s)=\frac{\left(N^{\gamma}_{\delta LEF}N^{\bar{u}}_{\delta PL}-N^{\gamma}_{\delta PL}N^{\bar{u}}_{\delta LEF}\right)}{\left(N^{\gamma}_{\delta e}N^{\bar{u}}_{\delta PL}-N^{\gamma}_{\delta PL}N^{\bar{u}}_{\delta e}\right)}$$

From there, we can decide upon where to place the transmission zeros of the square system depending upon our choice of x.

Looking at the root locus in Fig. 2.51, one can see that using an input of $u_{1/2}=\delta_e+1.95\delta_{LEF}$ can increase the achievable performance when compared to the system with only elevator δ_e and engine power lever δ_{PL}.

This approach to squaring down is good because it can give hardware insight before control synthesis even begins. For example, suppose that both the elevator and leading edge flap saturated at 30 deg. deflection. If one wanted to design a controller that took advantage of the NMP shift of

F-18 at 136kts, 50ft

$$A=\begin{bmatrix} -0.0705 & 0.0475 & -.1403 & 0.000 & -0.000058 \\ -.3110 & -0.3430 & 0.000 & 0.99133 & 0.00102 \\ 0.0000 & 0.0000 & 0.000 & 1.0000 & 0.0000 \\ 0.0218 & -1.1660 & 0.000 & -0.2544 & 0.0000 \\ 0.0000 & -1.000 & 1.000 & 0.000 & 0.000 \end{bmatrix}$$

$$B=\begin{bmatrix} 0.0121 & 0.2316 & 0.00248 \\ -0.0721 & -0.0338 & 0.0140 \\ 0.000 & 0.0000 & 0.0000 \\ -1.8150 & 0.0023 & -0.0790 \\ 0.000 & 0.000 & 0.0000 \end{bmatrix} \quad C=\begin{bmatrix} 0 & -1.000 & 1.000 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

State Space data from Ref. [2.20]. State vector is $x=\left[\Delta u/U_o \quad \Delta\alpha \quad \Delta\theta \quad q \quad \Delta h/U_o\right]^T$ units are [non-dimensional rad. rad. rad. sec^{-1}] respectively. Input vector is $u=[\delta_e \quad \delta_{PL} \quad \delta_{LEF}]^T$ units are [rad. deg. rad.]. The output vector is $y=\left[\gamma \quad \Delta u/U_o\right]^T$, units are [non-dimensional and rad.] respectively.

$s_o = 2.8195$ to $s_o = 4.0$, then one could only use this controller while the leading edge flap was not saturated. This would mean that the elevator could only move 15 deg. with this faster controller. If one were unhappy with this, an alternative would be to design a more powerful leading edge flap.

Oddly enough, some canard configurations such as the X-29, do not exhibit the NMP zero which is normally seen on conventional elevator-thrust configurations. Also, the F-15 Short-takeoff-and-landing Maneuver Technology Demonstrator (S/MTD) program, was very successful with a canard-elevator-vectored thrust configuration (though they might have done just fine with the canard alone).

Then why are aircraft with canards the exception rather than the norm? Oftentimes, aircraft with canards, such as the X-29 are open-loop unstable and require active control to stabilize them.

Step 2: Mixing in Rudder Toe-In (a strange actuator)

The F-18 in Ref. [2.24] also has a very unconventional fourth actuator which we will consider adding once again into the system. It is truly strange and almost unbelievable that one could actually generate forward thrust by moving a rudder "toe-in" angle. Yet, the linear model in Ref. [2.24], claims that this is possible. Regardless, we will move forward under the assumption that this is rudder "toe in"

concept is somewhat valid.

F-18 at 136kts, 50ft: After step 1

```
     -0.0705 0.0475 -.1403 0.000 -0.000058
     -.3110 -0.3430 0.000 0.99133 0.00102
A=   0.0000 0.0000  0.000  1.0000   0.0000
     0.0218 -1.1660 0.000  -0.2544 0.0000
     0.0000 -1.000  1.000  0.000    0.000

      0.0169  0.2316  0.1690
     -0.0448 -0.0338  0.0128  C=   0 -1.000 1.000 0 0
B=    0.000   0.0000  0.0000       1  0     0     0 0
     -1.9691  0.0023  0.1681
      0.000   0.000   0.0000
```

State Space data from Ref. [2.20]. State vector is $x = \left[\Delta u/U_o \quad \Delta\alpha \quad \Delta\theta \quad q \quad \Delta h/U_o \right]^T$ units are [non-dimensional rad. rad. rad. sec^{-1}] respectively. Input vector is $u = \left[u_{1/2} \quad \delta_{PL} \quad \delta_{RT} \right]^T$ $u_{1/2} = \delta_e + 1.95\delta_{LEF}$ units are [rad. deg. rad.]. The output vector is $y = \left[\gamma \quad \Delta u/U_o \right]^T$, units are [non-dimensional and rad.] respectively.

The system is fattened once again in right coprime form and we use the previous result from step 1, where $u_{1/2} = \delta_e + 1.95\delta_{LEF}$.

$$G^{fat}(s) = \begin{bmatrix} N^{\gamma}_{u1/2}(s) & N^{\gamma}_{\delta T} & N^{\gamma}_{\delta RT}(s) \\ N^{\bar{u}}_{u1/2}(s) & N^{\bar{u}}_{\delta T} & N^{\bar{u}}_{\delta RT}(s) \end{bmatrix} \frac{1}{\Delta_{lon}(s)} \tag{2.104}$$

We will consider squaring down in the following fashion in eq. 2.105.

$$G^{square}(s) = \left(\begin{bmatrix} N^{\gamma}_{u1/2}(s) & N^{\gamma}_{\delta PL}(s) \\ N^{\bar{u}}_{u1/2}(s) & N^{\bar{u}}_{\delta PL}(s) \end{bmatrix} + x \begin{bmatrix} N^{\gamma}_{\delta RT}(s) & \chi N^{\gamma}_{\delta RT}(s) \\ N^{\bar{u}}_{\delta RT}(s) & \chi N^{\bar{u}}_{\delta RT}(s) \end{bmatrix} \right) \frac{1}{\Delta_{lon}(s)} \tag{2.105}$$

Where x is a gain that we will use again and χ is some number chosen prior. To add in so that the final control scheme will be $u_1 = u_{1/2} + x\delta_{RT} = \delta_e + 1.95\delta_{LEF} + x\delta_{RT}$ and $u_2 = \delta_{PL} + x\chi\delta_{RT}$. Arranging the determinant again, one can express the transmission zeros (assuming no pole-zero cancellations) in eq. 2.106.

$$1 + x\frac{\left(N^{\gamma}_{\delta RT}N^{\bar{u}}_{\delta PL} - N^{\gamma}_{\delta PL}N^{\bar{u}}_{\delta RT} + \chi N^{\gamma}_{u1/2}N^{\bar{u}}_{\delta RT} - \chi N^{\bar{u}}_{u1/2}N^{\gamma}_{\delta RT}\right)}{\left(N^{\gamma}_{u1/2}N^{\bar{u}}_{\delta PL} - N^{\gamma}_{\delta PL}N^{\bar{u}}_{u1/2}\right)} = 0 \tag{2.106}$$

Now since the author is mostly skeptical that moving the rudder can actually produce some forward thrust, the parameter $\chi = .02$ will be chosen.

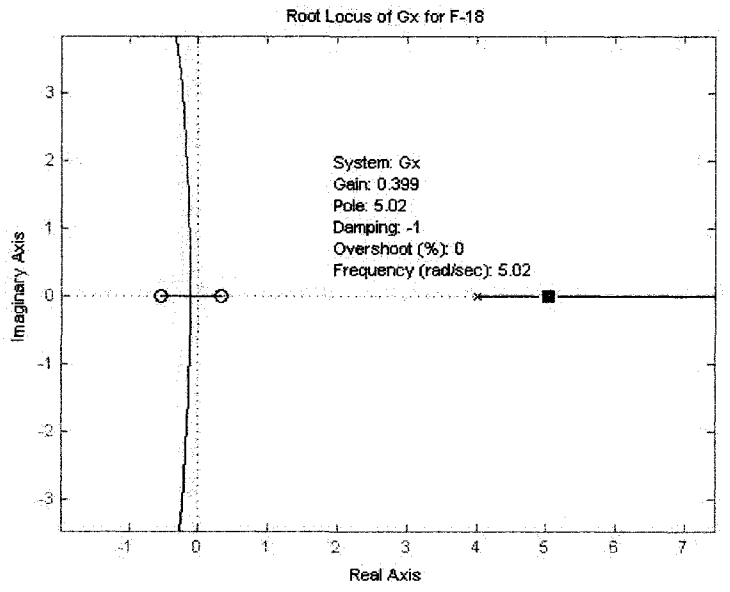


Fig. 2.52. Root locus of $G^x(s) = \dfrac{-0.8692s^2 - 0.193s + 0.1576}{s^2 - 0.1489s - 15.42}$, from this plot, one can decide where to place the transmission zeros. Using a value of $x = 0.4$ is enough to move the unstable transmission zero out to 5rad./sec. Now, one can use the control of $u_1 = u_{1/2} + 0.4\delta_{RT}$ and $u_2 = \delta_{PL} + 0.008\delta_{RT}$ to increase the achievable performance!

After inspecting the root locus in Fig. 2.52, one can place the transmission zero once again out at $s_o \approx 5.00$, which can increase achievable performance.

By mixing in the extra actuators of leading edge flap and rudder "toe-in" in the following amounts $u_1 = \delta_e + 1.95\delta_{LEF} + 0.4\delta_{RT}$, and $u_2 = \delta_{PL} + 0.008\delta_{RT}$, one can increase the achievable performance. Of course, whether or not this can work depends highly upon the available control authority from leading edge flap and rudder "toe-in". If a single compensator was designed to cross over aggressively at 3 rad./sec., and the leading edge flap saturated, then this design would most likely be unstable. This is because with the elevator and power lever alone, the NMP zero was at $s_o \approx 2.85$.

2.33 Squaring Down of a Fat System with Inverse Dynamics: General Case

In previous sections, it has been shown by example that one can use pole placement strategies to move transmission zeros around with a step-by-step, root locus approach. It seems natural to expect that one could get something similar with a state-feedback pole placement equivalent. However, it also seems as though this would be difficult to extend this universally to tall systems, since the X-15 generated an improper transfer function. We will pursue squaring down of fat systems in this section.

From section 2.20, it is true that for any biproper transfer function, $G(s)$, the poles of $G(s)^{-1}$ are the transmission zeros of $G(s)$. So, what we will done is to do pole placement for the inverse system. Then we know that we will have placed the zeros of the inverse system.

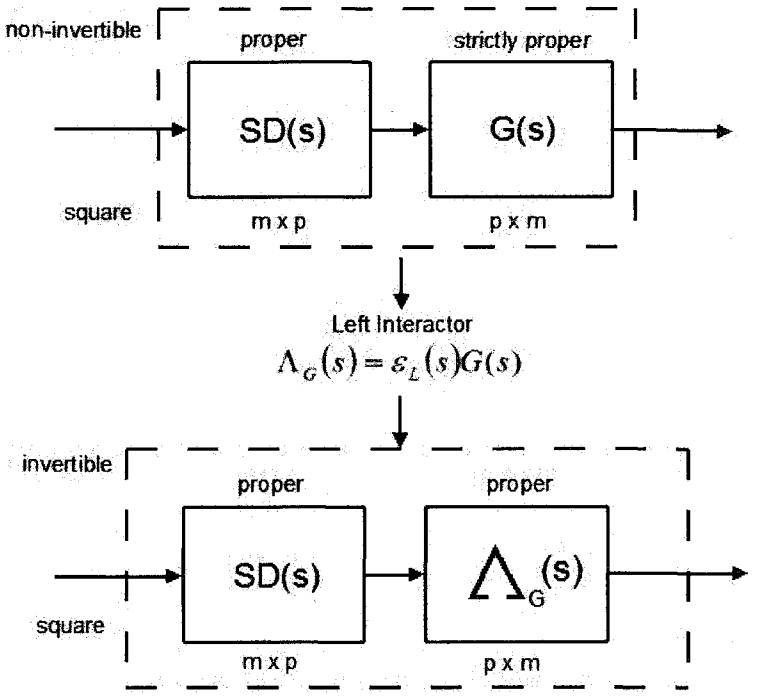


Fig. 2.53. Squaring down procedure involves designing the SD(s) transfer function which makes SD(s) and G(s) in series square and have desirable zeros.

A state space realization for the square down system, $SD(s)$, and $\Lambda_G(s)$ in series is listed below in eq. 2.107.

$$\begin{aligned} A_{\Lambda SD} &= \begin{bmatrix} A_\Lambda & B_\Lambda C_{SD} \\ 0 & A_{SD} \end{bmatrix} \\ B_{\Lambda SD} &= \begin{bmatrix} B_\Lambda D_{SD} \\ B_{SD} \end{bmatrix} \\ C_{\Lambda SD} &= \begin{bmatrix} C_\Lambda & D_\Lambda C_{SD} \end{bmatrix} \\ D_{\Lambda SD} &= \begin{bmatrix} D_\Lambda D_{SD} \end{bmatrix} \end{aligned} \tag{2.107}$$

The state space realizations are $\Lambda_G(s) = (A_\Lambda, B_\Lambda, C_\Lambda, D_\Lambda)$ and $SD(s) = (A_{SD}, B_{SD}, C_{SD}, D_{SD})$.

The state space realization of the inverse dynamics, $(\Lambda_G(s)SD(s))^{-1} = (A^{inv}_{\Lambda SD}, B^{inv}_{\Lambda SD}, C^{inv}_{\Lambda SD}, D^{inv}_{\Lambda SD})$, are given below in eq. 2.108.

$$\begin{aligned} A^{inv}_{\Lambda SD} &= \begin{bmatrix} A_\Lambda - B_\Lambda D_{SD}(D_\Lambda D_{SD})^{-1} C_\Lambda & B_\Lambda C_{SD} - B_\Lambda D_{SD}(D_\Lambda D_{SD})^{-1} D_\Lambda C_{SD} \\ -B_{SD}(D_\Lambda D_{SD})^{-1} C_\Lambda & A_{SD} - B_{SD}(D_\Lambda D_{SD})^{-1} D_\Lambda C_{SD} \end{bmatrix} \\ B^{inv}_{\Lambda SD} &= \begin{bmatrix} B_\Lambda D_{SD}(D_\Lambda D_{SD})^{-1} \\ B_{SD}(D_\Lambda D_{SD})^{-1} \end{bmatrix} \\ C^{inv}_{\Lambda SD} &= \begin{bmatrix} -(D_\Lambda D_{SD})^{-1} C_\Lambda & -(D_\Lambda D_{SD})^{-1} D_\Lambda C_{SD} \end{bmatrix} \quad D^{inv}_{\Lambda SD} = (D_\Lambda D_{SD})^{-1} \end{aligned} \tag{2.108}$$

Since the zeros of $\Lambda_G(s)SD(s)$ correspond to the poles of $(\Lambda_G(s)SD(s))^{-1}$, we can focus on the eigenvalues of $A^{inv}_{\Lambda SD}$. We notice that for the square case, cancellation of terms should occur in the (1,2)

entry of $A_{\Lambda SD}^{inv}$ as $B_\Lambda C_{SD} - B_\Lambda D_{SD}(D_\Lambda D_{SD})^{-1} D_\Lambda C_{SD} = 0$. We make the following association with an state feedback and observer design in eq. 2.109.

$$A_{\Lambda SD}^{inv} = \begin{bmatrix} A_\Lambda - B_\Lambda D_{SD}(D_\Lambda D_{SD})^{-1} C_\Lambda & B_\Lambda\left(I - D_{SD}(D_\Lambda D_{SD})^{-1} D_\Lambda\right)C_{SD} \\ -B_{SD}(D_\Lambda D_{SD})^{-1} C_\Lambda & A_{SD} - B_{SD}(D_\Lambda D_{SD})^{-1} D_\Lambda C_{SD} \end{bmatrix}$$

$$A_{\Lambda SD}^{inv} = \begin{bmatrix} A_f & -B_f K_f \\ L_f C_f & A_f - L_f C_f - B_f K_f \end{bmatrix}$$

$$A_f = A_\Lambda - B_\Lambda D_{SD}(D_\Lambda D_{SD})^{-1} C_\Lambda \quad L_f = -B_{SD}(D_\Lambda D_{SD})^{-1} \quad C_f = C_\Lambda \qquad (2.109)$$

$$K_f = -C_{SD} \quad B_f = B_\Lambda\left(I - D_{SD}(D_\Lambda D_{SD})^{-1} D_\Lambda\right)$$

$$A_{SD} = A_f - L_f C_f - B_f K_f + B_{SD}(D_\Lambda D_{SD})^{-1} D_\Lambda C_{SD}$$

Notice that in the event that D_Λ and D_{SD} are square, then the zero dynamics will be defined by $A_\Lambda - B_\Lambda D_\Lambda^{-1} C_\Lambda$ and no influence can be made by squaring down.

So, we outline the following procedure for squaring down a fat system.

Step 1: Find a Left Interactor for the Plant – Procedures for finding interactor matrices of fat systems are described in section 6.2. Regardless, we introduce the quick solutions for fat systems of uniform relative degree 1 and 2.

Uniform relative degree 1:
$$G(s) = (A, B, C, 0) \quad rank(CB) = p$$
$$sG(s) = \Lambda_G(s) = (A, B, CA, CB)$$

Uniform relative degree 2:
$$G(s) = (A, B, C, 0) \quad CB = 0 \quad rank(CAB) = p$$
$$s^2 G(s) = \Lambda_G(s) = (A, B, CA^2, CAB)$$

Step 2: Proportional Output Feedback Design – One needs to pick F such that the eigenvalues of $A_f = A_\Lambda - B_\Lambda F C_\Lambda$ are in approximately desirable positions. With proportional output feedback, one cannot achieve exact eigenvalue placement, so it is an approximate process. One could do this directly with $F = D_{SD}(D_\Lambda D_{SD})^{-1}$. Alternatively, one could decide on F first and then solve D_{SD} given $F = D_{SD}(D_\Lambda D_{SD})^{-1}$. Then, one would use $D_{SD} = null(FD_\Lambda - I)$. Where D_{SD} is the right null space of $FD_\Lambda - I$.

Step 3 (Optional): Observer-State Feedback Design – This step can only be completed once step 2 is done. If $D_{SD} = 0$, then the problem is not solvable. One would use the definitions in eq. 2.109 to design an observer for (A_f, C_f), then a state feedback regulator for (A_f, B_f).

Example 2.18 – MATLAB code for transmission zero placement via Inverse Dynamics

```
A=[-1 20 0;-0.1 -1 0;0 1 0.01];
B=[-2 -3;-1 1;0 0];
C=[-.04 0 1];

%NMP at s = 4.02 with only 1 actuator
zero((ss(A,B(:,1),C,0)));

%Fat system p < m
sys=ss(A,B,C,0);

%Step 1, left interactor
s=tf('s');
left_int=s;

%Step 1, define gamma system
gamma_g=minreal(left_int*sys);
[A_gam,B_gam,C_gam,D_gam]=ssdata(gamma_g);

%Step 2, proportional output feedback problem, use
%optimization to place at maximum eigenvalue at 6
D_sdo=D_gam';
D_sd=fminsearch(@(D_sd) (max(eig(A_gam-B_gam*D_sd*...
    inv(D_gam*D_sd)*C_gam)-6))^2,D_sdo);

%places NMP zero at 6
zero(sys*D_sd)

%Step 3 (optional), include state feedback - observer
A_f=A_gam-B_gam*D_sd*inv(D_gam*D_sd)*C_gam;
B_f=B_gam*(eye(2)- D_sd*inv(D_gam*D_sd)*D_gam);
C_f=C_gam;
%Observer - Kalman Filter
L_ft=lqr(A_f',C_f',10*eye(3),eye(1));
L_f=L_ft';
%Optimal lqr
K_f=lqr(A_f,B_f,eye(3),eye(2));

B_sd=-L_f*(D_gam*D_sd);
C_sd=-K_f;
A_sd=A_f-L_f*C_f-B_f*K_f+B_sd*inv(D_gam*D_sd)*D_gam*C_sd;

SD=ss(A_sd,B_sd,C_sd,D_sd);
A_inv_gamsd=[A_f -B_f*K_f;L_f*C_f A_f-L_f*C_f-B_f*K_f];
predicted=eig(A_inv_gamsd)
true=zero(minreal(s*sys*SD))
```

2.34 MATLAB Commands

Table 2.8 – Useful MATLAB commands

Command	Purpose	Example
eye(n)	Creates an [n x n] identity matrix	eye(5)
zeros(m,n)	Creates an [m x n] matrix with 0 entries	zeros(2,3)
ss(A,B,C,D)	Creates a linear transfer function from the state space	A=[0 1;-1 -2]; B=[0; 1]; C=[1 0]; sys = ss(A,B,C,zeros(2,2))
zpk(z,p,k)	Creates a linear transfer function from zero, pole and gain representation	sys = zpk([-2],[-1+j -1-j],2)
tf(num,den)	Creates a linear transfer function from polynomial numerator, num, and polynomial denominator, den	num = [2 4]; den=[1 2 2]; sys =tf(num,den)
conv(p1,p2)	Convolves two polynomials p1 and p2	p1=[1 1+j]; p2=[1 1-j]; p3 = conv(p1,p2)
roots(p1)	Calculates the roots of the polynomial of p1	p1=[1 1+j]; p2=[1 1-j]; r = roots(conv(p1,p2))
[V,D] = eig(A)	Calculates the eigenvalues of the square matrix A, eigenvectors are V, D is a diagonal matrix of eigenvalues	A = [-1 -1;0 -2]; [V,D] = eig(A)
[U,S,V] = svd(G)	Calculates singular value decomposition, $G = USV^T$	G = [1 2 3;4 5 6]; [U,S,V] = svd(G)
null(F)	Calculates right nullspace of matrix F	F = [1 1;1 1]; vec = null(F)
pinv(F)	Calculates the pseudo-inverse of the matrix F, F*pinv(F)*F = F	F = [1 2 3;4 5 6]; Finv = pinv(F);
rank(F)	Calculates the rank of a matrix F	F = [1 1;1 1]; one = rank(F)
det(F)	Calculates the determinant of a square matrix, F	F = [1 1;-1 1]; two = det(F)
zero(sys)	Calculates the transmission zeros of a linear system, sys	sys = zpk([-2],[-1 -3],5) neg_two = zero(sys)
minreal(sys,TOL)	Performs close pole-zero	sys = zpk([-2],[-2.02],5)

	cancellations of sys with the relative tolerance, TOL	sysmin = minreal(sys,0.01)
inv(sys)	Inverts a biproper transfer function, sys *Older versions of MATLAB will generate an error message if sys is not biproper, Newer versions of MATLAB will generate a very strange transfer and high order function, do not use inv(sys) unless sys is biproper	A = zeros(2,2); B = eye(2); C = eye(2); D = eye(2); det(D) is non-zero sysinv = inv(ss(A,B,C,D));
rlocus(sys)	Draws a root locus for the SISO system sys	sys = zpk([-2],[-1 -3],5) rlocus(sys)
bode(sys)	Draws a bode plot for the system sys (works for MIMO systems as well)	sys = zpk([-2],[-1 -3],5) bode(sys)
sigma(sys)	Draws a singular value plot of the system, sys	A = zeros(2,2); B = eye(2); C = eye(2); D = eye(2); sigma(ss(A,B,C,D))
sisotool(sys)	A very useful graphical design tool which simultaneously draws a bode plot and the root locus. Sisotool can be used for continuous domain design or direct digital design	sys = zpk([-2],[-1 -3],5) sisotool(sys)
feedback(G,H)	Connect the system G and the system H in the feedback configuration, can also be used for tracking systems with unity feedback	G = zpk([],[-2],1); H = 1; C = zpk([-2],[0],2); Greg = feedback(G,H); T = feedback(G*C,eye(2));
place(A,B,poles)	Performs pole placement that minimizes condition number of eigenvectors with state feedback	A = 0; B = 1; three = place(A,B,-3);
loopsyn(sys,L)	Performs a loop-shaping procedure for square and minimum phase systems with H-inifinity model matching, result is	s = tf('s'); G = [-1 1;1 1]*1/(s+2); C = loopsyn(G,6/s);

	somewhat similar to diagonal decoupling with uniform bandwidth	
margin(sys)	Calculates gain and phase margins for siso systems	GC = zpk([10],[0 -10],-1); ten = margin(GC)
loopmargin(sys,C)	Calculates individual gain and phase margins for MIMO systems, loop by loop. loopmargin( ) does not consider loop interactions. Example 2.16 in section 2.26 takes into account what loopmargin( ) does not.	
lqr(A,B,W,V)	Calculates an optimal control law set by the parameters W and V	
delayss(A,B,C,D,delayterms)	Creates a state space with delayed terms	

evalfr(sys,f)	Evaluates a linear system at the complex value f	s = tf('s');
		G = [-1 1;1 1]*1/(s+2);
		f = -2-j;
		Gf = evalfr(G,f)

c2d(sys,T,'method')	Converts a continous transfer function to a digital representation with sampling rate T with a given method	sys = zpk([10],[0 -10],-1); sysd = c2d(sys,0.1,'zoh')
d2c(sysd)	Converts a digital transfer function to a continuous representation with sampling rate T with a given method	sysd = zpk([],[0.9],1,0.1); sys = d2c(sysd)

Chapter 3 – Background Material – Data Specific to Flight Control

3.1 Linearized Undamaged Rigid Body Aircraft Dynamics

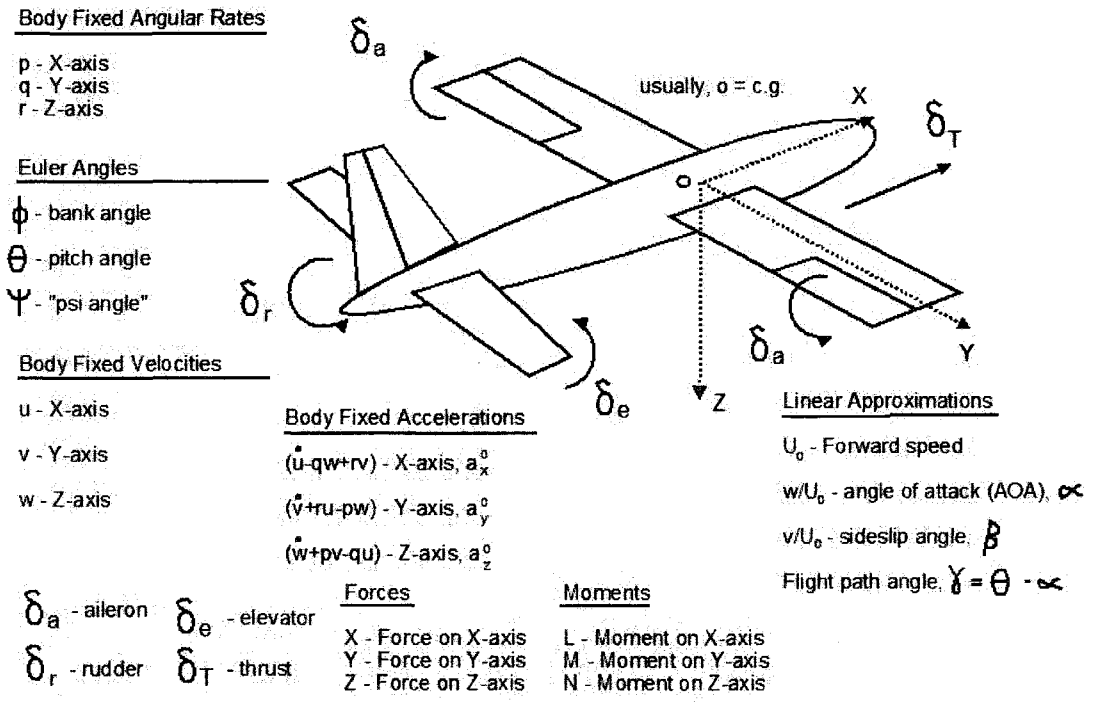


Fig. 3.1. Definition of body fixed reference frame for rigid aircraft dynamics.

Linearized aircraft dynamics can be used to design controllers for aircraft. These linear controllers are typically quite good in the realm of $-30\boldsymbol{deg}. \leq \phi \leq 30\boldsymbol{deg}.$ and reasonable flight path angles. Basically, there is a trajectory assumption associated with the linearization.

Stability/Wind Axis System (Body Fixed)

The idea of the stability or wind axis system is that the X-axis is aligned with the equilibrium forward velocity, U_0. Confusion may arise because of the terminology associated with different axis systems. An attempt to explain this will be made, though more thorough discussions on axis systems can be found in [3.1-3.4]

The wind axis system and the stability axis system are the same axis system. The X-axis is aligned with the equilibrium forward velocity. The X-axis of the stability axis system may not be aligned with a Fuselage Reference Line (FRL).

The stability axis is not at all associated with dynamic stability.

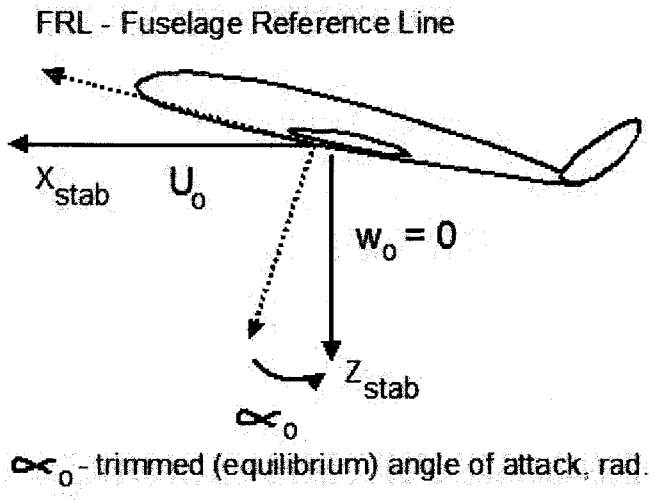


Fig. 3.2. The stability or wind axis system has its X-axis aligned with the total forward velocity of the aircraft. By definition, the equilibrium downward velocity w_o is 0.

The stability axis is a body fixed axis system. It is an Eulerian frame of reference. It is not fixed to the horizon, like a Newtonian frame of reference.

At a cruising flight condition, the difference between the stability/wind X-axis system and the FRL is often negligible. This is because the equilibrium, or trimmed, angle of attack is small.

At a landing flight condition, the difference between and the FRL will be significant. The angle of attack can become quite large at a landing configuration.

Typically, aircraft are designed to be handle well while cruising. A secondary design goal is good handling at landing. Consequently, some of the most difficult aircraft control problems occur at landing. This can become very frustrating for control engineers. When performance is most vital, the aircraft has slow and unstable transmission zeros.

Below are the linearized lateral equations of motion in the stability axis system. State variables are β, sideslip, p, roll rate, r, and bank angle ϕ. We will expect that $\theta_o = \alpha_o \approx 0$.

$$\dot{\beta} = Y_v\beta - r + {}^{g}\!/_{U_o}\cos(\theta_o)\phi + \sum_{k=1}^{m} Y^*_{\delta_k}\delta_k$$

$$\dot{p} = L'_\beta\beta + L'_p p + L'_r r + \sum_{k=1}^{m} L'_{\delta_k}\delta_k$$

$$\dot{r} = N'_\beta\beta + N'_p p + N'_r r + \sum_{k=1}^{m} N'_{\delta_k}\delta_k$$

$$\dot{\phi} = p + r\tan(\theta_o)$$

$$\textit{trim} \quad \textit{condition:} \quad u = U_o, \theta = \theta_o, \phi_o = 0$$

$$L'_{(\,)} = \frac{L_{(\,)}}{1 - \frac{I_{xy}^2}{I_{xx}I_{zz}}} + \frac{I_{xz}}{I_{xx}}\frac{N_{(\,)}}{1 - \frac{I_{xy}^2}{I_{xx}I_{zz}}} \qquad N'_{(\,)} = \frac{I_{xz}}{I_{zz}}\frac{L_{(\,)}}{1 - \frac{I_{xy}^2}{I_{xx}I_{zz}}} + \frac{N_{(\,)}}{1 - \frac{I_{xy}^2}{I_{xx}I_{zz}}}$$

The numbers $Y_v, L'_\beta, L'_p, N'_p$ etc., are called aerodynamic stability derivatives. These numbers come from wind tunnel test, Computational Fluid Dynamics (CFD) data, or flight data. These stability derivatives are noted like $Y_v = \frac{1}{m}\frac{dY}{dv}$ [mass]$^{-1}$[Force/velocity]. Another example is $L_p = \frac{1}{I_{xx}}\frac{dL}{dp}$ [inertia]$^{-1}$[Torque / (rad./sec.)]. A special stability derivative is $Y^*_{\delta_k} = \frac{1}{m}\frac{dY}{d\delta_v}\frac{1}{U_o}$, which has a * superscript to indicate that it is divided by U_o.

Below are the linearized longitudinal equations of motion. State variables are Δu, perturbed forward velocity, w, downward velocity, q, pitch rate, θ, pitch angle. We will expect that $\theta_o = \alpha_o \approx 0$.

$$\Delta\dot{u} = X_u\Delta u + X_w w - g\cos(\theta_o) + \sum_{k=1}^{m} X_{\delta_k}$$

$$\dot{w} = Z_u\Delta u + Z_w w + U_o q - g\sin(\theta_o) + \sum_{k=1}^{m} Z_{\delta_k}$$

$$\dot{q} = M_u\Delta u + M_w w + M_q q + \sum_{k=1}^{m} M_{\delta_k}$$

$$\dot{\theta} = q$$

$$\textit{trim} \quad \textit{condition:} \quad u = U_o + \Delta u, \theta = \theta_o \approx \alpha_o$$

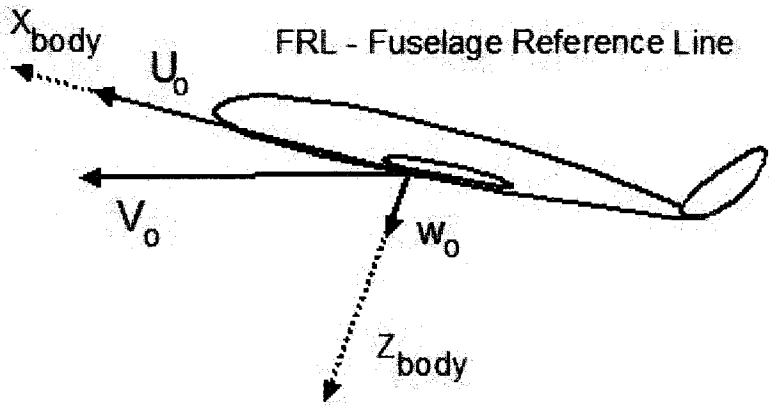


Fig. 3.3. Body axis system where the X-axis is aligned with the FRL.

At a cruising flight condition, the difference between the stability/wind axis system and the body axis system, covered in the next segment, is often negligible. This is because the equilibrium, or trimmed, angle of attack is small.

At a landing flight condition, the difference between the body axis system, covered in the next segment, and the stability landing

Wild trajectories may cause one to simply use the feedback variables of

$\dot{\beta} = {a_y^{cg}}/{U_o} - r + p\left({w}/{U_o}\right)$, or $\dot{\beta} = {a_y^{cg}}/{U_o} - r\cos(\alpha) + p\sin(\alpha)$ in place of $\dot{\beta} = {a_y^{cg}}/{U_o} - r$.

Alternatively, one may want to

$y = \dot{\beta} - \left(\frac{g}{U_o}\right)\cos(\theta)\sin(\phi) = {a_y^{cg}}/{U_o} - r + p\left(\frac{w}{U_o}\right) - \left(\frac{g}{U_o}\right)\cos(\theta)\sin(\phi)$. This way, one can still use linear controllers for non-linear trajectories.

Body Axis System (Body Fixed)

The idea of the body axis system is that the X-axis is aligned with the FRL.

The stability axis is a body fixed axis system. It is an Eulerian frame of reference. It is not fixed to the horizon, like a Newtonian frame of reference.

For further discussion on this matter, consult Ref. [3.1-3.4]

At a cruising flight condition, the stability axis and body axis should be pretty much the same assuming a small angle of attack. However, at a landing flight condition, they should be pretty different.

The equations of motion are much the same for the longitudinal system assuming small angle of attack. Conversions can be found in Refs. [3.1,3.2].

The lateral equations of motion are very similar, one would only add in an angle of attack term, where $\alpha_o \approx {}^{w_o}/_{U_o}$.

$$\dot{\beta} = Y_v\beta + p\,\boldsymbol{sin}(\alpha_o) - r\,\boldsymbol{cos}(\alpha_o) + {}^{g}/_{U_o}\,\boldsymbol{cos}(\theta_o)\phi + \sum_{k=1}^{m} Y^*_{\delta_k}$$

$$\dot{p} = L'_\beta\beta + L'_p p + L'_r r + \sum_{k=1}^{m} L'_{\delta_k}$$

$$\dot{r} = N'_\beta\beta + N'_p p + N'_r r + \sum_{k=1}^{m} N'_{\delta_k}$$

$$\dot{\phi} = p + r\,\boldsymbol{tan}(\theta_o)$$

Data for many aircraft in both the stability/wind and body axis can be found in Refs. [2.19,2.21]. Higher fidelity aerodynamic data for some fighter aircraft can be found in Refs. [3.5,3.6].

Lateral Modes (Eigenvalues and Eigenvectors)

1. Dutch Roll - a lightly damped oscillatory motion of the aircraft where the swings from side to side. The dutch roll typically involves larger amounts of sideslip, β, and yaw rate, r, than bank angle, ϕ. Example: $s = -0.1 \pm 2j$
2. Roll Subsidence - the bank angle ϕ just decays exponentially, this mode is always on the real axis. Example: $s = -3$
3. Spiral Root - Basically, this mode is involved with the aircraft slowly drifting to stay on course, or slowly drifting off course. This mode is typically located at s = 0. Some lateral control models neglect this entirely and just use 3 states [Klein]. These 3 state models neglect gravity and are thought to represent the system dynamics across a frequency range relevant to control. Example: $s = -0.0001, 0.002$

Longitudinal Modes (Eigenvalues and Eigenvectors)

1. Short Period Mode - a typically well damped quick oscillatory mode. Damping ratios are typically around, 0.5 or better open loop. This mode typically involves pitch rate, q and angle of attack, α. Example: $s = -1 \pm 1j$

2. Phugoid Mode - a lightly damped slow oscillatory mode with periods around 100 seconds. The aircraft tends to dive up and down. Damping ratios are typically around, 0.1 or less open loop. This mode typically involves forward velocity Δu and θ. Example: $s = -0.001 \pm 0.01j$

3.2 Low Frequency Non-Linear Effects

The rigid and linear equations of motion are quite good in the frequency range of at least 0.01–2.0 Hz for most aircraft [3.7]. Helicopters and rotorcraft may be a different issue. Small and rigid aircraft may be highly linear beyond 2.0 Hz, up to 5 or 10 Hz, where flexible modes become prominent. Large and flexible aircraft may have flexible modes as low as 1 Hz [3.8]. Even then, flexible modes are considered to be linear.

Regardless, there is perceived to be non-linearity at very low frequency. The author is unaware of the exact cause of this problem. One nonlinear contributor would be loss of mass due the burning of fuel. Inertial Navigation Systems (INS), which have slow drifts associated on the order of 0.01 degrees/hour. It could be due simply to non-linearities at the output like $a_y^{cg} = \dot{v} + ru - pw$, not $a_y^{cg} \approx \dot{v} + rU_o$. It could also be due to small approximations such as $\boldsymbol{sin}(\phi) \approx \phi$. It is difficult to say

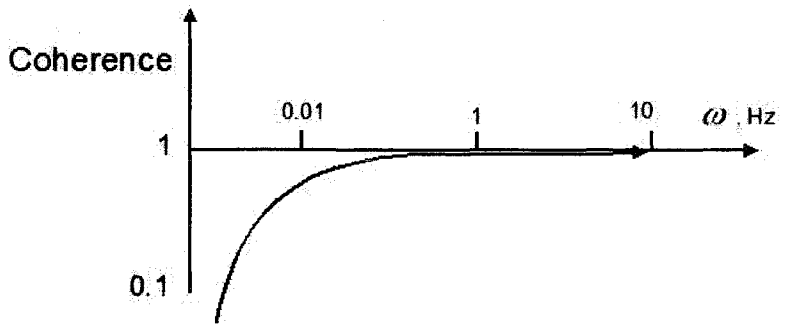


Fig. 3.4. The coherence is a measure of the linearity of a transfer function based purely on input-output data. A coherence value of 1 means that the input-ouput is linear.

exactly what it is.

The coherence is a measure of the linearity of a transfer function based solely upon input-output data [3.7]. We will show very briefly what a coherence plot from flight data might look like. This could

be because there is not enough flight data. Low frequency information requires a great deal of data. Getting the true dc gain of an aircraft would require an infinitely long test flight! This is not possible.

Here, $y(j\omega)=\int_0^{\infty} y(t)e^{-j\omega}dt \approx \sum_{k=1}^{N} y(k)e^{-j2k/T}$, where T would be the sampling frequency and $u(j\omega)=\int_0^{\infty} u(t)e^{-j\omega}dt \approx \sum_{k=1}^{N} u(k)e^{-j2k/T}$. Here, the * operator represents the conjugate transpose.

Hence, $u(j\omega)^* = u(-j\omega)$ and $y(j\omega)^* = y(-j\omega)$. Outputs and inputs are *y*, and *u*, respectively.

$$Coherence^2 = \frac{\left|y(j\omega)u(j\omega)^*\right|^2}{\left|y(j\omega)y(j\omega)^*\right|\left|u(j\omega)u(j\omega)^*\right|}$$

A plot of a typical coherence, from flight data is shown above. Notice that at low frequency, the coherence becomes quite poor. Sometimes during the development of an autopilot, integral tracking will be abandoned. These aircraft will require a pilot.

3.3 Non-Linear Equations for Aggressive Trajectories

Linear equations of motion will not be very accurate for aggressive and strange maneuvers where products such as, qr, p^2 or pw, become large. Many large transport aircraft are not made to accommodate these maneuvers and would be unrecoverable in the event that an overly ambitious pilot or autopilot flew such a highly non-linear trajectory.

Smaller aircraft made for "aerobatics" can sometimes handle more aggressive trajectories that include vertical loops and steep bank angles above 60 degrees. However, even with smaller aircraft, there is the potential for mishaps. Sometimes, these aircraft have short wings where I_{xx}/I_{zz} becomes small, this creates a problem known as "inertial coupling". $(I_{xx}-I_{zz})pr \Rightarrow$ pitching moment

It is not uncommon for the longitudinal system feed into lateral system. Lightweight aircraft sometimes have a tendency to yaw on approach as a result of changes in thrust and pitch [3.3]. We have also seen that the sine and cosine of angle of attack effected the linear lateral equations of motion in the body axis system.

This model was originally motivated specifically for damaged aircraft [1.19] when it was discovered that previous models [3.9,1.21] *were not summing forces and moments at the same location as the velocities, u,v and w*. This is not a unique error and can be found even in the F-16 model of Ref. [3.10]. Althouth the JSBSim code [3.11] cites [3.10], the author believes that this error is not present in the JSB code due to the special usage of the term "AERORP", an aero reference point. If one were to use the JSBSim code, one would use the mass properties about the c.g., not the reference point O.

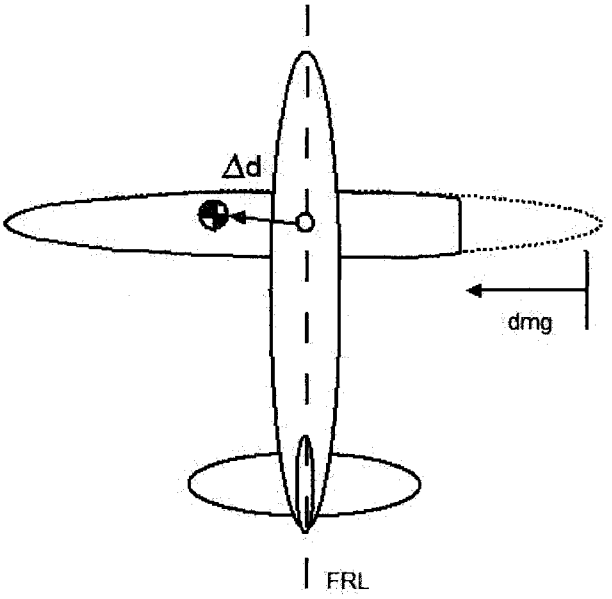


Fig. 3.5. Spanwise Full-Loss (SFL) wing damage definition. The wing is progressively removed from the tip to the root.

One should note that one can set certain parameters to 0, such as I_{xy}, I_{yz}, S_x, S_y, and S_z for normal operational of an aircraft.

We define the following mass properties *about the reference point O.*

$$S_x = m\Delta d_x, S_y = m\Delta d_y, S_z = m\Delta d_z$$

$$I_{xx} = \int_V \rho_{ac}\left(y^2 + z^2\right)dV \quad I_{yy} = \int_V \rho_{ac}\left(x^2 + z^2\right)dV \quad I_{zz} = \int_V \rho_{ac}\left(x^2 + y^2\right)dV$$

$$I_{xy} = \int_V \rho_{ac} xydV \quad I_{yz} = \int_V \rho_{ac} yzdV \quad I_{xz} = \int_V \rho_{ac} xzdV$$

All of these inertias exist *about the reference point O.* In most cases, the reference point O would be the quarter chord of the aircraft, i.e. the aerodynamic center. We use term, ρ_{ac}, to denote the density per unit volume of the aircraft.

Below are the generalized non-linear equations of motion about an arbitrary reference point O, which may not be the same as the c.g.

$$m(\dot{u}-rv+qw)+S_z\dot{q}-S_y\dot{r}+S_z pr+S_y pq-S_x q^2-S_x r^2=X_{gravity}+\sum X_{aero}$$
$$m(\dot{v}+ru-pw)-S_z\dot{p}+S_x\dot{r}+S_z qr+S_x pq-S_y p^2-S_y r^2=Y_{gravity}+\sum Y_{aero}$$
$$m(\dot{w}+pv-qu)+S_y\dot{p}-S_x\dot{q}+S_x pr+S_y qr-S_z p^2-S_z q^2=Z_{gravity}+\sum Z_{aero}$$

We remain silent on the aerodynamic forces, $\sum X_{aero}, \sum Y_{aero}, \sum Z_{aero}$. These forces come from some knowledge of aerodynamics. However, we note that these should be in a unit system of force, i.e. Newtons or lbf. The same is true of aerodynamic moments $\sum L_{aero}, \sum M_{aero}, \sum N_{aero}$ and should be in units of Nm or lbf.ft.

$$\begin{aligned}
&I_{xx}\dot{p}-I_{xy}\dot{q}-I_{xz}\dot{r}+I_{xy}pr-I_{xz}pq+(I_{zz}-I_{yy})qr+I_{yz}(r^2-q^2)\\
&-S_z(\dot{v}+ru-pw)+S_y(\dot{w}+pv-qu)=L_{gravity}+\sum L_{aero}\\
&-I_{xy}\dot{p}+I_{yy}\dot{q}-I_{yz}\dot{r}+I_{yz}pq-I_{xy}qr+(I_{xx}-I_{zz})pr+I_{xz}(p^2-r^2)\\
&+S_z(\dot{u}+qw-rv)-S_x(\dot{w}+pv-qu)=M_{gravity}+\sum M_{aero}\\
&-I_{xz}\dot{p}-I_{yz}\dot{q}+I_{zz}\dot{r}+I_{xz}qr-I_{yz}pr+(I_{yy}-I_{xx})pq+I_{xy}(q^2-p^2)\\
&+S_x(\dot{v}+ru-pw)-S_y(\dot{u}+qw-rv)=N_{gravity}+\sum N_{aero}
\end{aligned}$$

The forces and moments from gravity are listed below.

$$\vec{F}_{gravity}=\begin{bmatrix}X_{gravity}\\Y_{gravity}\\Z_{gravity}\end{bmatrix}=\Gamma^{-1}\begin{bmatrix}0\\0\\mg\end{bmatrix}=\begin{bmatrix}-mgs\theta\\mgs\phi c\theta\\mgc\phi c\theta\end{bmatrix}$$

$$\vec{M}_{gravity}=\begin{bmatrix}L_{gravity}\\M_{gravity}\\N_{gravity}\end{bmatrix}=\frac{1}{m}\tilde{S}\begin{bmatrix}X_{gravity}\\Y_{gravity}\\Z_{gravity}\end{bmatrix}=\begin{bmatrix}S_y gc\theta c\phi-S_z gc\theta s\phi\\-S_x gc\theta c\phi-S_z gs\theta\\S_y gs\theta+S_x gc\theta s\phi\end{bmatrix}$$

$$\Gamma^{-1}=\begin{bmatrix}c\theta c\psi & c\theta s\psi & -s\theta\\(-c\phi s\psi+s\phi s\theta c\psi) & (c\phi c\psi+s\phi s\theta s\psi) & s\phi c\theta\\(s\phi s\psi+c\phi s\theta c\psi) & (-s\phi c\psi+c\phi s\theta s\psi) & c\phi c\theta\end{bmatrix}$$

Abbreviations are used here, $c\phi=\boldsymbol{cos}(\phi)$, $s\theta=\boldsymbol{sin}(\theta)$.

We list the kinematic navigation equations below.

$$\begin{Bmatrix}p\\q\\r\end{Bmatrix}=\begin{bmatrix}1 & 0 & -s\theta\\0 & c\phi & s\phi c\theta\\0 & -s\phi & c\phi c\theta\end{bmatrix}\begin{Bmatrix}\dot{\phi}\\\dot{\theta}\\\dot{\psi}\end{Bmatrix}$$

$$\begin{Bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{Bmatrix} = \begin{bmatrix} 1 & s\phi\tan\theta & c\phi\tan\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi\sec\theta & c\phi\sec\theta \end{bmatrix} \begin{Bmatrix} p \\ q \\ r \end{Bmatrix}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi - s\psi c\phi & c\psi s\theta c\phi - s\psi s\phi \\ s\psi c\theta & s\psi s\theta s\phi + c\psi c\phi & s\psi s\theta c\phi - c\psi s\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

These navigation equations are in a North-East-Down (NED) coordinate system. Altitude, $h \approx \int_0^t (U_o\theta(\tau) - w(\tau))d\tau$, should be the opposite of z, such that $\dot{h} = -\dot{z}$.

3.4 Actuator Effects

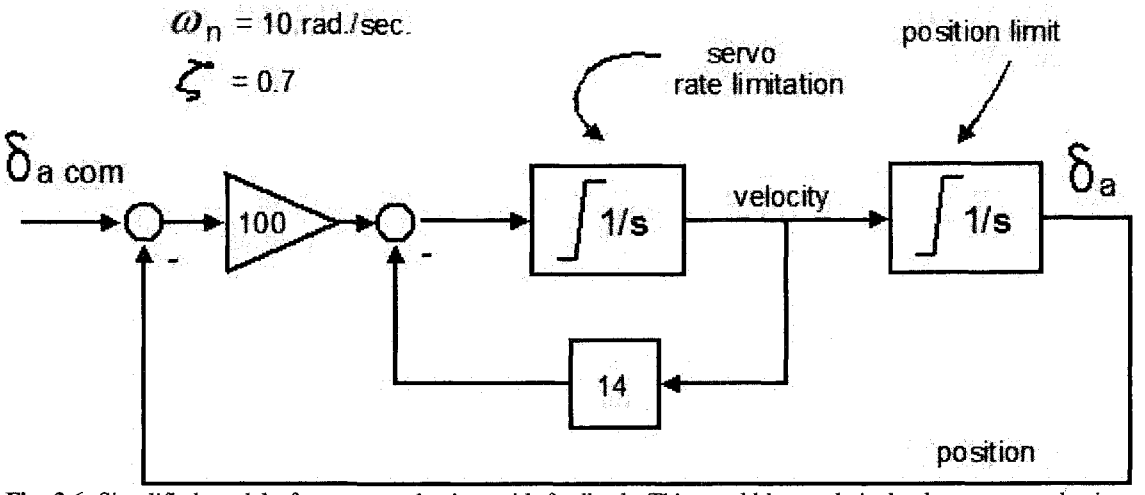


Fig. 3.6. Simplified model of a servomechanism with feedback. This would be a relatively slow servomechanism. Many servomechanisms have bandwidths approaching 60 rad./sec. or 10Hz.

Actuator effects play a significant role when hydraulic servomechanisms are used. There is usually what seems to be an "effective time delay" of approximately 0.1 to 0.2 seconds as a result of servo rate limitations. Hydraulic servomechanisms typically have rate limitations on the order of 40 to 60 degrees/sec.

Pilots and autopilots alike become very annoyed by these "time delays" from rate limitation.

The author is not an expert in the area of hydraulic servomechanism design, but they do actually have fast time delays on the order of several milliseconds which are related to fluid transport. The most recent servomechanisms themselves have their own control loops which achieve bandwidths of about

10Hz or 60 rad./sec. Therefore, we know that these servomechanisms cannot actually have true time delays at 0.1 to 0.2 seconds. The bandwidth of a control system in rad./sec. must always be slower than 2/(time delay, sec).

A really good visualization of servo rate limitations comes from Ref. [3.12] and is shown below.

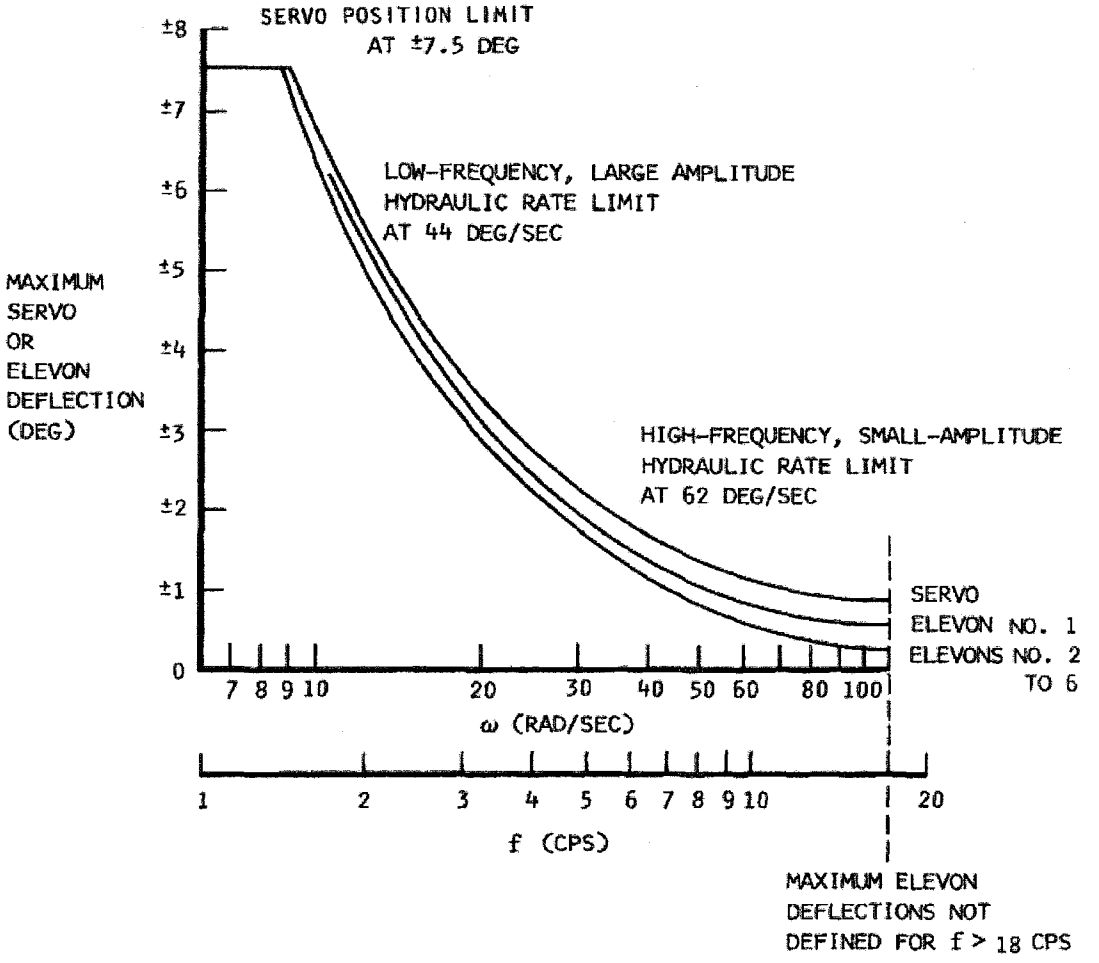


Fig. 3.7. "Bode plot" of servomechanism limitation on XB-70 aircraft. Taken from Ref. [3.12] and is used with permission from NASA.

3.5 Handling Qualities

There are a total of 9 handling qualities metrics [3.13], most of which are reasonably accurate predictors of Pilot Induced Oscillation (PIO). We will only discuss one particular metric for gauging pilot

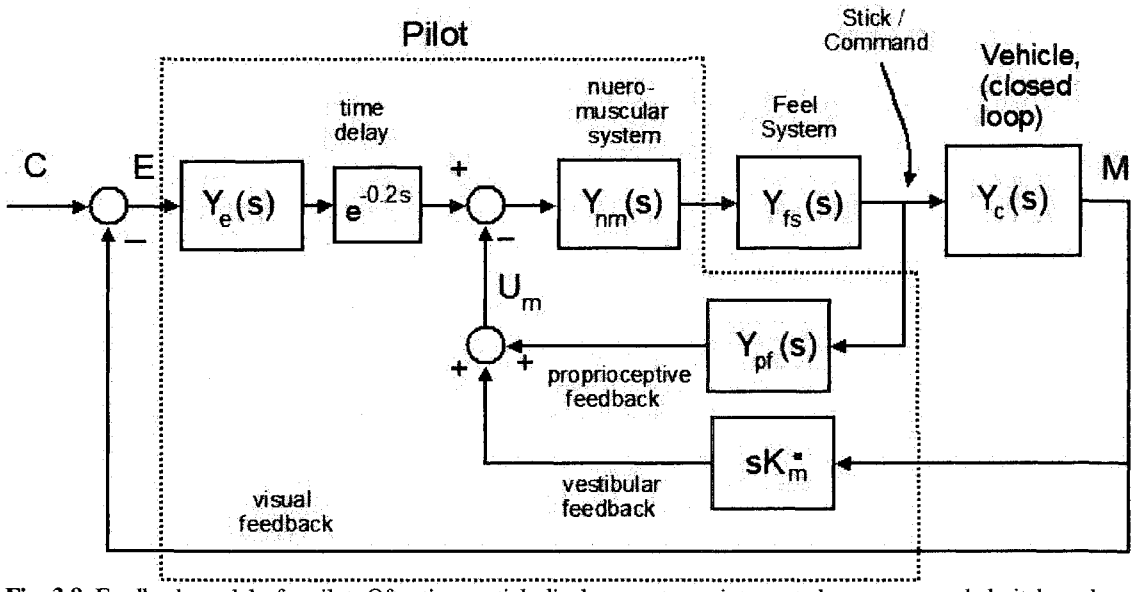


Fig. 3.8. Feedback model of a pilot. Oftentimes, stick displacements are interpreted as a commanded pitch angle or bank angle.

handling qualities from ref [3.14].

First, one needs to select a signal, M, to feedback. This might be pitch attitude, θ, or bank angle, ϕ, or some other signal.

The following transfer functions are described below.

$$Y_{nm}(s) = \frac{10^2}{\left(s^2 + 14.14s + 10^2\right)}$$

$$Y_{pf}(s) = K(s+a), \quad K, \quad K/(s+a)$$

$$K_{\dot{m}} = 0 \quad \text{typically}$$

We give no information on $Y_{fs}(s)$. One might set it simply to $Y_{fs}(s) = 1$ in the absence of information.

Procedures to assess handling qualities are listed below.

1. Choose K in $Y_{pf}(s)$, such that a minimum damping ratio of 0.15 is achieved in the closed loop system.
2. Design $Y_e(s)$ such that a bandwidth of 2.0 rad./sec. is achieved. This should typically be a PI controller. $Y_e(s) = \dfrac{K_e(s+1)}{s}$
3. The Handling Qualities Sensitivity Function (HQSF) is defined below.

$$HQSF(j\omega) = \frac{1}{|K_e|}\left|\frac{U_M}{C}(j\omega)\right|$$

Next one would want to plot the HQSF vs. frequency to gauge the pilot opinion rating according to the following plot below.

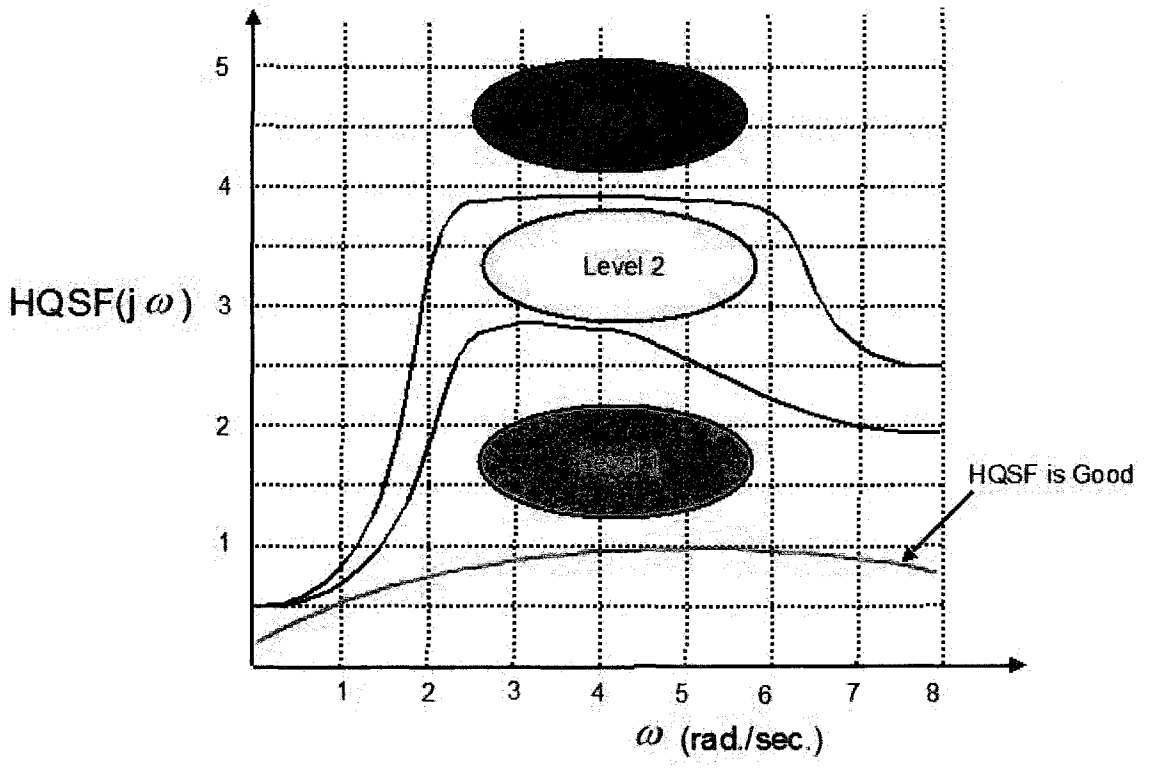


Fig. 3.9. Handling Qualities Sensitivity Function (HQSF) vs. frequency. Taken from Ref. [3.14] and used with permission.

3.6 Gain Scheduling

An aircraft with large variations in altitude and airspeed typically has a controller which is varied based on velocity and altitude. Alternatively, one can use impact pressure, also called dynamic pressure $\bar{q} = \frac{1}{2}\rho_{air}U_o^2$.

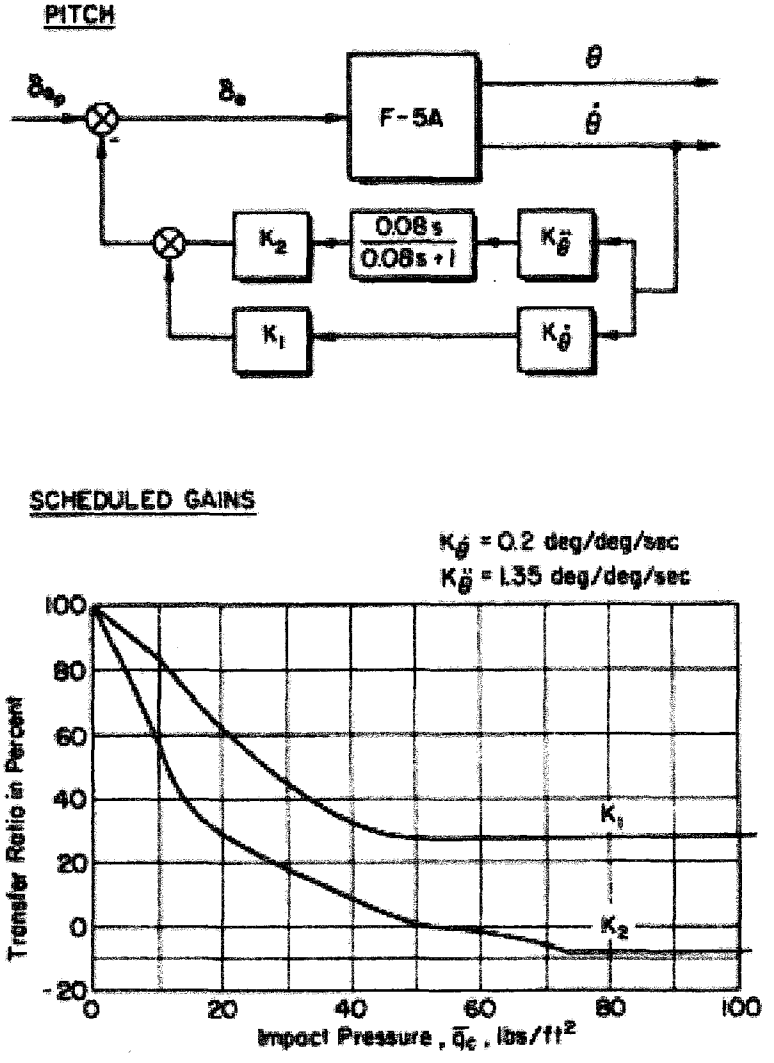


Fig. 3.10. This regulator is scheduled based upon impact pressure, also called dynamic pressure $\bar{q} = \frac{1}{2}\rho_{air}U_o^2$. Notice that as dynamic pressure increases, the gain decreases. This figure is taken from Ref. [2.19] and is used with permission from NASA.

Design procedure for a gain scheduled controllers are as follows below.

1. Get linear dynamics at a given altitude and airspeed or dynamic pressure.
2. Design the controller for *that altitude and* airspeed or dynamic pressure. Use this controller at this altitude and airspeed.
3. Interpolate between altitudes and airspeeds.

A gain scheduled regulator for the F-5 aircraft is shown in Fig. 3.10 [2.19].

Notice that as dynamic pressure increases, the gain decreases across a region of slow airspeed. At sea-level, a dynamic pressure of 60 psf (pounds per square foot) corresponds to an airspeed of 210 fps (feet per second).

3.7 Flexible Effects

All aircraft have flexible dynamics at some frequency, some are high enough that they can be neglected. Small and lightweight aircraft often have structural modes above 10 Hz. These modes can be mostly neglected if the digital sampling rate is at 0.1 seconds, which is a fairly popular digital sampling rate for aircraft control. For these aircraft, one mostly needs to keep in mind aliasing issues with digital control.

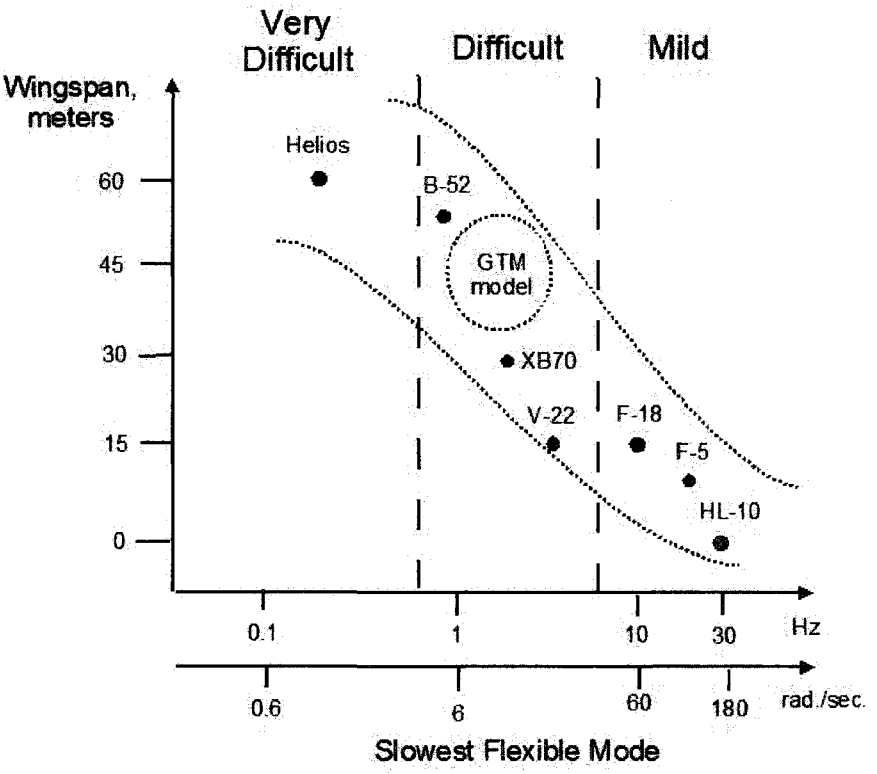


Fig. 3.12. Data on flexible modes compiled from various sources. Notice that aircraft with large wingspan have low frequency flexible modes and aircraft with shorter wingspans have faster flexible modes. Bear in mind that flexible modes shift in frequency depending mostly upon loading configuration. Adding cargo or fuel will slow down the structural vibrations. Removing cargo or fuel will speed up those flexible modes. Also, adding massive propellers on the ends of the wings will reduce flexible frequencies as well, as with the V-22.

Significant challenges are encountered when dealing with large transport aircraft like the Airbus A300 series and Boeing 700 series. These aircraft loosely fall within a set of mildly variable numbers which is termed a Generic Transport Model (GTM) model.

Inevitably, sensors will pick up engine noise and other high frequency disturbances independent of the aircraft model. This is a very mild challenge when dealing with fast flexible modes above 8-10Hz. However, when dealing with large aircraft, the flexible modes can be as low as 1 Hz.

Ideally, one would want the first flexible mode to be at least a factor of 10 faster than the bandwidth of the control system.

Large aircraft can have flexible modes which are a factor of 10 slower than lightweight aircraft. However, the bandwidth requirements of the control system are only reduced by a factor of 2 or 3 at most. Therefore, controllers for large aircraft cannot enjoy the luxury of robustness to large uncertainties in flexible dynamics that small aircraft do.

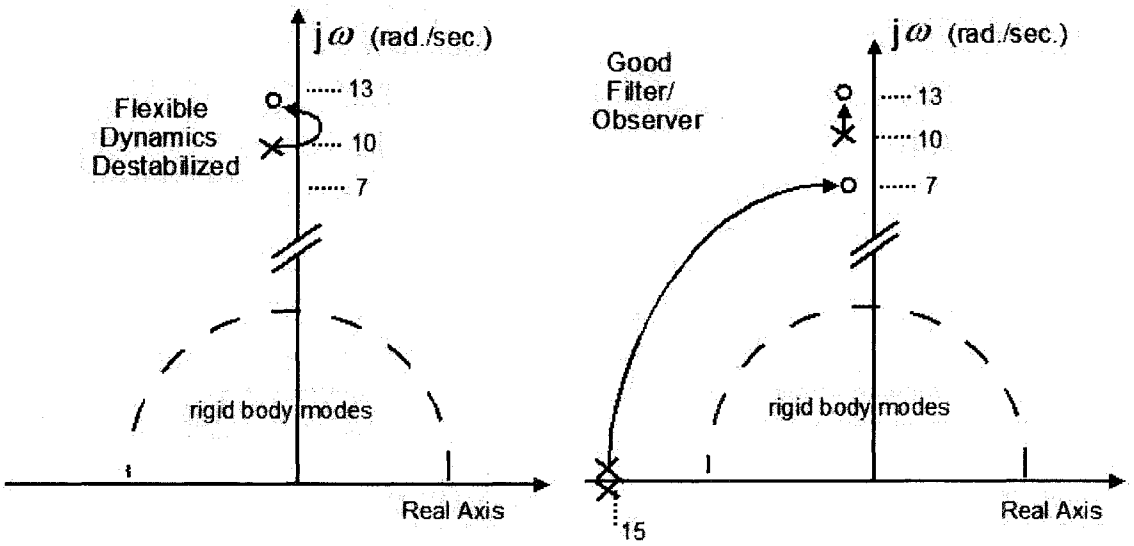


Fig. 3.13. Notch filtering which is robust to frequency shifts of the flexible modes is important.

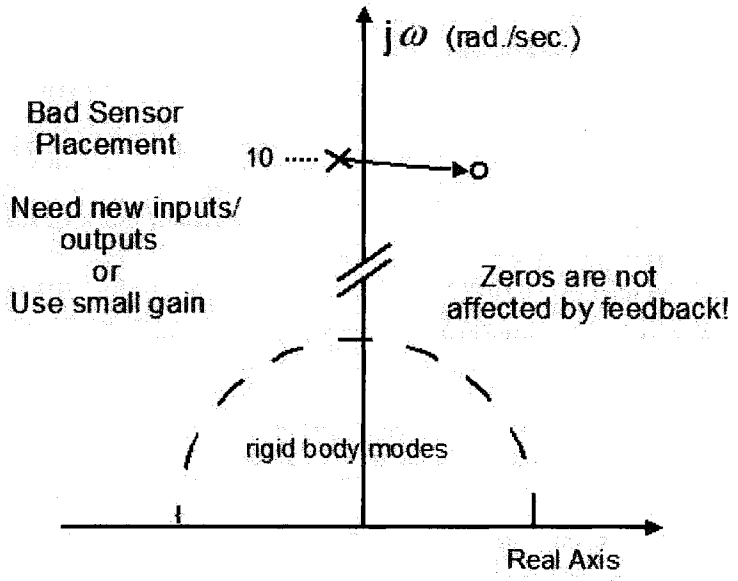


Fig. 3.14. Zeros are not affected by feedback at all. One needs to get different inputs or outputs in this case or use small gain.

Robustness issues always become noticeable with poles and zeros near the $j\omega$ axis. Large frequency shifts can cause instability when filters are involved. It matters little whether one uses a notch filter [3.15] or whether one uses an observer based controller such as a Loop Transfer Recovery (LTR). A filter is a filter. A controller is a controller. A malicious shift in frequency of the flexible dynamics could destabilize the closed loop. Whether or not this is physically possible is an important thing to investigate.

Almost any feedback system with poles and zeros near the $j\omega$ axis has an increased risk of instability, especially if there are unstable or non-minimum phase (NMP) zeros near the $j\omega$ axis. Therefore, it is a challenging task to design a robust controller for large transport aircraft with uncertain flexible dynamics. Accurate modeling of flexible dynamics is an important task for large transport aircraft.

In the event that zeros appear on the right half plane (RHP) of the $j\omega$ axis, filtering will really not prevent instability. Zeros are not affected by feedback, refer to section 2.19 on this matter. The only solution to this problem is to look at different inputs or different outputs.

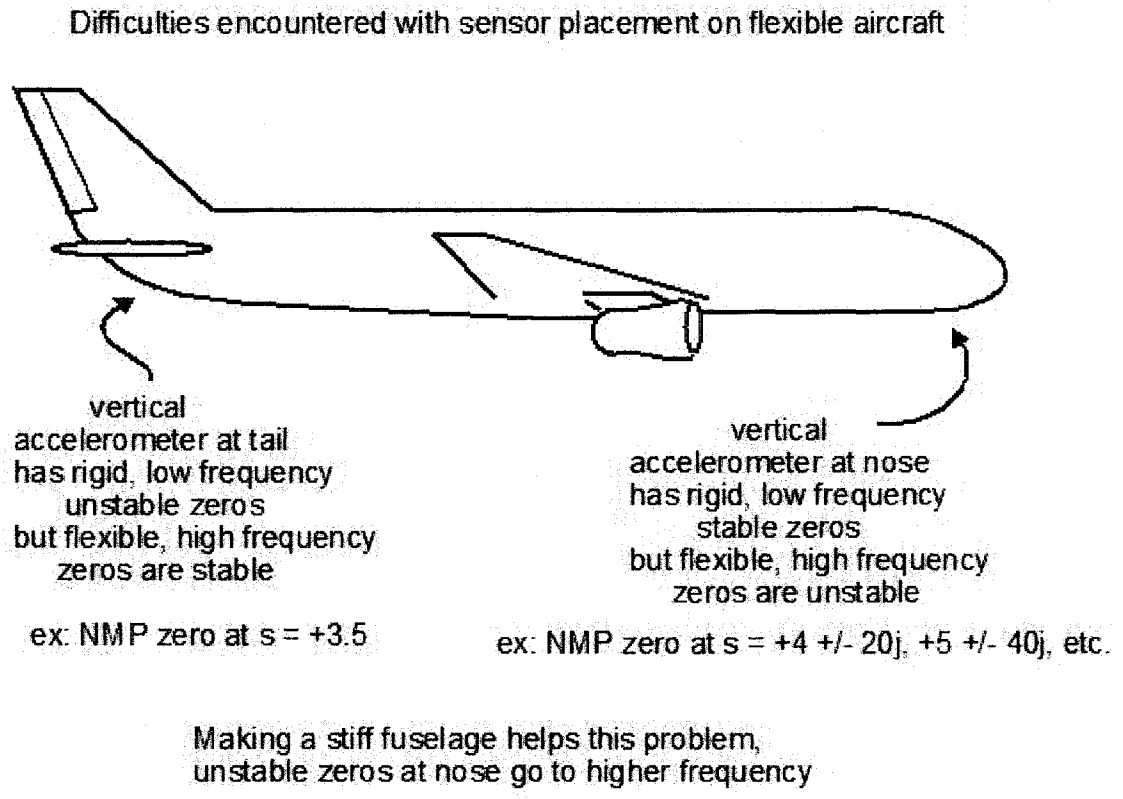


Fig. 3.15. Technically, these zeros may not be transmission zeros. However, blending these two sensors together does not really provide that great a benefit.

Sensor placement for a large transport can become very tricky if the fuselage is not stiff enough. An advancement over the past several decades involves an analytical proof showing that a co-located actuator-sensor pair on a flexible system has stable zeros. This means that if you put an accelerometer near a force actuator, then you can use high gain feedback with a flexible system.

For an aircraft with an elevator and not a canard, this means that you would be putting vertical accelerometers at the tail. However, this is not good for rigid body modes. With almost any purely rigid model of an aircraft with an elevator, feedback of a vertical accelerometer placed at the tail into the elevator would be destabilizing due to an unstable zero in the RHP.

Mild acceleration feedback is sometimes placed at the nose of a rigid aircraft via a bobweight. This is not a big issue since rigid zeros are stable. However, high frequency zeros which are closely related to flexible modes are usually unstable when one uses an accelerometer placed at the nose. Having a stiff fuselage helps significantly with this issue.

Other authors have found unstable zeros when dealing with flexible dynamics [3.16,3.17]. However, sensor placement was not investigated. Experimental sensor placement and blending techniques are discussed in Ref. [3.12].

We outline the following extra challenges associated with feedback control of *stable* flexible dynamics.

1. Sensors will detect flexible modes. If the flexible modes are below 5 or 10 Hz, filtering those flexible modes becomes an extra burden. A notch filter or an observer may be used. If a digital sampling frequency of 10 Hz is used and there is structural mode at 5 Hz, filtering may suffer. One may resort to analog filters in this case. A gain margin of 8dB is typically required for notch filter design [3.15]. If the notch filter is in error by as little as 1 Hz, then the feedback loop might become destabilized [3.18]. Incidentally, the use of notch filters has also been shown to adversely affect handling qualities [3.19].
2. Sensor placement for flexible aircraft with slow structural modes is a significant challenge. Sometimes, one may abandon this altogether and simply use feedback from Inertial Navigation System (INS) which is typically located near the c.g. of the aircraft.
3. Ambitious goals can be to use high gain feedback to add damping to flexible modes. In this case, one would want to be very proficient in the field of modeling Aero-Servo-Elasticity (ASE) systems. It is difficult to pronounce and it is also difficult to understand. Furthermore, ASE should incorporate sensors dynamics as well. It really

should be Aero-Servo-Elasticity and Sensors (ASES). Sensor dynamics may become an issue when getting into the area of high frequency dynamics.

3.8 Modal Analysis and Dynamics in a Vacuum

This section will discuss only the single aspect of elasticity, not aero-elasticity. We will assume that there is no stiffness added by aerodynamics.

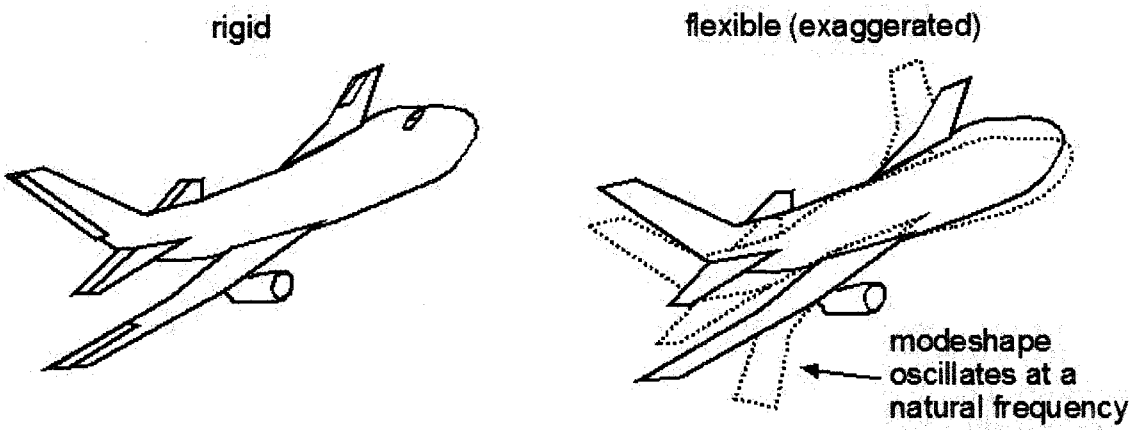


Fig. 3.16. A flexible aircraft has a modeshape that oscillates at a natural frequency

We will briefly discuss modal analysis techniques via finite element analysis (FEA) [3.20-3.23]. The idea of FEA is that you chop the solid into little pieces, called elements. Each element has its own mass matrix and stiffness matrix, K^e_{stiff}, and M^e, associated with it. The figure below has "plate" elements for the wing.

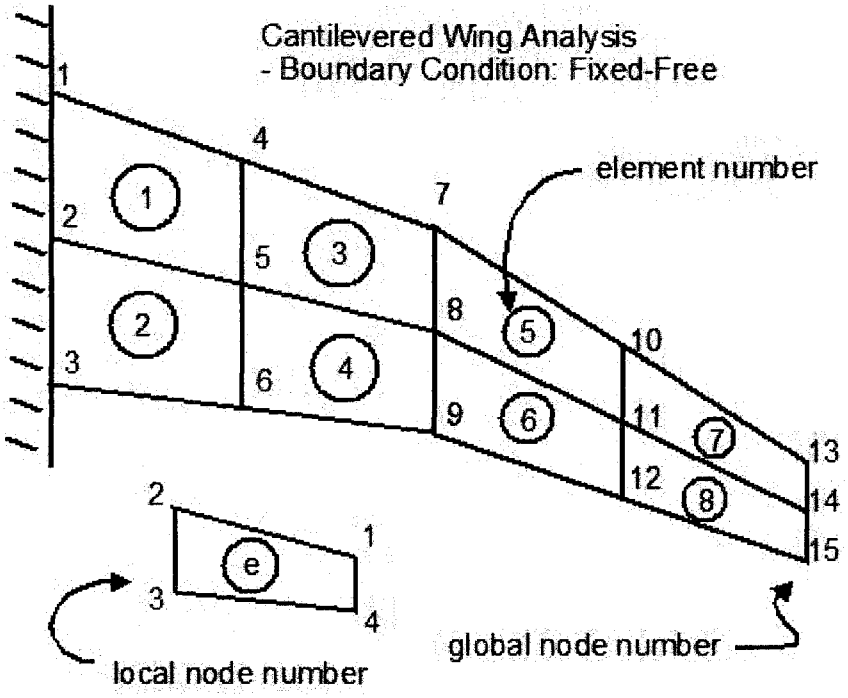
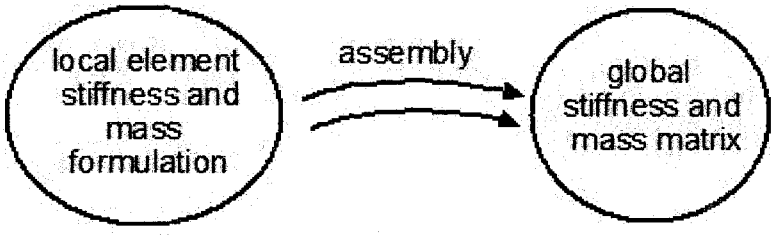


Fig. 3.17. Finite element analysis of a cantilevered wing. Many studies are conducted in this manner.



-element 5 has global nodes 10,7,8,11
-add local stiffness 1,2,3,4 to global stiffness locations 10,7,8,11
-element 6 has global nodes 11,8,9,12
-add local stiffness 1,2,3,4 to global stiffness locations 11,8,9,12
-element 7 has global nodes 13,10,11,14
-add local stiffness 1,2,3,4 to global stiffness locations 13,10,11,14
-etc.

Fig. 3.18. The assembly procedure for the global stiffness matrix

The figure above shows a cantilevered wing broken up into "plate" finite elements. Most aeroelasticity problems focus on a single wing in this configuration. The wing is assumed to be rigidly attached to fuselage.

From this figure, we can see that element number 5 has global nodes 7,8,10,11 associated with it. We also say that every element, including

element 5, has local nodes 1,2,3,4. One needs a mapping technique to get from an elemental stiffness to a global stiffness for the entire structure. An example of the global stiffness assembly procedure for the wing is shown to the above. Assembly procedures for the global mass matrix are exactly the same.

Now, once the global stiffness, K_{global}, and mass matrices, M_{global}, have been assembled, then we can check the natural frequencies and get to modeshapes.

The natural frequencies squared will be eigenvalues of $M_{global}^{-1}K_{global}$.

$$\begin{aligned}
&\boldsymbol{det}\left(s^2 I + M_{global}^{-1}K_{global}\right)=0\\
&roots \quad are: s^2 = -100,-400,-900,... = -\omega_{n1}^2,-\omega_{n2}^2,-\omega_{n3}^2,...\\
&or\\
&solve \quad eigenvalues \quad \lambda\left[-M_{global}^{-1}K_{global}\right]=-\omega_{n1}^2,-\omega_{n2}^2,-\omega_{n3}^2,...
\end{aligned}$$

The modeshapes will correspond to eigenvectors. Sometimes, one would be looking at solving for natural frequencies alone. However, for sensor placement, one would be interested particularly in modeshapes.

$$solve \quad eigenvectors \quad \Phi_{\bullet,ni} = null\left[\begin{bmatrix} \omega_{ni}I & -I \\ M_{global}^{-1}K_{global} & \omega_{ni}I \end{bmatrix}\right]$$

If the boundary condition is free-free, then K_{global} and M_{global} will always be symmetric. The boundary condition for the entire aircraft is free-free.

1. The stiffness matrices, K_{global}, and mass matrices, M_{global} are symmetric.
2. Because, $M_{global}^{-1}K_{global}$, is symmetric, the modeshapes are orthogonal to each other $\Phi^T\Phi = I$. Quick proof: $A = A^T, VDV^{-1} = V^{-T}DV^T, V^{-1} = V^T$, V are the eigenvectors, D is a diagonal matrix of eigenvalues.
3. For the free-free boundary condition, there is a set of rigid body modes that correspond to natural frequencies of 0. Like all other modes, they are orthogonal to each other.

Aside from a higher fidelity supersonic beam example shown in sections 7.1 through 7.6, we will be using a low fidelity beam element analysis technique for our flexible dynamics study. However, we will be solving both the wings and the fuselage with these beam elements.

Below is a graphic of a finite beam element analysis done as a feasibility study for a transonic bi-plane [3.23]. We will be using much of the same style in sections 7.7 through 7.14.

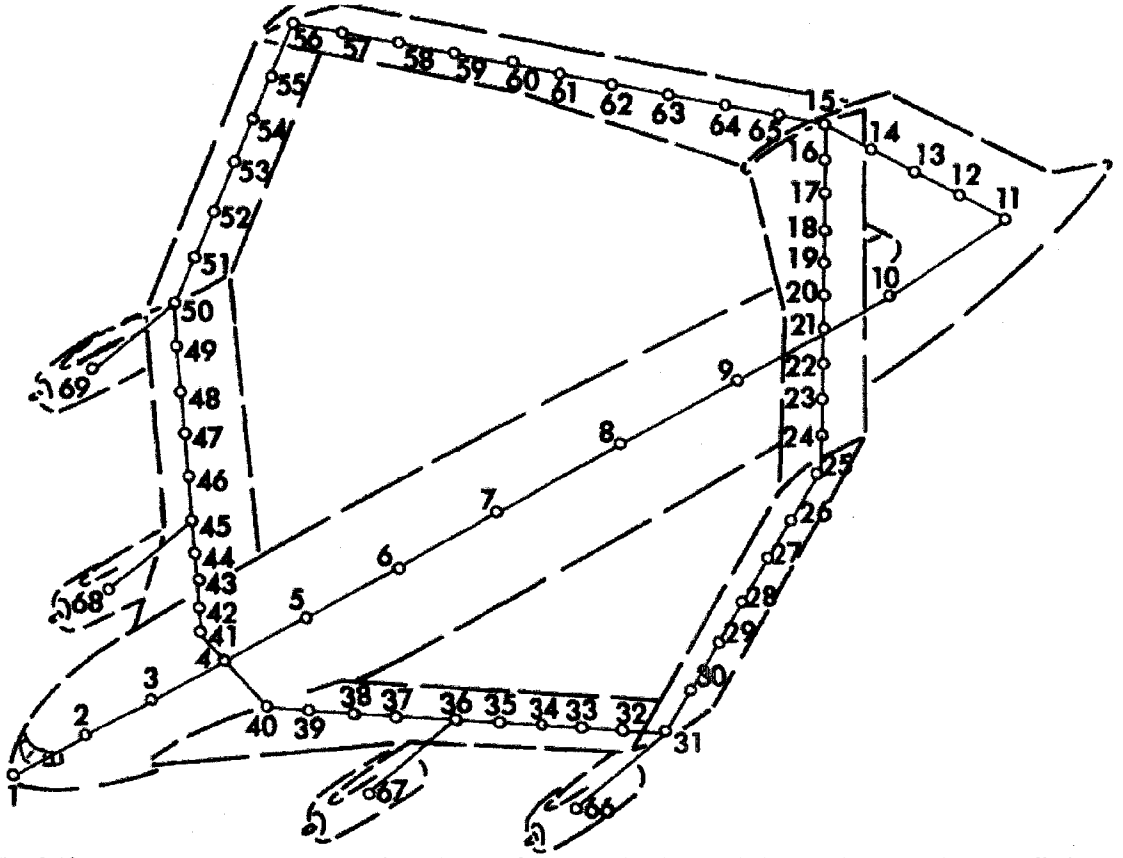


Fig. 3.19. Approximate beam element formulation of a transonic biplane. This model was used as a preliminary study which showed that the transonic biplane was not feasible. Taken from Ref. [3.23] and used with permission from NASA.

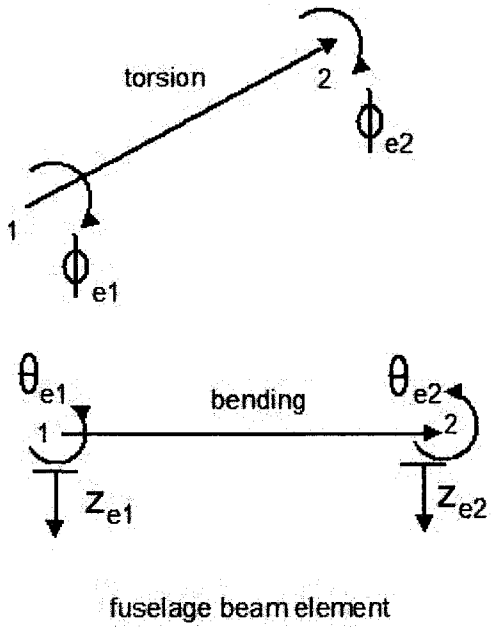


Fig. 3.20. Fuselage beam element with twist and bending.

A graphic of the basic beam element one would use for the fuselage is shown to the left. The beam element can bend elastically in pitch, θ_e, twist elastically in roll, ϕ_e, or plunge vertically downward, z_e. The beam element is shown to the left.

This model is a simple 3-degree of freedom (dof) beam element and does not have distortions in yaw. Including this effect would would mandate at least a 5-dof beam element. However, this 5-dof can sometimes become difficult to mate this with other beam elements. To capture these effects, the author recommends using the full 6-dof beam found in Ref [3.23].

Although there are still some who use beam elements for structural aircraft analysis, they are not considered to be highly accurate and should be used for coarse feasibility studies.

The beam elemental stiffness is shown below for the 3-dof case.

$$L_e = element \quad length \quad EI_e = bending \quad stiffness \quad GJ_e = torsional \quad stiffness$$

$$K^e_{stiff} = \begin{bmatrix} \frac{12EI_e}{L_e^3} & \frac{6EI_e}{L_e^2} & 0 & \frac{-12EI_e}{L_e^3} & \frac{6EI_e}{L_e^2} & 0 \\ \frac{6EI_e}{L_e^2} & \frac{4EI_e}{L_e} & 0 & \frac{-6EI_e}{L_e^2} & \frac{2EI_e}{L_e^2} & 0 \\ 0 & 0 & \frac{GJ_e}{L_e} & 0 & 0 & \frac{-GJ_e}{L_e} \\ \frac{-12EI_e}{L_e^3} & \frac{-6EI_e}{L_e^2} & 0 & \frac{12EI_e}{L_e^3} & \frac{-6EI_e}{L_e^2} & 0 \\ \frac{6EI_e}{L_e^2} & \frac{2EI_e}{L_e^2} & 0 & \frac{-6EI_e}{L_e^2} & \frac{4EI_e}{L_e} & 0 \\ 0 & 0 & \frac{-GJ_e}{L_e} & 0 & 0 & \frac{GJ_e}{L_e} \end{bmatrix} \quad x^e = \begin{bmatrix} z_{e1} \\ \theta_{e1} \\ \phi_{e1} \\ z_{e2} \\ \theta_{e2} \\ \phi_{e2} \end{bmatrix}$$

Basically, we will be approximating the aircraft as a two sticks, neglecting wing sweep, horizontal and vertical tail deflections. Although this may seem to be too great a simplifying assumption,

we will be tuning the stiffness of the airplane such that the first flexible mode will be at 1 to 2 Hz. This will make the control design significantly difficult alone.

A figure of the beam approximation is shown below. Bear in mind that even with this simple approximation, control can become challenging.

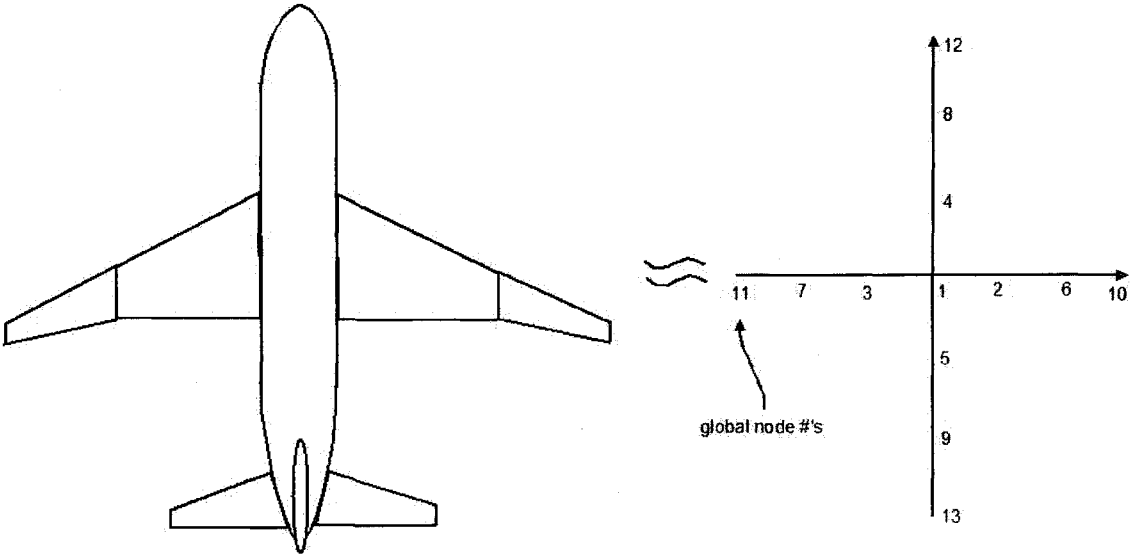


Fig. 3.21. We will be approximating the aircraft as a two sticks, neglecting wing sweep, horizontal and vertical tail deflections. Although this may seem to be too great a simplifying assumption, we will be tuning the stiffness of the airplane such that the first flexible mode will be at 1 to 2 Hz. This can make the control design significantly difficult.

Provided that the structural modes are decoupled, which is always true for a solid floating freely in a vacuum, a *mean axis* assumption [3.25] is made which is sufficiently accurate for any reasonable control design. The mean axis is a very reasonable assumption, even with ridiculous geometries. Section 7.1 uses a more technically complicated unified theory to validate the mean axis axis for a superonic beam with a 1% thickness over chord ratio.

The equations of motion under the *mean axis assumption* from Ref. [3.25] are listed in the equations below. The reference point for these equations of motion would be the center of gravity of the aircraft.

$$m_r\left(\dot{u}_r - r_r v_r + q_r w_r + g\,\boldsymbol{\sin}\,\theta_r\right) = \sum X_{aero}$$
$$m_r\left(\dot{v}_r - p_r w_r + r_r u_r - g\,\boldsymbol{\sin}\,\phi_r\,\boldsymbol{\cos}\,\theta_r\right) = \sum Y_{aero}$$
$$m_r\left(\dot{w}_r - q_r u_r + p_r v_r - g\,\boldsymbol{\cos}\,\phi_r\,\boldsymbol{\cos}\,\theta_r\right) = \sum Z_{aero}$$

$$I^r_{xx}\dot{p}_r - \left(I^r_{xy}\dot{q}_r + I^r_{xz}\dot{r}_r\right) + \left(I^r_{zz} - I^r_{yy}\right)q_r r_r + \left(I^r_{xy}r_r - I^r_{xz}q_r\right)p_r + \left(r_r^2 - q_r^2\right)I^r_{yz} = \sum L_{aero}$$
$$I^r_{yy}\dot{q}_r - \left(I^r_{xy}\dot{p}_r + I^r_{yz}\dot{r}_r\right) + \left(I^r_{xx} - I^r_{zz}\right)p_r r_r + \left(I^r_{xy}p_r - I^r_{yx}r_r\right)q_r + \left(p_r^2 - r_r^2\right)I^r_{xz} = \sum M_{aero}$$
$$I^r_{zz}\dot{r}_r - \left(I^r_{xz}\dot{p}_r + I^r_{yz}\dot{q}_r\right) + \left(I^r_{yy} - I^r_{xx}\right)p_r q_r + \left(I^r_{xz}q_r - I^r_{yz}p_r\right)r_r + \left(q_r^2 - p_r^2\right)I^r_{xy} = \sum N_{aero}$$
$$\ddot{\eta}_i + 2\zeta\omega_{ni}\dot{\eta}_i + \omega^2_{ni}\eta_i = \sum Q_{aero}$$

Here, u_r, v_r, w_r represent the rigid, body fixed velocities *at the center of mass*, in the x, y, and z axes respectively. Also, p_r, q_r, r_r represent the rigid, body fixed angular rates *at the center of mass*, in the x, y, and z axes respectively. Aerodynamic forces are represented as $\sum X_{aero}, \sum Y_{aero}, \sum Z_{aero}$ which are summed in the x, y, and z axes respectively. Aerodynamic moments are represented as $\sum L_{aero}, \sum M_{aero}, \sum N_{aero}$ which are summed about the x, y, and z axes respectively. The η_i terms represent modal coordinates of the flexible subsystem and the $\sum Q_{aero}$ term would represent aerodynamic forces and moments expressed in terms of modal coordinates.

3.9 Modal Analysis and Dynamics with Aerodynamics

This section will discuss the idea of aero-elasticity. One explanation of the influence of aerodynamics on flexible dynamics are the unsteady Aero-Influence Coefficients (AIC). The idea of the unsteady AIC is that they contribute stiffness to the elastic stiffness proportional to dynamic pressure, $\overline{q}$. The difficulty is that when a wing oscillates up and down, it induces *unsteady* flow.

Accurately solving *steady* aerodynamic flow is usually a reasonable task. Accurately solving *unsteady* aerodynamic flow is not usually a reasonable task. Therefore, some simplifying assumptions are needed. They are listed below.

1. Only the unsteady lift with an oscillating wing is of interest.
2. Unsteady drag is not of interest.

Quantifying unsteady lift was done both theoretically and experimentally early on in the development of aircraft [3.26,3.27]. These early developments led to Wagner's lift growth function and Kussner's lift growth function. Wagner's lift growth function, $W_{ag}(s)$ is show in the figure below.

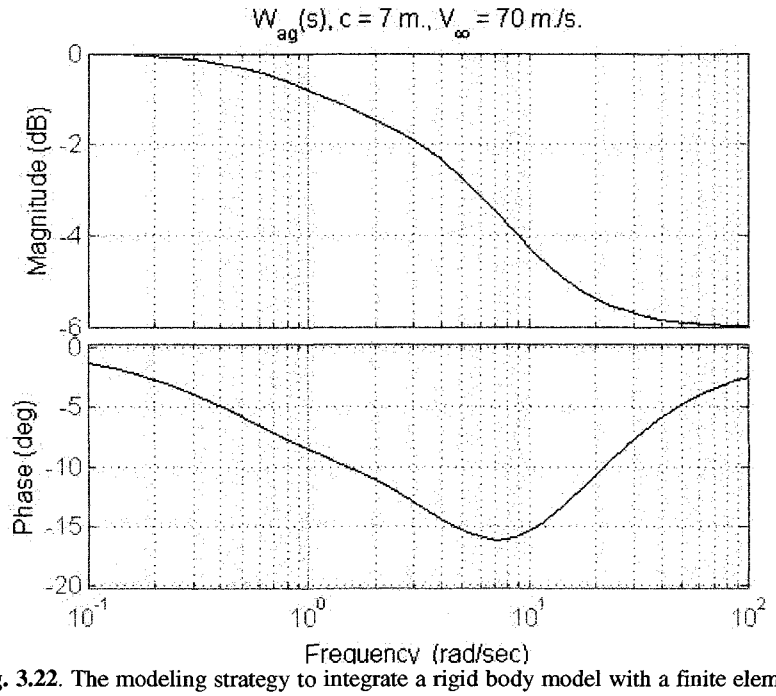


Fig. 3.22. The modeling strategy to integrate a rigid body model with a finite element aero-elastic model. It is assumed that there is a global

Many more nonlinear lift growth functions involving Bessel functions have been proposed starting in the 1950's, but despite these efforts, the frequency domain behavior is still pretty much the same as Wagner's lift growth function. Basically, there is a small dip in the phase by about 15 degrees and a high frequency gain of about 1/2. This means that the lift instantly rises up to half of its steady value and slowly grows up to its steady value.

Note that this does not take into account wing sweep or 3D aerodynamics. These early experiments only involved straight wings and lifting line theory. Accurately solving a 3D unsteady aerodynamics problem with moving boundary conditions (i.e. the wing) is a very ambitious task. Nonetheless, software is available that claims it can do this thing. The ZONATECH corporation has some proprietary numerical technique for unsteady AIC formulation. Today, many researchers refer to commercial software to figure out these unsteady aerodynamics for complex problems [3.17,3.28]. Over the past several decades, knowledge upon unsteady AIC formulation and aeroelasticity in general has been almost entirely commercialized. The author is not aware of any convenient or reasonably comprehensible table one can find to look up these unsteady AIC numbers even given reasonable

geometry factors like wing sweep or dihedral. Control engineers may find themselves utterly lost in a field where there are simply no comprehensible answers available.

Due to this overall lack of comprehensible knowledge upon unsteady AIC for swept wings, wing sweep will be neglected. Wagner's lift growth function can be used to get a reasonable approximation for the flexible model as shown below.

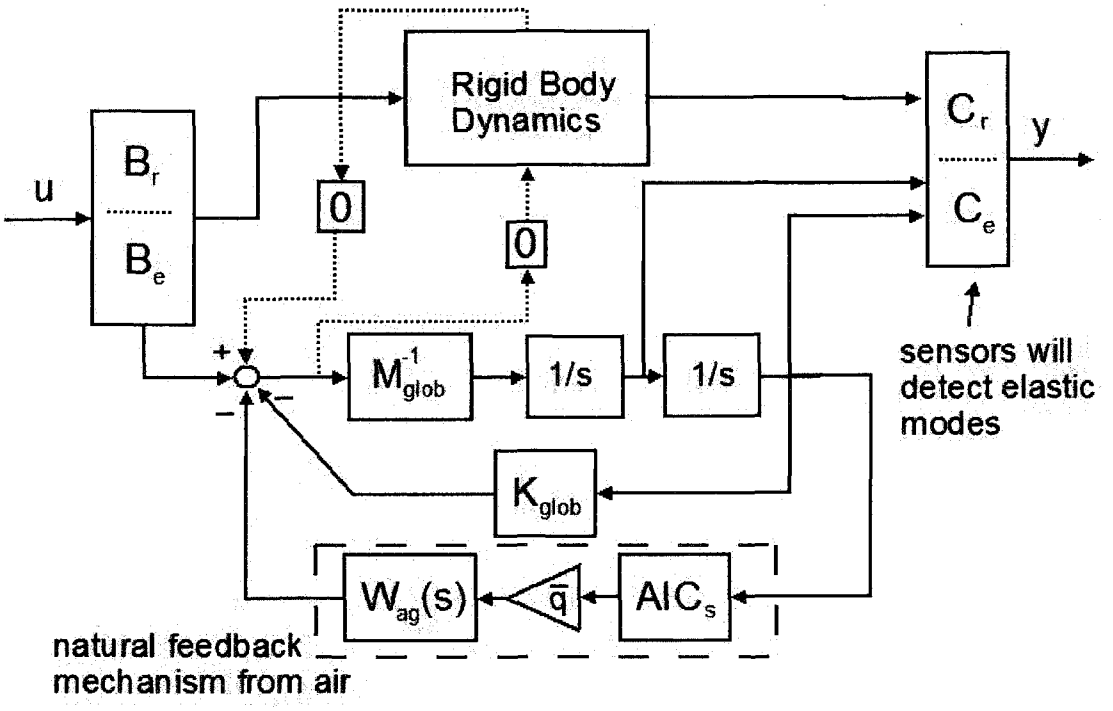


Fig. 3.23. The modeling strategy to integrate a rigid body model with a finite element aero-elastic model. It is assumed that there is a global mass matrix, M_{glob}, a global stiffness matrix, K_{glob}, a steady aerodynamic influence coefficient matrix, AIC_s, dynamic pressure, $\bar{q}$. The dynamic pressure, $\bar{q}$, is just a scalar. The author decided to use Wagner's lift growth function, $W_{ag}(s)$, to approximate unsteady aerodynamic lift generation. The B matrix is partitioned into an elastic component, B_e, and a rigid component, B_r. The C matrix is partitioned into an elastic component, C_e, and a rigid component, C_r.

The idea is that we use *steady* AIC, AIC_s, which is developed using vortex lattice methods [3.28]. Then, we introduce wagner's lift growth function to lag the lift generated by twist in the wings. Details of this are discussed further in section 7.10.

3.10 Aileron Control Reversal

Apart from sensor placement, an unstable zero can present itself when dynamic pressure becomes too high or the wing is too weak in twist. When an aileron moves down to increase lift, the wing

can twist so greatly that the angle of attack is reduced such that a downward force is produced and roll reverses. This creates a NMP zero (which is SISO). If this occurs at a given flight condition, ailerons are typically locked and asymmetric spoilers are used to turn the aircraft [3.4].

Chapter 4 – Decentralized Control

4.1 Choosing Input-Output Pairings in Decentralized Control

Many Multi-Input Multi-Output (MIMO) plants can have their inputs and outputs arranged such that they are diagonally dominant.

$$|G_{ii}(j\omega)| > \sum_{k=1 \quad k\neq i}^{m} |G_{ik}(j\omega)|, \quad \forall i, \quad \forall \omega$$

$$y_i \quad \text{is} \quad \text{paired} \quad \text{with} \quad u_i, \quad y_i \rightarrow u_i$$

If this is indeed the case, then one can sometimes design separate controllers, $C_{ii}(s)$, for each $G_{ii}(s)$ as in Fig. 4.1.

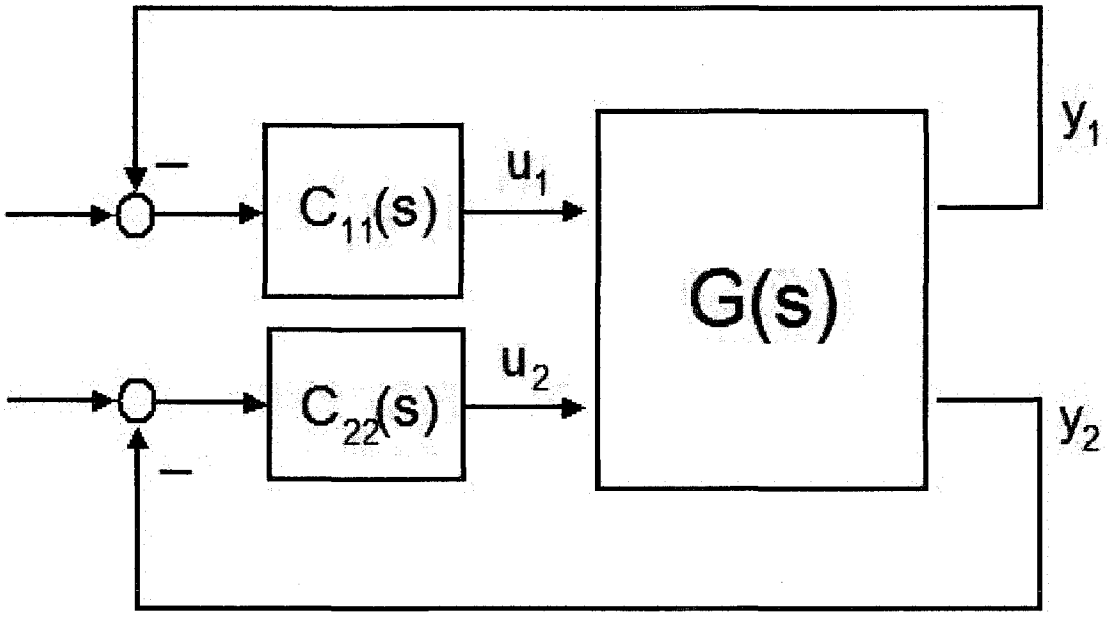


Fig. 4.1. Decentralized Control for a Two-Input, Two-Output (TITO) system.

Sometimes, one can simply use intuition to make input output pairings because they are obvious. For instance, when driving a car, one pairs steering angle with turning rate and the brakes or gas pedal would be paired with forward velocity. You would not expect to slow down or speed up by turning the steering wheel, nor would you expect to turn by pressing down on the gas pedal. Many things which are made for humans to operate are often diagonally dominant and are good candidates for decentralized control. A car whose inputs and outputs are steering angle and brakes or gas pedal is an example of a diagonally dominant and well conditioned plant.

On the other hand, poorly conditioned plants that exhibit strong interactions between inputs and outputs can exist, although they are less common. The two-tank apparatus in section 2.17 is just such an example (note that it is purely hypothetical). Suppose one began to put fluid into the first tank. At first, almost no fluid would end up in the second tank. However, after some time half the fluid would be in the first tank and the other half would be in the second tank. The same can be said about the second tank. Therefore, we can say that at low frequency, there is a very strong interaction between the inputs and outputs. At low frequency, the plant is poorly conditioned and also does not exhibit diagonal dominance.

Various metrics can be used to quantify interactions between control loops and to help make input output pairings. We will discuss the relative gain array (RGA) which can be used to gauge interaction between inputs and outputs and can also suggest input output pairings. Additionally, we will discuss the method of coupling numerators to help plan out successive loop closures in decentralized control. A theoretical discussion of decentralized control can be found in Ref. [4.1].

4.2 The Relative Gain Array (RGA)

The RGA has been in use for quite some time in control [4.2] and for plants with fewer than 4 inputs and outputs has predictable characteristics [2.10]. The relative gain array is defined by element by element multiplication of the plant and its inverse.

$$RGA(G(j\omega)) = G^{-T}(j\omega) \otimes G(j\omega)$$
$$\otimes \quad \text{is} \quad \text{element} - \text{by} - \text{element} \quad \text{multiplication}$$

It is typical to look at the RGA at steady state, RGA(0). Interpreting the RGA at frequencies with oscillatory poles or zeros can give strange results. We list the following things that can be inferred from the RGA.

1. *If some of the entries of the RGA are large, above 10 or 100, then there is a large interaction between inputs and outputs.* The plant is poorly conditioned.
2. For plants with fewer than 4 inputs and outputs, a norm of the RGA, is directly related to a normalized condition number [4.3, 4.4, 2.10]. For any plant with more than 4 inputs there is an inequality relationship with the condition number [4.5, 2.10].
3. If $RGA_{ik} \approx 1$, then one can pair y_i with u_k, $y_i \rightarrow u_k$.

4. If $RGA_{ik} < 0$, avoid pairing y_i with u_k, $y_i \rightarrow u_k$.

5. For the two-input two-output case, if the RGA is identity, $RGA = I$, then the plant is triangular or diagonal.

6. The elements in each row of the RGA sum to 1. $\sum_{k=1}^{m} RGA(G)_{ik} = 1$

The RGA is not a strict guide to input and output pairing, especially since it is frequency dependent. Pairings may be poor at one frequency and very good at other frequencies.

Example 4.1

Consider the following arbitrary plant with nominal, $G(s)$, and perturbed configurations, $G^{p1}(s)$ and $G^{p2}(s)$.

$$G(s) = \begin{bmatrix} \dfrac{(s+8)}{4(s+1)(s+2)} & 0 \\ 0 & \dfrac{2(s+10)}{10(s+1)(s+2)} \end{bmatrix}$$

$$G^{p1}(s) = \begin{bmatrix} \dfrac{(s+8)}{4(s+1)(s+2)} & \dfrac{(s+8)}{10(s+1)(s+2)} \\ \dfrac{10(s+10)}{(s+1)(s+2)} & \dfrac{2(s+10)}{10(s+1)(s+2)} \end{bmatrix}$$

$$G^{p2}(s) = \begin{bmatrix} \dfrac{(s+8)}{4(s+1)(s+2)} & \dfrac{-(s+8)}{10(s+1)(s+2)} \\ \dfrac{-10(s+10)}{(s+1)(s+2)} & \dfrac{2(s+10)}{10(s+1)(s+2)} \end{bmatrix}$$

Notice that for this plant, the RGA is wildly variable.

$$RGA(G(s)) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$RGA(G^{p1}(s)) \approx \begin{bmatrix} -0.0526 & 1.0526 \\ 1.0526 & -0.0526 \end{bmatrix}$$

$$RGA(G^{p2}(s)) \approx \begin{bmatrix} -0.0526 & 1.0526 \\ 1.0526 & -0.0526 \end{bmatrix}$$

These several plant configurations would be very tough to control not only because of the reversal in the RGA but also because of the sign reversal in the non-diagonal terms. Note that the decentralized controller, which is designed for the nominal configuration $G(s)$ and is shown below, destabilizes both perturbed configurations. *However, one should not use the RGA as a definite indicator of stability or instability.*

$$C(s)=\begin{bmatrix} \dfrac{4(s+1)(s+2)}{s(s+8)} & 0 \\ 0 & \dfrac{25(s+1)(s+2)}{s(s+10)} \end{bmatrix}$$

Example 4.2

Consider the two tank apparatus in example 2.10. At low frequency, the following approximation is largely true. This is true because half the fluid ends up in either tank after a long time.

$$G(j\omega)\approx\begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}\frac{1}{j\omega},\quad \omega \text{ small } \Rightarrow \quad RGA(G(j\omega))=\infty$$

Now consider the high frequency behavior when the fluid does not have enough time to "leak" into the other tank.

$$G(j\omega)\approx\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\frac{1}{j\omega},\quad \omega \text{ large } \Rightarrow \quad RGA(G(j\omega))=\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Figure 4.2 shows the RGA plot versus frequency. We consider the case where the fluid is corn syrup, $\rho=1380\,{}^{kg}/_{m^3}$ $\mu=1.3806\,{}^{N\cdot s}/_{m^2}$. The diameter of tank 1 and tank 2 are both 0.1 meters; the length of the pipe is 0.5 meters and has a diameter of 0.025 meters; the gravity is $g=9.81\,{}^{m}/_{s^2}$.

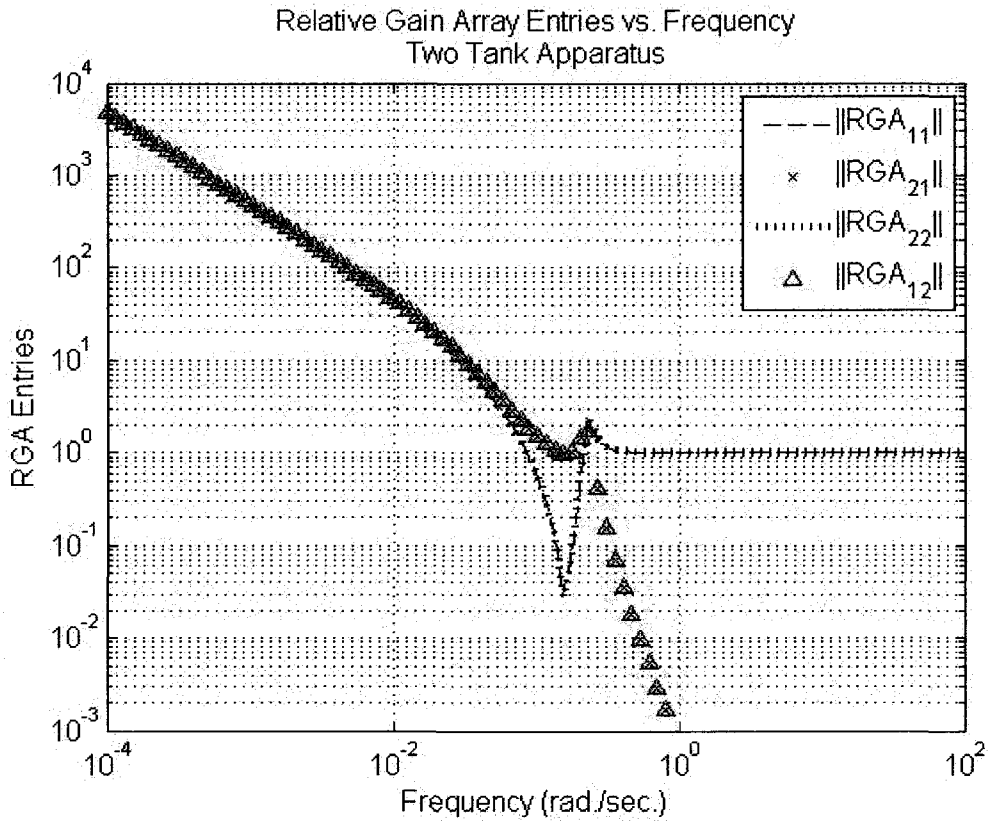


Fig. 4.2. Notice that the RGA values become very large at low frequency and would eventually tend towards infinity in the true steady state. There is a transition region where the RGA values shift towards being remarkably well conditioned at high frequency.

Example 4.3

Consider the lateral dynamics of the Navion at Mach = 0.158 and Sea-Level (SL). The inputs are aileron, δ_a, and rudder, δ_r, respectively. The outputs are bank angle, ϕ, and yaw rate, r. Notice that at low frequency, the plant is very poorly conditioned (i.e. the rows of G(s) are very close to being linearly dependent).

$$G(j\omega) \approx \begin{bmatrix} 2.6668 & -1.3298 \\ 0.46785 & -0.23329 \end{bmatrix} \frac{1}{j\omega + 0.008243} \quad \omega < 0.05 \quad (rad./sec.)$$

This is because the aircraft is made to enter into a steady turn open loop, where

$$r \approx \frac{g}{U_o \cos(\theta)} \tan(\phi) \approx \frac{g}{U_o} \phi .$$

At high frequency, most aircraft will approximately behave like the two tank apparatus with pure integration as the aileron and rudder simply torque the aircraft about its body-fixed x-axis and z-axis, respectively. The steady turn does not have time to develop.

$$G(j\omega) \approx \begin{bmatrix} \dfrac{-L'_{\delta a}}{\omega^2} & \dfrac{-L'_{\delta r}}{\omega^2} \\ \dfrac{N'_{\delta a}}{j\omega} & \dfrac{N'_{\delta r}}{j\omega} \end{bmatrix} \approx \begin{bmatrix} \dfrac{-28.98}{\omega^2} & \dfrac{-2.19}{\omega^2} \\ \dfrac{-0.22}{j\omega} & \dfrac{-4.60}{j\omega} \end{bmatrix} \quad \omega \text{ large}$$

G(s), lateral Navion at Mach 0.158, SL

$$A = \begin{bmatrix} -0.2543 & 0 & -1.0000 & 0.1830 \\ -15.9820 & -8.4020 & 2.1930 & 0 \\ 4.4950 & -0.3498 & -0.7605 & 0 \\ 0 & 1.0000 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0.0708 \\ 28.9840 & 2.1930 \\ -0.2218 & -4.5970 \\ 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

State Space. Input vector is $u = [\delta_a \quad \delta_r]^T$ units are [rad. rad.]. The output vector is $y = [\phi \quad r]^T$, units are [rad. rad.] respectively.

The two tank apparatus and the navion aircraft are two completely different things, yet they have similar characteristics. There is poor conditioning at low frequency and very good conditioning at high frequency as indicated by the RGA.

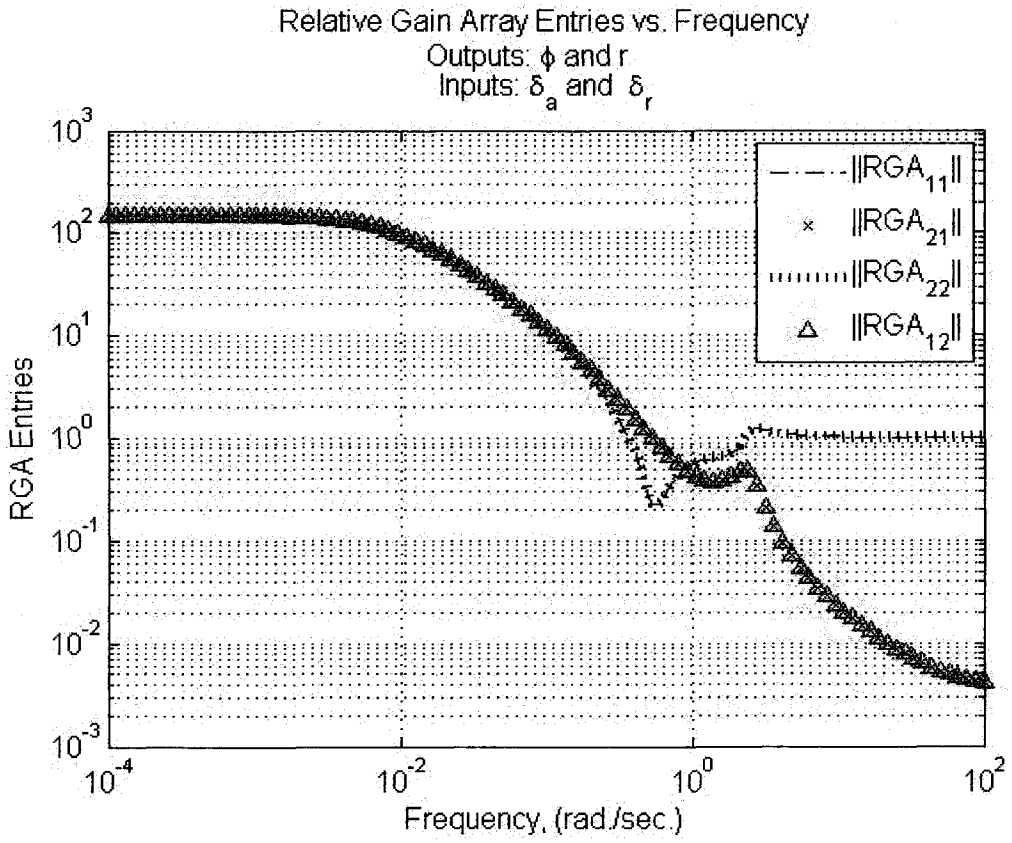


Fig. 4.3. The relative gain array is very high at low frequency, which means that the plant is poorly conditioned at low frequency. There will inevitably be a strong level of interaction between outputs at low frequency. This is due to the aircraft's tendency to enter into a steady turn. At higher frequencies, above 1 rad./sec., the plant becomes strongly diagonal. It would be a really bad idea to invert this plant across all frequencies.

Example 4.4

With this poor conditioning in mind, we consider different outputs. The outputs are bank angle, ϕ, and sideslip angle, β, respectively. The inputs are aileron, δ_a, and rudder, δ_r, respectively. The data comes from the lateral dynamics of the Navion at Mach = 0.158 and Sea-Level (SL). Notice that the condition is much better at low frequency because the maximum RGA element is less than 5, which is much better.

G(s), lateral Navion at Mach 0.158, SL

$$A=\begin{bmatrix} -0.2543 & 0 & -1.0000 & 0.1830 \\ -15.9820 & -8.4020 & 2.1930 & 0 \\ 4.4950 & -0.3498 & -0.7605 & 0 \\ 0 & 1.0000 & 0 & 0 \end{bmatrix}$$

$$B=\begin{bmatrix} 0 & 0.0708 \\ 28.9840 & 2.1930 \\ -0.2218 & -4.5970 \\ 0 & 0 \end{bmatrix}$$

$$C=\begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

State Space. Input vector is $u=[\delta_a \quad \delta_r]^T$ units are [rad. rad.]. Outputs are $y=[\phi \quad \beta]^T$, units are [rad. rad.] respectively.

The RGA for the navion at d.c. is shown below. The outputs are bank angle, ϕ, and sideslip angle, β, respectively

$$RGA(G(0))=\begin{bmatrix} -3.607 & 4.607 \\ 4.607 & -3.607 \end{bmatrix}$$

However, at high frequency the dynamics of the plant are approximately triangular and the relative gain array is identity.

$$G(j\omega)\approx\begin{bmatrix} \frac{-L'_{\delta a}}{\omega^2} & \frac{-L'_{\delta r}}{\omega^2} \\ 0 & \frac{Y^*_{dr}}{j\omega} \end{bmatrix}\approx\begin{bmatrix} \frac{-L'_{\delta a}}{\omega^2} & 0 \\ 0 & \frac{Y^*_{dr}}{j\omega} \end{bmatrix} \quad \omega \text{ large}, \; RGA(\|G(j\omega)\|)\approx\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

This is significant because there is a reversal in the sign of the RGA as one travels from low frequency to high frequency. It might be that a diagonal pairing could lead to instability at low frequency. However, if an off diagonal pairing were pursued, then high frequency instability may result because the pairing would be incorrect.

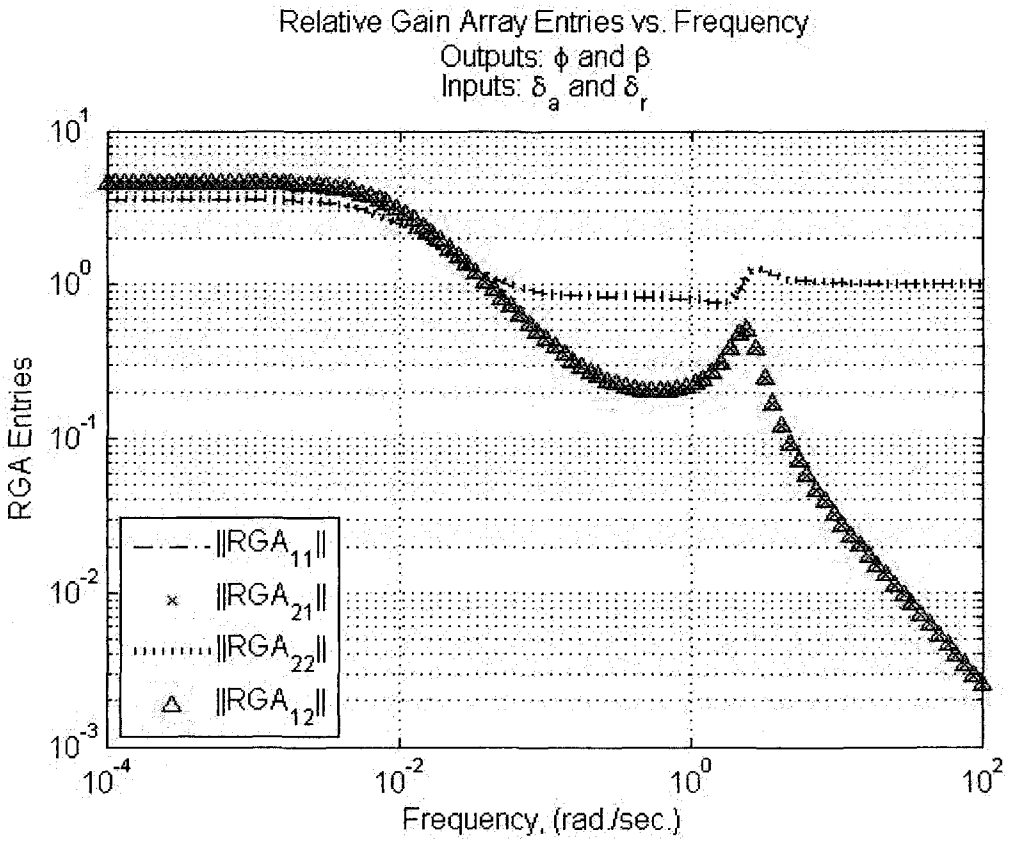


Fig. 4.4. The relative gain array between the outputs of ϕ and β. The low frequency behavior is much better conditioned and the interaction between outputs will be smaller. Decoupling these outputs is less unreasonable than decoupling ϕ and r.

4.3 Coupling Numerators for TITO Plants

Coupling numerators have been popular in aircraft control [4.6, 3.2]. Coupling numerators allow one to plan decentralized loop closures given a pairing scheme. Coupling numerators can be very successful for minimum phase plants. Non-minimum phase plants can be more difficult to extend coupling numerator planning.

Given a TITO system below

$$G(s)=\begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix}=\begin{bmatrix} N_{u1}^{y1}(s) & N_{u2}^{y1}(s) \\ N_{u1}^{y2}(s) & N_{u2}^{y2}(s) \end{bmatrix}\frac{1}{\Delta(s)} \tag{4.1}$$

Suppose we design want to pair $y_1 \rightarrow u_1$, we would design $C_{11}(s)$ to achieve a high bandwidth. We want to see what would happen to $G_{22}(s)\big|_{y1\rightarrow u1}$ with $C_{11}(s)$.

$$\begin{aligned} G_{22}(s)\big|_{y1\rightarrow u1} &= \frac{(G_{11}(s)G_{22}(s)-G_{21}(s)G_{12}(s))C_{11}(s)+G_{22}(s)}{G_{11}(s)C_{11}(s)+1} \\ G_{22}(s)\big|_{y1\rightarrow u1} &= \frac{\boldsymbol{det}(G(s))C_{11}(s)+G_{22}(s)}{G_{11}(s)C_{11}(s)+1} \end{aligned} \tag{4.2}$$

Next we assume that $C_{11}(s)>>1$ and approximate as follows.

$$G_{22}(s)\big|_{y1\rightarrow u1} \approx \frac{\boldsymbol{det}(G(s))}{G_{11}(s)} \tag{4.3}$$

By symmetry, we can assume that we would close the $y_2 \rightarrow u_2$ loop and get the following result with $C_{22}(s)>>1$.

$$G_{11}(s)\big|_{y2\rightarrow u2} \approx \frac{\boldsymbol{det}(G(s))}{G_{22}(s)} \tag{4.4}$$

With $y_1 \rightarrow u_2$ and $C_{12}(s)>>1$ the following is the case.

$$G_{21}(s)\big|_{y1\rightarrow u2} \approx \frac{\boldsymbol{det}(G(s))}{G_{12}(s)} \tag{4.5}$$

Also, with $y_2 \rightarrow u_1$ and $C_{21}(s)>>1$, the following is the case.

$$G_{12}(s)\big|_{y2\to u1} \approx \frac{\boldsymbol{det}(G(s))}{G_{21}(s)} \tag{4.6}$$

Notice that the numerator of these transfer functions is strongly effected by $\boldsymbol{det}(G(s))$, whose roots are the *transmission zeros* of the plant, $G(s)$.

Example 4.5

Consider the two tank apparatus in example 2.10, at low frequency the following approximation is largely true. We consider the case where the fluid is corn syrup, $\rho = 1380\,{}^{kg}/_{m^3}$ $\mu = 1.3806\,{}^{N\cdot s}/_{m^2}$. The diameter of tank 1 and tank 2 are both 0.1 meters; the length of the pipe is 0.5 meters and has a diameter of 0.025 meters; the gravity is $g = 9.81\,{}^{m}/_{s^2}$.

Outputs are volume of tank 1 and tank 2, respectively. Inputs are flow to tank 1 and tank 2, respectively. The coupling numerators are summarized below.

$$G(s) = \begin{bmatrix} \dfrac{(s^2+0.02514s+0.02452)}{(s^2+0.02514s+0.04905)} & \dfrac{0.024525}{(s^2+0.02514s+0.04905)} \\ \dfrac{0.024525}{(s^2+0.02514s+0.04905)} & \dfrac{(s^2+0.02514s+0.02452)}{(s^2+0.02514s+0.04905)} \end{bmatrix} \frac{1}{s}$$

$$G_{22}(s)\big|_{y1\to u1} = \frac{(s+0.02514)}{(s^2+0.02514s+0.02452)}$$

$$G_{11}(s)\big|_{y2\to u2} = \frac{(s+0.02514)}{(s^2+0.02514s+0.02452)}$$

$$G_{21}(s)\big|_{y1\to u2} = \frac{\infty(s+0.02514)}{(s^2-3.751s+\infty)} \approx (s+0.02514)$$

$$G_{12}(s)\big|_{y2\to u1} = \frac{\infty(s+0.02514)}{(s^2-3.751s+\infty)} \approx (s+0.02514)$$

It seems natural to pair the volume of tank 1 with the flow to tank 1 and the volume of tank 2 with the flow to tank 2. The coupling numerators show that this is a reasonable choice because both

$G_{22}(s)\big|_{y1\to u1}$ and $G_{11}(s)\big|_{y2\to u2}$ are stable and easily controllable. However, $G_{12}(s)\big|_{y2\to u1}$ both $G_{21}(s)\big|_{y1\to u2}$ seem strange.

We decide to pair $y_1 \to u_1$ and design $C_{11}(s)$ with a high bandwidth of 10 rad./sec. We show the loop-shape for the first loop closure of $L_{11}(s) = G_{11}(s)C_{11}(s)$. Here, we use $C_{11}(s) = 10$.

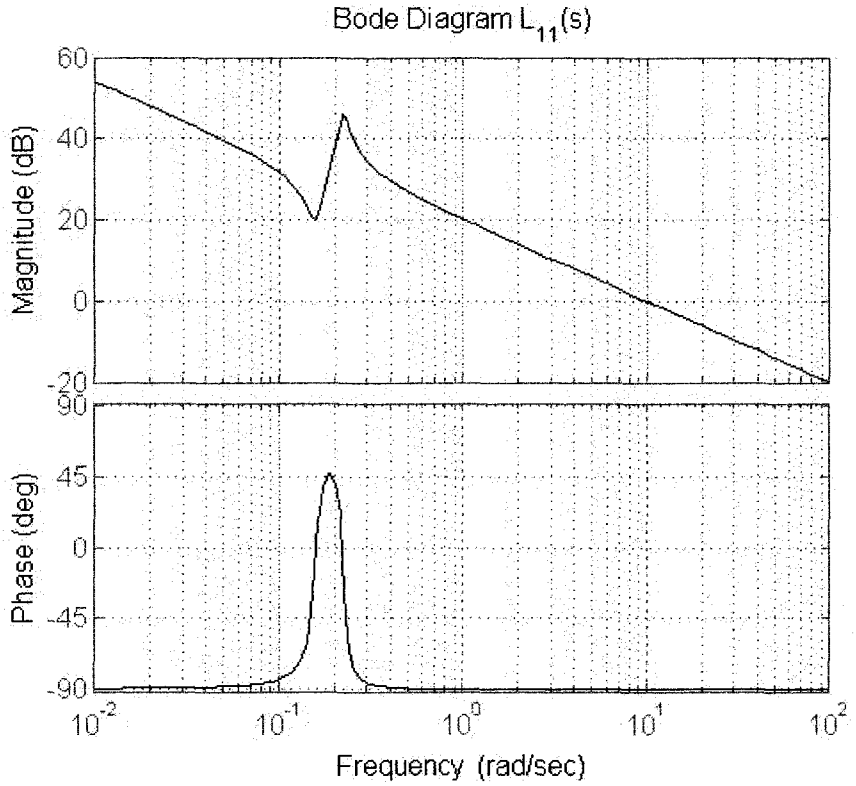


Fig. 4.5. Notice that the loop shape has a phase margin of 90 degrees and a slope of -20 dB/dec. at cross-over, where $L_{11}(s)$ is near 0 dB.

Now recall from example 4.1, we know that the relative gain array is very large at low frequency. Therefore, the interaction between loops will be very strong at low frequency. We decide to make a performance decision and avoid integral tracking of the second tank.

Therfore, we design a washout filter such that there is damping of the oscillatory mode and the dc gain of the second loopshape is 0, $L_{22}(0) = 0$. This loop shape is shown in Fig. 4.6. We show the loop-shape for the second loop closure of $L_{22}(s) = G_{22}(s)\big|_{y1\to u1} C_{22}(s)$, we have used the washout design of $C_{22}(s) = \dfrac{0.5s}{(s+0.025)}$.

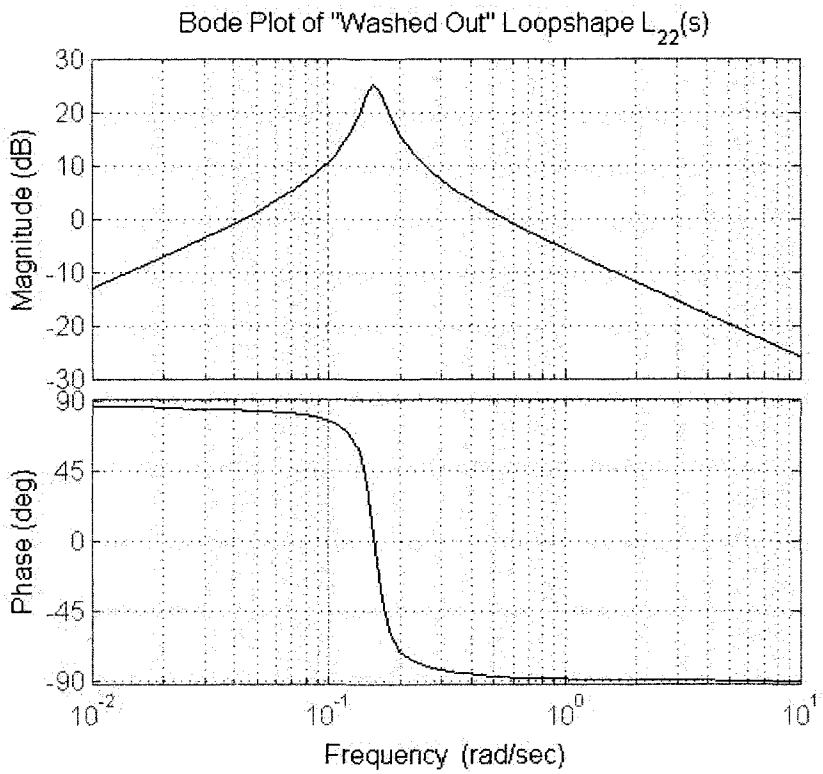


Fig. 4.6. The washout filter damps out a pair of oscillatory poles and also respects the poorly conditioned low frequency behavior indicated by the RGA.

The resulting complimentary sensitivity with this design has a dc gain of $T(0)=\begin{bmatrix}1 & 0\\ 1 & 0\end{bmatrix}$. This means that if one were to command the volume in tank 1 to reach a certain point, the second tank would reach the same volume. However, if one were to command the volume to reach a certain point in the second tank, eventually everything would settle to zero.

Example 4.5 – MATLAB code for Decentralized Design of Two-Tank Apparatus

```
g=9.81;

%fluid params corn syrup
rho=1380;
mu=1.3806;

%pipe params for laminar flow
Dp=.025;
Lp=1;
Ap=pi/4*Dp^2;
Ip=rho*Lp/Ap;
Rp=1/2*64*mu*Lp/(Dp^2);  %laminar assumption

%tank 1 params
Dt1=.5;
At1=pi/4*Dt1^2;
C1=At1/(rho*g);

%tank 2 params
Dt2=.5;
At2=pi/4*Dt2^2;
C2=At2/(rho*g);

a31=1/(Ip*C1);
a32=-1/(Ip*C2);
a33=-Rp/Ip;
b11=1;
b22=1;

C=[eye(2) zeros(2,1)];
A=[0 0 -1;0 0 1;a31 a32 a33];
B=[b11 0;0 b22;0 0];
tankss=ss(A,B,C,zeros(2));

[y2u2_y1u1,y1u1_y2u2,y2u1_y1u2,y1u2_y2u1]...
    = coupling_num_2(A,B,C,zeros(2));

C11=10; %cross over at 10 rad./sec.
G2ndloop=feedback(tankss*[C11 0;0 1],[1 0;0 0]); %close first loop

C22=zpk([0],[-0.025],0.5); %washout

T=feedback(tankss*[C11 0;0 C22],eye(2)); %close second loop
T0=dcgain(T);

plotRGA2by2(A,B,C,zeros(2));
```

Example 4.2-4.5 – MATLAB Subroutine for Plotting RGA vs. Frequency

```
function plotRGA2by2(A,B,C,D)

omega=logspace(-4,2,101);

for i=1:101
    [MAG,PHASE]=bode(ss(A,B,C,zeros(2)),omega(i));
    G=MAG.*(cos(PHASE/180*pi)+sin(PHASE/180*pi).*j);
    R=inv(G').*G;
    absR=abs(R);
    R11(i)=absR(1,1);
    R12(i)=absR(1,2);
    R21(i)=absR(2,1);
    R22(i)=absR(2,2);
    absIO=abs(IO);
    IO11(i)=absR(1,1);
    IO21(i)=absR(2,1);
end

figure(1);
loglog(omega,R11,'k--',omega,R21,'rx',omega,R22,'k:',omega,R12,'r^')
legend('||RGA_1_1||','||RGA_2_1||','||RGA_2_2||',...
    '||RGA_1_2||','FontSize',12);
grid on
title('Relative Gain Array Entries vs. Frequency','FontSize',12)
ylabel('RGA Entries','FontSize',12);
xlabel('Frequency (rad./sec.)','FontSize',12);
```

Example 4.5 – MATLAB Subroutine for Computing Coupling Numerators

```
function [y2u2_y1u1,y1u1_y2u2,y2u1_y1u2,y1u2_y2u1] = ...
    coupling_num_2(A,B,C,D)

[numq,denq]=ss2tf(A,B,C,D,1);
[numV,denV]=ss2tf(A,B,C,D,2);
[z,p,k] = tf2zpk(numq(1,:),denq);
G_11 = zpk(z,p,k);
[z,p,k] = tf2zpk(numq(2,:),denq);
G_21 = zpk(z,p,k);
[z,p,k]= tf2zpk(numV(1,:),denV);
G_12 = zpk(z,p,k);
[z,p,k]= tf2zpk(numV(2,:),denV);
G_22 = zpk(z,p,k);

%look at y2u2 after pairing y1 with u1 tightly constrained
detG = G_11*G_22-G_12*G_21;
y2u2_y1u1 = detG*inv(G_11);
%look at y1u1 after pairing y2 with u2 tightly constrained
y1u1_y2u2 = detG*inv(G_22);
%look at y2u1 after pairing y1 with u2 tightly constrained
y2u1_y1u2 = detG*inv(G_12);
%look at y1u2 after pairing y2 with u1 tightly constrained
y1u2_y2u1 = detG*inv(G_21);
```

Example 4.6

Reconsider the lateral dynamics of the Navion at Mach = 0.158 and Sea-Level (SL). The inputs are aileron, δ_a, and rudder, δ_r, respectively. The outputs are bank angle, ϕ, and yaw rate, r. Recall that the low frequency dynamics had poorly conditioned dynamics with RGA values approaching 100. We check the coupling numerators for a loop closure plants.

$$G(s)=\begin{bmatrix} N_{\delta a}^{\phi}(s) & N_{\delta r}^{\phi}(s) \\ N_{\delta a}^{r}(s) & N_{\delta r}^{r}(s) \end{bmatrix}\frac{1}{\Delta_{lat}(s)}$$

$$N_{\delta a}^{\phi}(s)=28.984\left(s^2+0.998s+4.562\right)$$

$$N_{\delta r}^{\phi}(s)=2.193(s-7.902)(s+3.804)$$

$$N_{\delta a}^{r}(s)=-0.2218(s+54.08)(s+1.543)(s-1.253)$$

$$N_{\delta r}^{r}(s)=-4.597(s+8.613)\left(s^2+0.1413s+0.2939\right)$$

$$\Delta_{lat}(s)=(s+8.435)(s+0.008758)\left(s^2+0.9735s+5.688\right)$$

$$G_{r\delta r}(s)\big|_{\phi\to\delta a}=\frac{-4.58(s+0.1867)}{\left(s^2+0.998s+4.562\right)}$$

$$G_{\phi\delta a}(s)\big|_{r\to\delta r}=\frac{28.8782(s+0.1867)}{(s+8.613)\left(s^2+0.1413s+0.2939\right)}$$

$$G_{r\delta a}(s)\big|_{\phi\to\delta r}=\frac{-60(s+0.1867)}{(s+3.804)(s-7.902)}$$

$$G_{\phi\delta r}(s)\big|_{r\to\delta a}=\frac{598.5259(s+0.1867)}{(s-1.253)(s+1.543)(s+54.08)}$$

The coupling numerators show that pairing is a reasonable choice because both $G_{r\delta r}(s)\big|_{\phi\to\delta a}$ and $G_{\phi\delta a}(s)\big|_{r\to\delta r}$ are stable. However, $G_{\phi\delta r}(s)\big|_{r\to\delta a}$ both $G_{r\delta a}(s)\big|_{\phi\to\delta r}$ are unstable, therefore one would not want to yaw rate with the aileron, nor would one want to pair bank angle with the rudder.

Recall that the plant is poorly conditioned at low frequency and there will be a strong interaction between loops. We decide that keeping the wings level is more important than maintaining a steady yaw rate. Therefore, we will not be pursuing integral tracking in the $r \to \delta r$ loop.

First we pursue a loop-shape which will level the wings, which is perceived to be more important. Therefore, we pair $\phi \rightarrow \delta_a$ and try to make $L_\phi(s) = G_{\phi\delta a}(s)C_{\delta a\phi}(s) \approx \omega_{BW\phi}/s$ across a broad frequency range, especially near $\omega_{BW\phi} \approx 5$ rad./sec. We design

$C_{\delta a\phi}(s) = \dfrac{7(s+0.009)(s+8.4)}{s(s+40)}$, which turns out to be a PID with a filter. The loopshape is shown below in Fig. 4.7

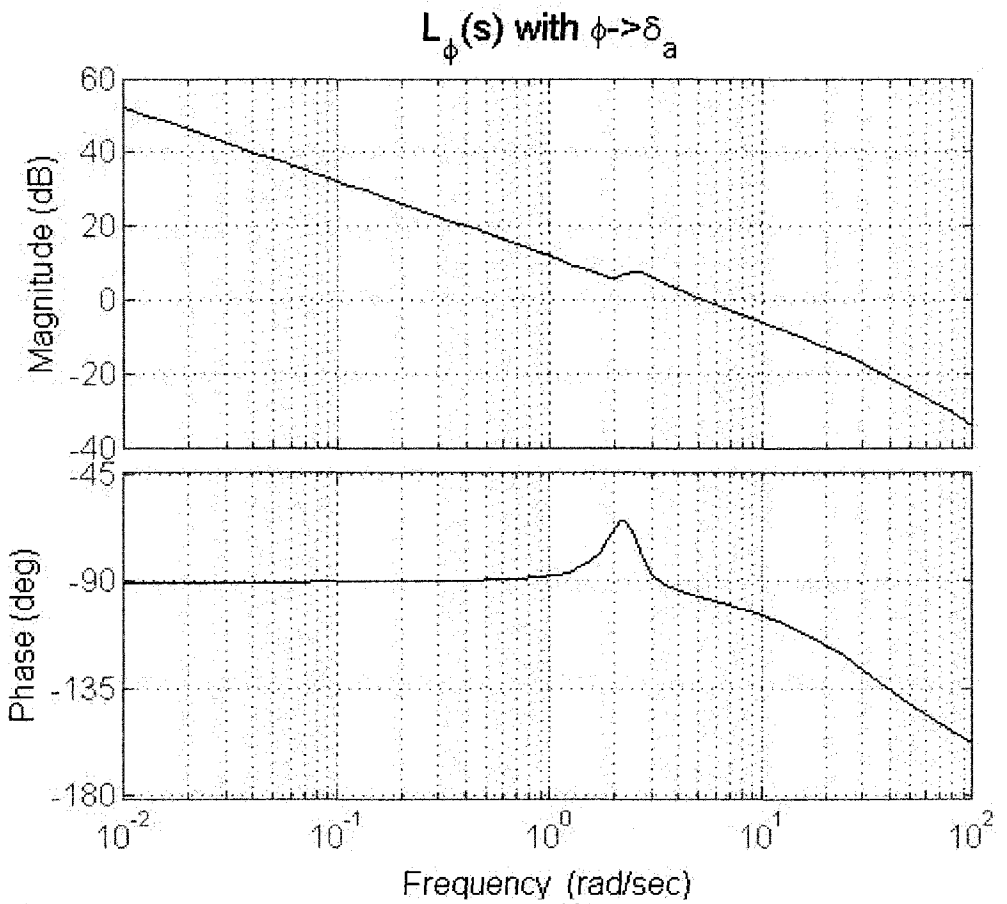


Fig. 4.7. Notice that the loop shape has a phase margin of 90 degrees and a slope of -20 dB/dec. at cross-over, where $L_\phi(s)$ is near 0 dB.

Now the next thing to do is to design the second loop closure,

$G_{r\delta r}(s)\big|_{\phi\rightarrow\delta a} = \dfrac{-(s+0.1867)}{(s^2+0.998s+4.562)}$. We consider the following washout design,

$C_{\delta r}(s) = \dfrac{-k_{wo}s}{(s+0.2)}$ $k_{wo} > 0$. Upon inspecting the root locus of Fig. 4.8, we decide on a modest gain of $k_{wo} = 0.1$. The loop shape with the washout is shown in Fig. 4.8.

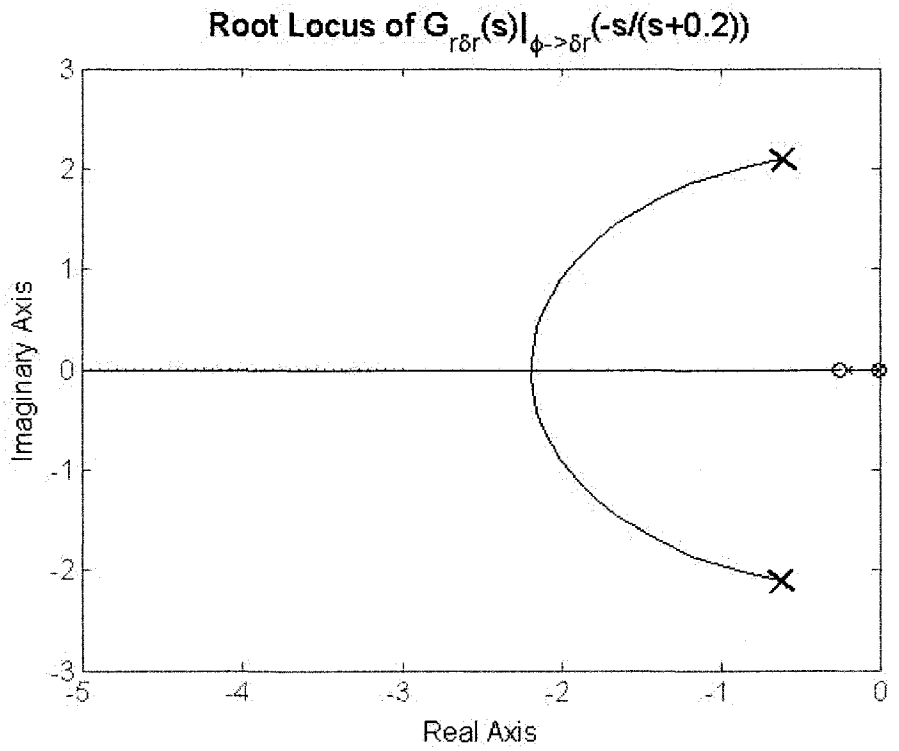


Fig. 4.8 One of the principle purposes of a washout filter is to add damping to the dutch roll mode, which are the oscillatory poles shown in bold.

The purpose of the washout filter is to only interfere with dynamics of the dutch roll which are the pair oscillatory poles shown in Fig. 4.8. If integral tracking were pursued with the second loop closure, there would be a significant effect on the first loop closure, and low frequency tracking of bank angle would be compromised. This is due to the high RGA at low frequency as seen in Fig. 4.3.

We now inspect the dc gain of our complimentary sensitivity, or closed loop transfer function.

We notice that $T(0) = \begin{bmatrix} 1 & 0 \\ 0.1754 & 0 \end{bmatrix}$. Interestingly enough,

$g/U_o = (32.2 ft./sec/sec)/(176 ft./sec) \approx 0.183$, which is very close to $T(0)_{21} = 0.1754$.

Thus, a step input into a bank angle command will give a nearly steady turn.

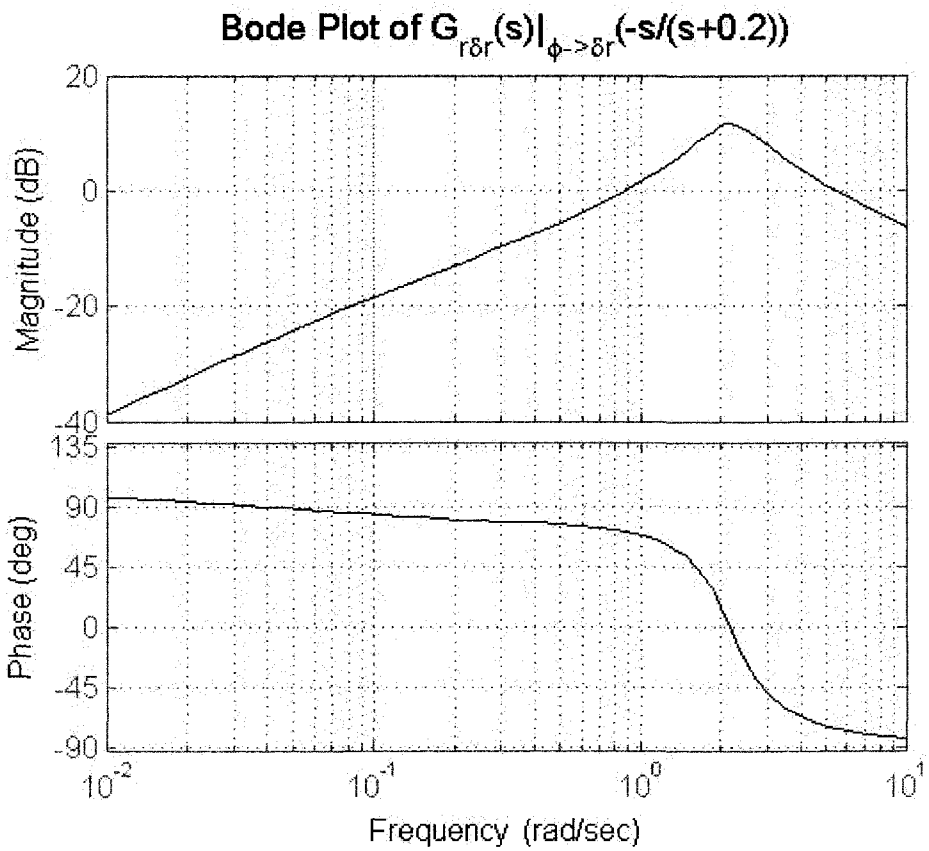


Fig. 4.9. The other purpose of the washout filter is not to exert too much influence at low frequency. If we used an integator in this control loop, then the $L_{\phi}(s)$ loop would be adversely affected.

The purpose of the washout filter is to only interfere with dynamics of the dutch roll which are the pair oscillatory poles shown in Fig. 4.8. If integral tracking were pursued with the second loop closure, there would be poorer low frequency performance of the bank angle channel. This is because of the strong interaction between loops at low frequency, as indicated by the RGA in Fig. 4.3.

Example 4.6 – MATLAB code for Navion Decentralized Design

```
clc
clear all

%lateral Navion at SL mach 0.158
%Calculate ff precomp with phi and r as outputs

Uo=176;
bank=0/180*pi;
cbank=cos(bank);
Yv=-.2543;
Lbeta=-15.982;
Nbeta=4.495;
Lp=-8.402;
Np=-0.3498;
Lr=2.193;
Nr=-0.7605;

Ysda=0;
Lda=28.984;
Nda=-0.2218;
Ysdr=.0708;
Ldr=2.193;
Ndr=-4.597;

Alat=[Yv 0 -1 32.2/Uo*cbank;Lbeta Lp Lr 0;
    Nbeta Np Nr 0;0 1 0 0];
Blat=[Ysda Ysdr;Lda Ldr;Nda Ndr;0 0];
Clat=[0 0 0 1;0 0 1 0;Uo*Yv 0 -Uo*0 32.2*cbank];
Dlat=[zeros(2,2);Ysda*Uo    Ysdr*Uo];

syslat=ss(Alat,Blat,[Clat(1,:);Clat(2,:)],zeros(2));

[y2u2_y1u1,y1u1_y2u2,y2u1_y1u2,y1u2_y2u1]=...
    coupling_num_2(Alat,Blat,Clat,Dlat);

%sisotool(syslat(1,1))
C11=zpk([-0.009 -8.4],[0 -40],7);

syscl1=feedback(syslat*[C11 0;0 1],[1 0;0 0]); %close first loop

%sisotool(syscl1(2,2))
C22=zpk([0],[-0.2],-.1);

T=feedback(syslat*[C11 0;0 C22],[1 0;0 1]); %close both loops
```

Example 4.7

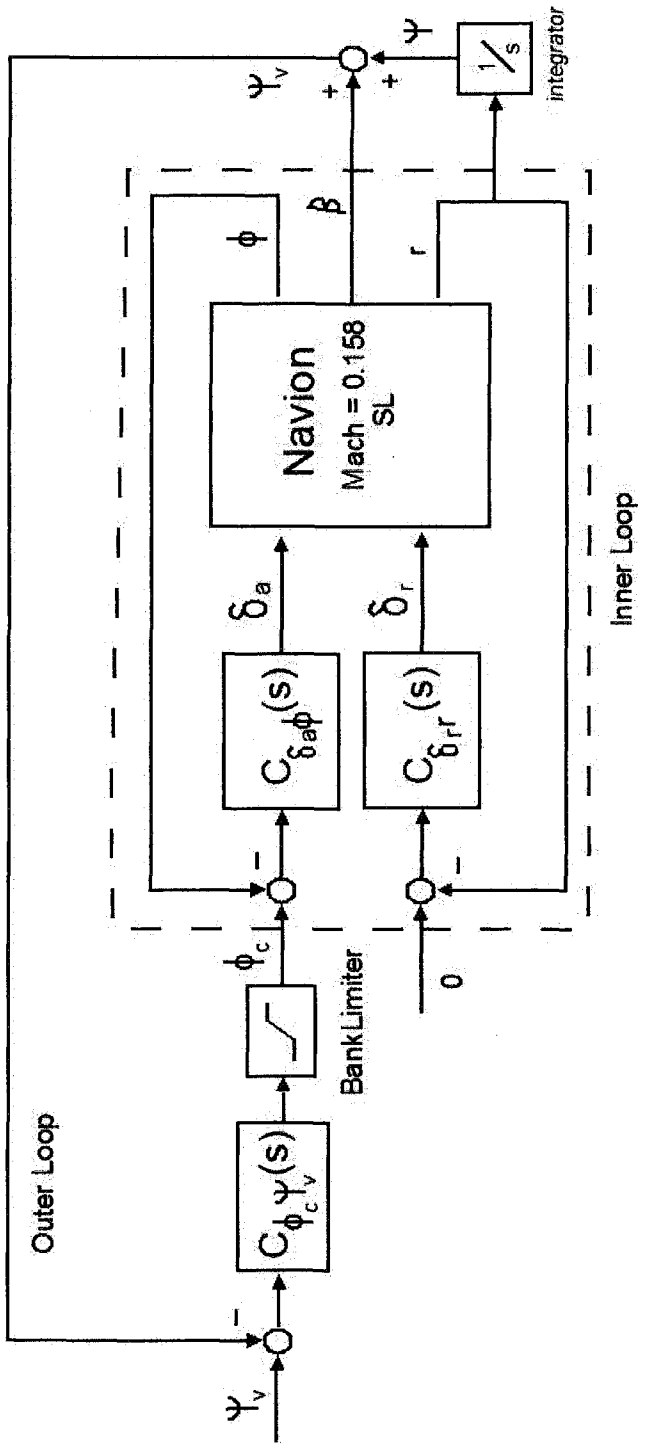


Fig. 4.10. We have an outer loop and an inner loop control system. The outer loop tells the inner loop what to do. We have included a bank command limiter which makes sure that the commanded bank angle does not exceed unreasonable limits. Usually it is set to 15 to 30 degrees.

In example 4.6, we designed an inner loop autopilot that consisted of a PID wing lever and a washout filter for yaw rate. Now we will complete the design by adding a heading angle controller which gives bank angle commands. Figure 4.10 shows the architecture of the autopilot.

We saw that the complimentary sensitivity indicated that an approximately steady turn would be achieved if we gave a step input to bank command. We can use this behavior to design a heading angle controller. Our heading angle will be called $\psi_v \approx \psi + \beta$. A truly steady turn approximation would be achieved with $\dot{\psi} = \frac{g\tan(\phi)}{U_o \cos(\theta)} \approx \frac{g}{U_o}\phi$, where ψ would be the Euler angle, ψ. Some may choose to approximate this as $\psi_v \approx \psi$ and use this as the controlled output. However, this interpretation does not take into account sideslip. We include sideslip angle, apparently it is a better approximation for outer loop design.

The bank limiter is a very important design feature in the control system because heading errors could be in the range of 0 to 360 degrees, but you would not want to command a bank angle 360 degrees. Therefore, the bank angle limiter is there to prevent that and usual limits are between 15 and 30 degrees of bank command.

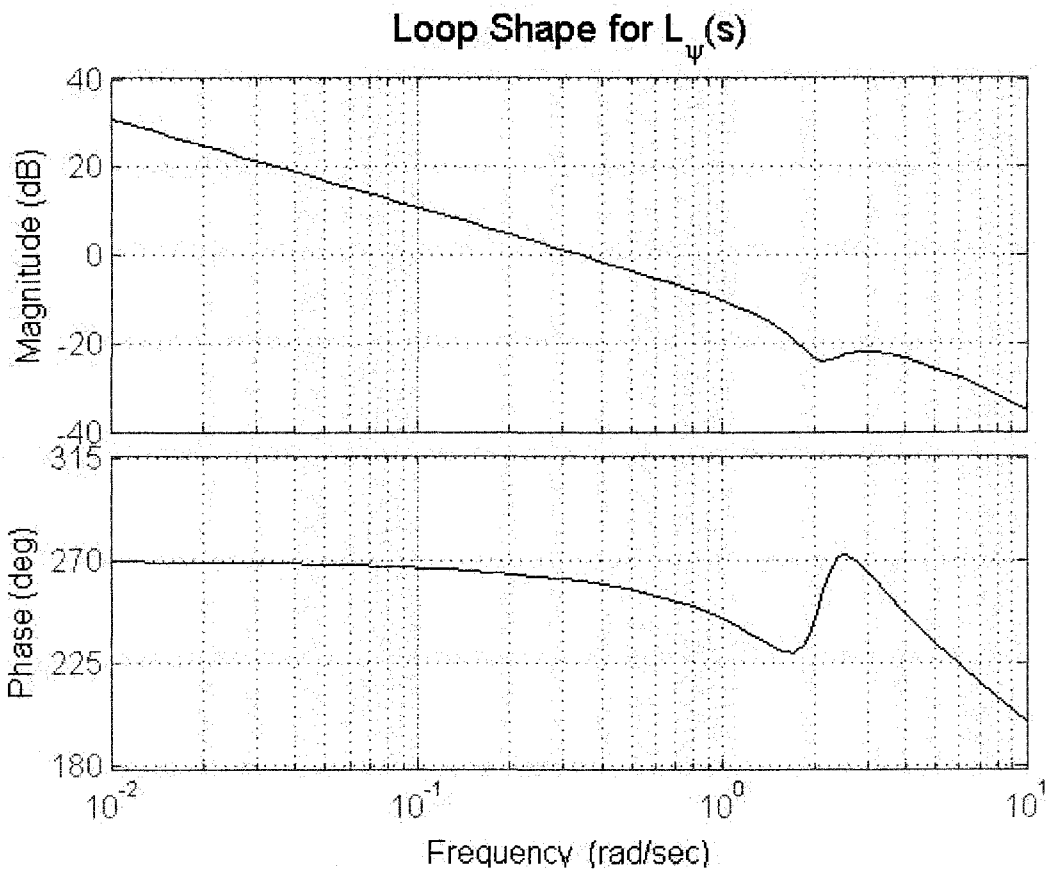


Fig. 4.11. Notice that the loop shape has a phase margin of 90 degrees and a slope of -20 dB/dec. at cross-over, where $L_{\psi}(s)$ is near 0 dB.

We will use loop-shaping once again to close this outer loop. Since it is SISO, we do not need to take into account any coupling effects.

$$G_{\psi_v \phi_c}(s) \approx \frac{-0.010992(s-3288)(s^2+0.6434s+4.395)}{s(s+5.62)(s+34.04)(s^2+1.713s+4.395)}$$

To cross over at 0.3 rad./sec. we simply use a gain of $C_{\psi_v \phi_c}(s) = 2$. The simplicity of the controller is quite good and the behavior is approximately $1/s$-like in the cross-over region, which is good. This is the loop shaping principle.

Example 4.7 – MATLAB code for Decentralized Design Navion

```
clc
clear all

%lateral Navion at SL mach 0.158
%Calculate ff precomp with phi and r as outputs

Uo=176;
bank=0/180*pi;
cbank=cos(bank);
Yv=-.2543;
Lbeta=-15.982;
Nbeta=4.495;
Lp=-8.402;
Np=-0.3498;
Lr=2.193;
Nr=-0.7605;

Ysda=0;
Lda=28.984;
Nda=-0.2218;
Ysdr=.0708;
Ldr=2.193;
Ndr=-4.597;

Alat=[Yv 0 -1 32.2/Uo*cbank 0;Lbeta Lp Lr 0 0;
    Nbeta Np Nr 0 0;0 1 0 0 0;0 0 1 0 0];
Blat=[Ysda Ysdr;Lda Ldr;Nda Ndr;0 0;0 0];
Clat=[0 0 0 1 0;0 0 1 0 0;1 0 0 0 1];
Dlat=[zeros(3,2)];

syslat=ss(Alat,Blat,Clat,Dlat);

C11=zpk([-0.009 -8.4],[0 -40],7);
C22=zpk([0],[-0.2],-.1);

Til=feedback(syslat*[C11 0;0 C22],...
    [1 0 0;0 1 0]); %close inner loops

%sisotool(Til(3,1))
```

Example 4.8

Now we consider longitudinal control of the navion. We will preemptively design a pitch attitude controller and pair that with the elevator. The figure for this controller is shown in Fig. 4.12.

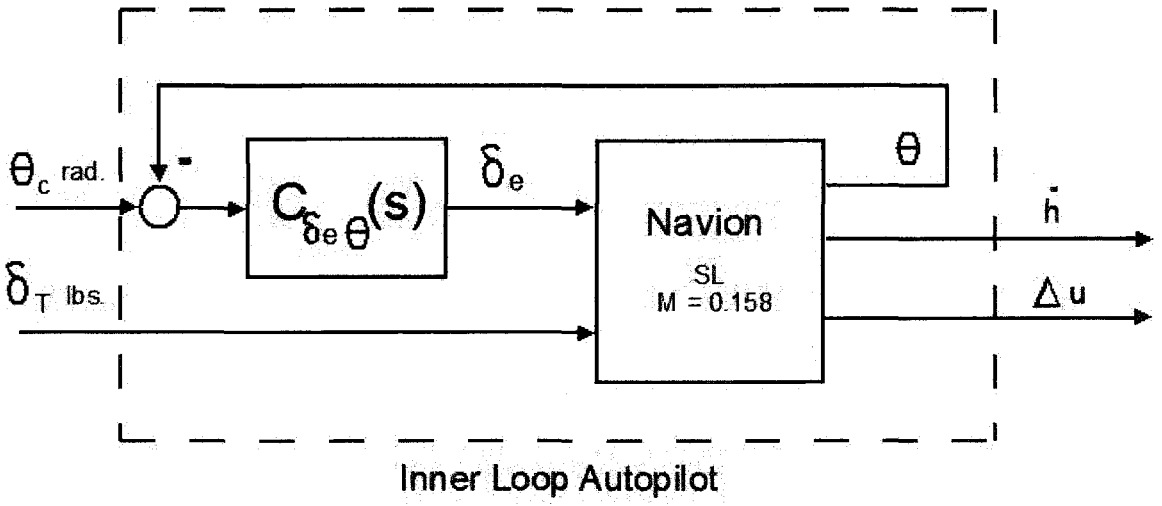


Fig. 4.12 We design an inner loop autopilot which pairs pitch angle with the elevator.

The achieved loop-shape is shown in Fig. 4.13. Notice that there is a -20 dB/dec. slope near crossover at 5 rad./sec. This is the fundamental idea of loop shaping.

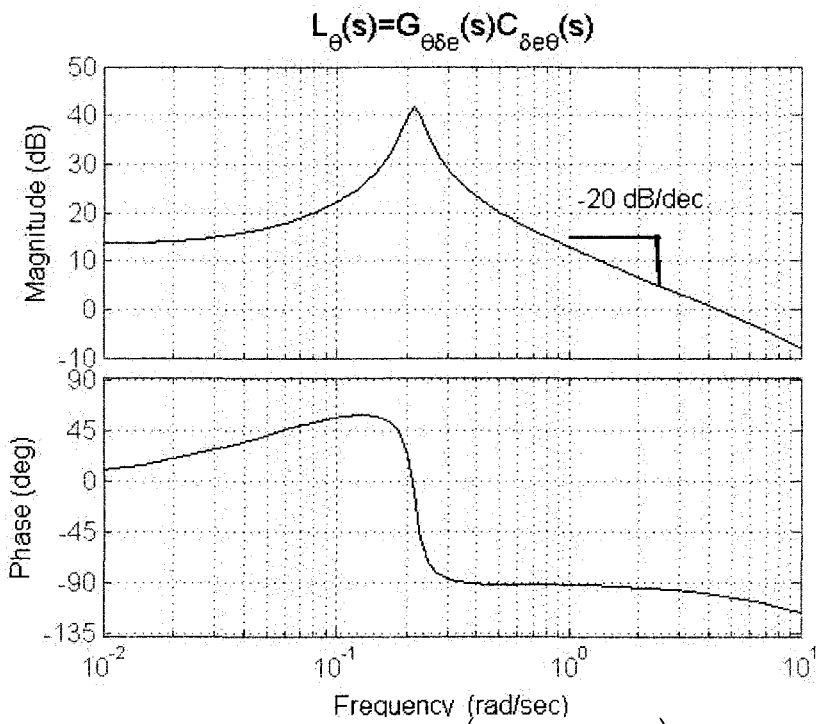


Fig. 4.13 Loop shape with $C_{\delta\theta}(s)=\dfrac{-8(s^2+5s+12.5)}{(s+2)(s+20)}$ for inner loop.

Notice that the loop shape has a phase of -90 degrees and a slope of -20 dB/dec. at cross-over, where $L_\theta(s)$ is near 0 dB.

Integral action will not be pursued in this loop, however, it will be pursued in outer loop closures.

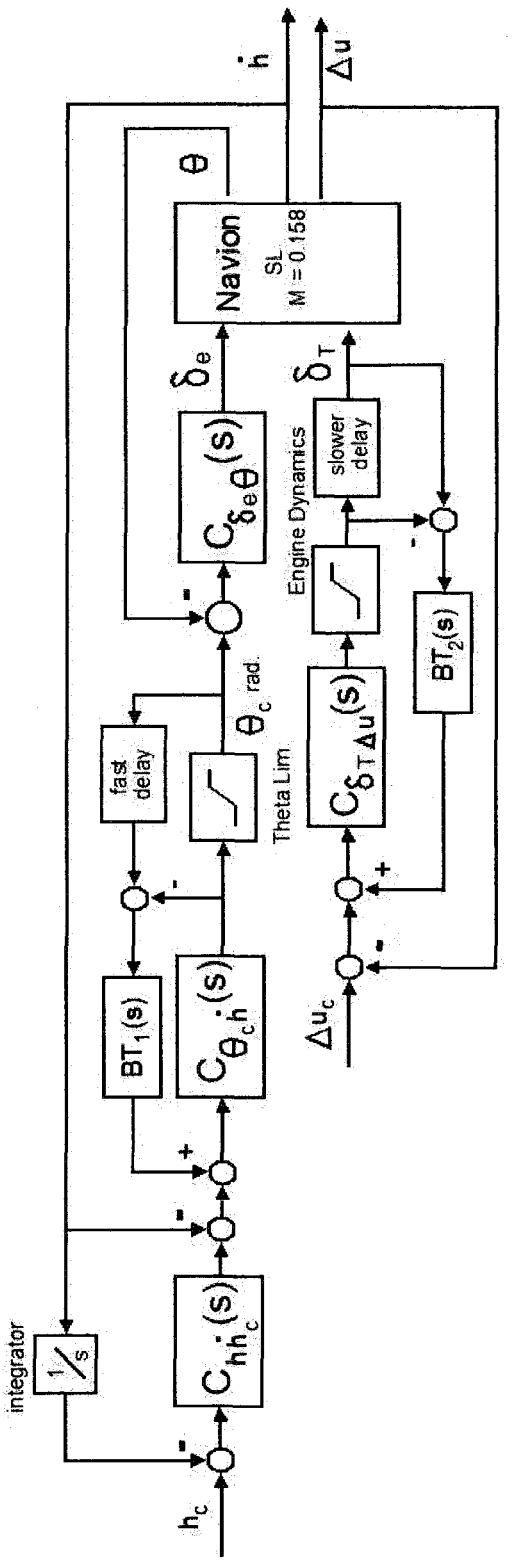


Fig. 4.14. One possible implementation of an altitude and speed control autopilot. It is important to include saturation elements in any outer loop autopilot. The placement of these saturation elements can vary somewhat.

The outer loop closures are shown in Fig. 4.14. This pairing will be analyzed with the RGA and with coupling numerators.

Our idea is that we tell the aircraft to climb by pitching the nose up, then we also tell the aircraft to speed up by increasing thrust. This behavior is less pronounced with aircraft that must land near its stall speed. In this flight condition, there are usually very low frequency transmission zeros for the outputs of altitude and forward velocity. This makes landing a plane very difficult for both the pilot and the autopilot.

We have paired forward speed with thrust , $\Delta u \rightarrow \delta_T$, and we have paired altitude rate with pitch angle command, $\dot{h} \rightarrow \theta_c$.

Now we will analyze the pairing by computing the coupling numerators. They are as follows.

$$G_{\Delta u \delta T}(s)\big|_{\dot{h}\rightarrow\theta_c} \approx \frac{0.01171}{(s+0.01653)}$$

$$G_{\dot{h}\theta_c}(s)\big|_{\Delta u\rightarrow\delta T} \approx \frac{-225.36(s-10.2)}{(s+6.734)(s+1.862)}$$

$$G_{\Delta u\theta_c}(s)\big|_{\dot{h}\rightarrow\delta T} \approx \frac{-600(s-10.2)}{(s+7.025)}$$

$$G_{\dot{h}\delta T}(s)\big|_{\Delta u\rightarrow\theta_c} \approx \frac{-0.3246(s-10.2)(s+13.19)}{(s-284.1)(s+2.403)}$$

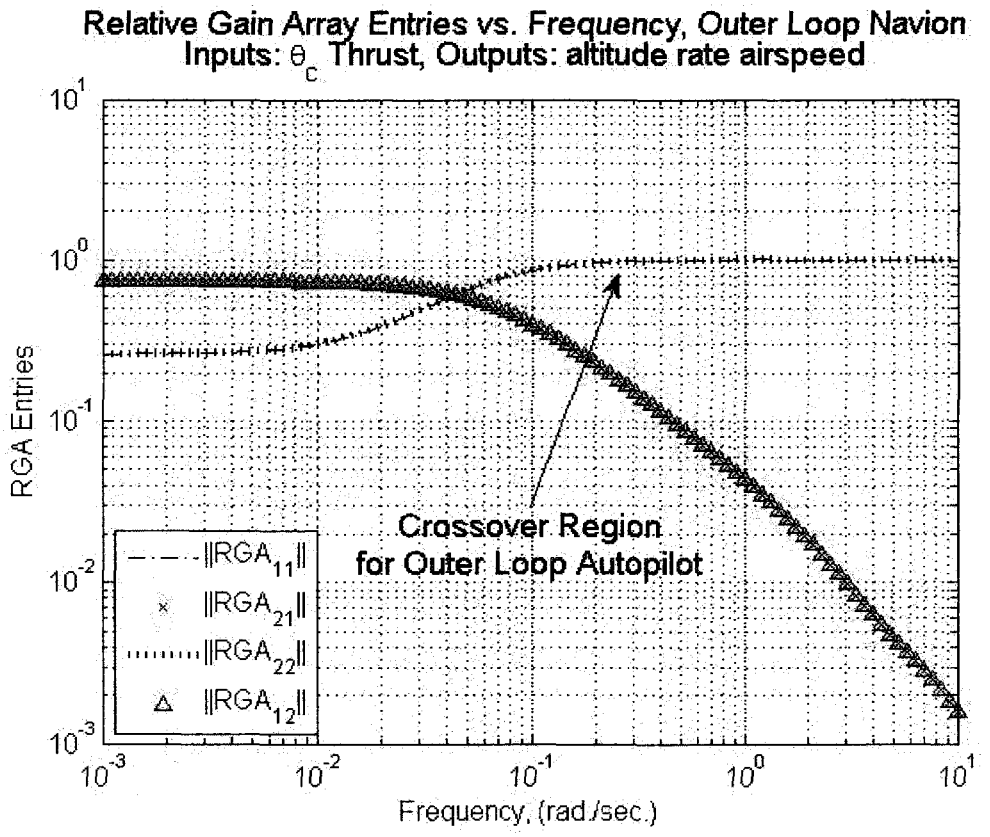


Fig. 4.15. The RGA indicates that a diagonal pairing would be poor at very low frequency. However, in the realm of 0.1 rad./sec.to 1 rad./sec. the plant becomes more diagonal. The coupling numerators show that a diagonal controller will be stable, however there may be strange performance at low frequency.

From the coupling numerators, it would be a bad idea to pair $\Delta u \rightarrow \theta_c$ because of the unstable pole of $s - 284.1$ in the denominator of $G_{h\delta T}(s)\big|_{\Delta u \rightarrow \theta_c}$. It would seem as though pairing forward velocity with thrust, $\Delta u \rightarrow \delta_T$, and pairing altitude rate with theta command, $\dot{h} \rightarrow \theta_c$ is a much better idea.

One should note that the coupling numerators are used as indicators for high gain feedback, not low gain feedback. We plot the RGA vs frequency in Fig. 4.15. We see that the diagonal pairing suggested by the coupling numerators is not very good at low frequencies of .01 rad./sec. However, it improves dramatically as we go towards higher frequency. Outer loop longitudinal autopilots typically have bandwidths ranging between 1 rad./sec. and 0.3 rad./sec. in both channels. It is not an odd coincidence that the RGA values become strongly diagonal at that point.

We will now continue with our loop-shaping procedure by closing the outer loops. From coupling numerators, it looks as though we will mostly be using PI controllers to achieve these outer loop closures.

The first loop closure will be paired $\dot{h} \rightarrow \theta_c$. The loop shape is shown in Fig. 4.16. The loop shape of Fig. 4.14 loses magnitude at low frequency. This was intentionally left alone because we will be taking of these low frequency dynamics with the $\dot{h} \rightarrow \theta_c$ loop closure. We pursue a bandwidth in the range of 0.75 to 1 rad./sec. which is very close the cross-over region. The controller is a PI type controller and is $C_{\theta_c \dot{h}}(s) = \dfrac{0.0025(s+1.8)}{s}$.

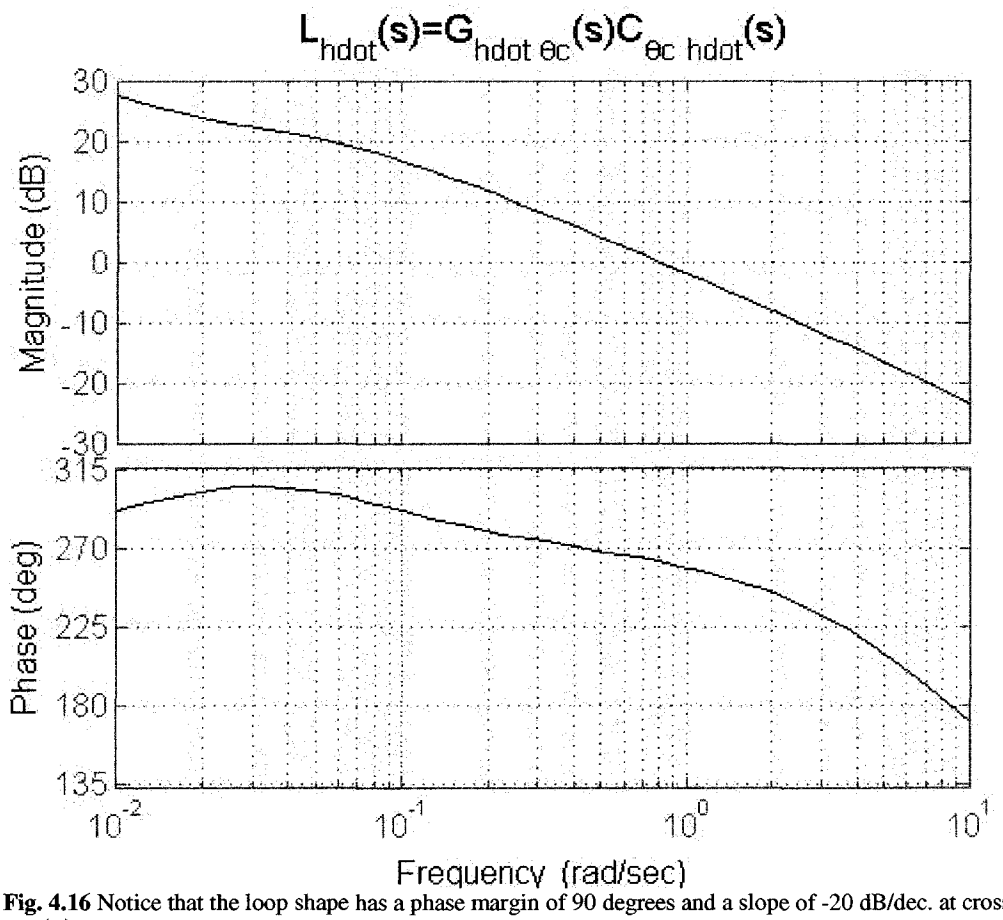


Fig. 4.16 Notice that the loop shape has a phase margin of 90 degrees and a slope of -20 dB/dec. at cross-over, where $L_{\dot{h}}(s)$ is near 0 dB.

Next, we close the $\Delta u \rightarrow \delta_T$ loop, the controller is a PI type and crosses over at around 0.3 rad./sec., which is about one third of the bandwidth of the low frequency dynamics. This gives a PI controller with values of $C_{\delta T \Delta u}(s) = \frac{25(s+0.015)}{s}$

Although we will not show the design of $C_{\dot{h}ch}(s)$, understand that it is designed using the same loopshaping principle as always. A value of $C_{\dot{h}ch}(s) = 0.3$ was chosen, to achieve a bandwidth of 0.3 rad./sec. in the altitude loop.

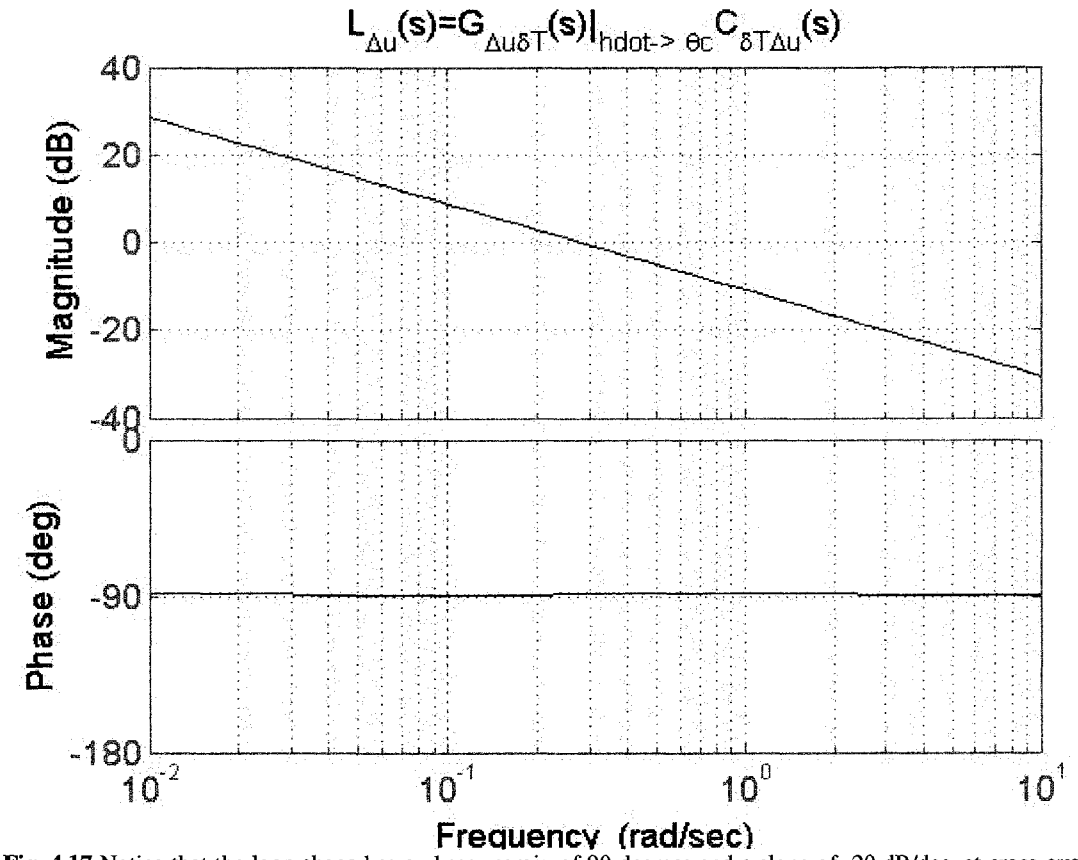


Fig. 4.17 Notice that the loop shape has a phase margin of 90 degrees and a slope of -20 dB/dec. at cross-over, where $L_{\Delta u}(s)$ is near 0 dB.

MATLAB code used to design this controller is shown on the next page.

We will now discuss the $BT_1(s)$ and $BT_2(s)$ transfer functions. Both of these transfer functions are Bumpless Transfer (BT) designs. We will be using the Hanus bumpless transfer technique. Other techniques do exist, though they are mostly for strictly proper or observer based controllers. For

Example 4.8 – MATLAB code for Decentralized Design Navion

```
clc
clear all

g=32.2; %ft/sec/sec
Uo=176;

Xw=0.03607;
Xu=-0.0451;
Zw=-2.0244;
Zu=-0.3697;
Zq=0;
Mu=0;
Mw=-0.04997;
Mq=-2.0767;
Mwdot=-0.005165;
Zwdot=0;
Xde=0;
Zde=-28.17;
Mde=-11.1892;
XdT=1/85.4; %lbs/slugs

Alon=[Xu Xw 0 -g;1/(1-Zwdot)*[Zu Zw Zq+Uo 0];
    Mu+Mwdot*Zu/(1-Zwdot) Mw+Mwdot*Zw/(1-Zwdot)...
    Mq+Mwdot*(Zq+Uo)/(1-Zwdot) 0;
    0 0 1 0];
Blon=[0 XdT;Zde 0;Mde 0;0 0];
Clon=[0 0 0 1;0 -1 0 Uo;1 0 0 0];
syslon=ss(Alon,Blon,Clon,zeros(3,2));

C11=zpk([-2.5+2.5j -2.5-2.5j],[-2 -20],-8);
Til=feedback(syslon*[C11 0;0 1],[1 0 0;0 0 0]);

Gouter=Til(2:3,1:2);

[y2u2_y1u1,y1u1_y2u2,y2u1_y1u2,y1u2_y2u1]...
    = coupling_num_2(Gouter.a,Gouter.b,Gouter.c,Gouter.d);

plotRGA2by2(Gouter.a,Gouter.b,Gouter.c,Gouter.d);

%sisotool(Gouter(1,1))
Couter11=zpk([-1.8],[0],0.0025);
Gouter2=feedback(Gouter*[Couter11 0;0 1],[1 0;0 0]);

%sisotool(Gouter(2,2))
Couter22=zpk([-0.015],[0],25);
Touter=feedback(Gouter*[Couter11 0;0 Couter22],eye(2));
```

proper controllers, it is easier to use the hanus technique. Designing a BT with the hanus configuration is like designing a controller for a controller. The original controller $C(s)$ is the "plant" and the $BT(s)$ is

the transfer function is the "controller".

If the $BT(s)$ is to be used for purposes other than anti-wind up, then it is best not to achieve integral action. For instance, one can use it for a bumpless controller transfer, which is what one can use to swap between two controllers. One can achieve better performance with bumpless transfer, discussed in section 2.28, with by digitally sampling the bumpless transfer loop at a higher rate than the normal control loop. However, this requires multiple digital sampling rates and is more difficult to implement. If one assumes a digital sampling rate of 0.1 seconds, reasonable bumpless transfers can be given by: $BT_1(s)=\frac{s}{0.0025(s+1.8)}\frac{8}{(s+1)}$ and $BT_2(s)=\frac{s}{25(s+0.015)}\frac{8}{(s+1)}$. These controllers and bumpless transfers and can be emulated with the MATLAB command "c2d( )" for implementation.

One should note that these bumpless transfer loops are not perfect. They do not actually achieve a perfectly bumpless transfer. However, the bumpless transfers designed above should reduce mild bumps slower than roughly 0.2 seconds by about 80 to 90 percent.

As a final note, one should add that the navion may not actually need any feedback at all to meet handling qualities requirements. Another possible autopilot design for the navion is shown below.

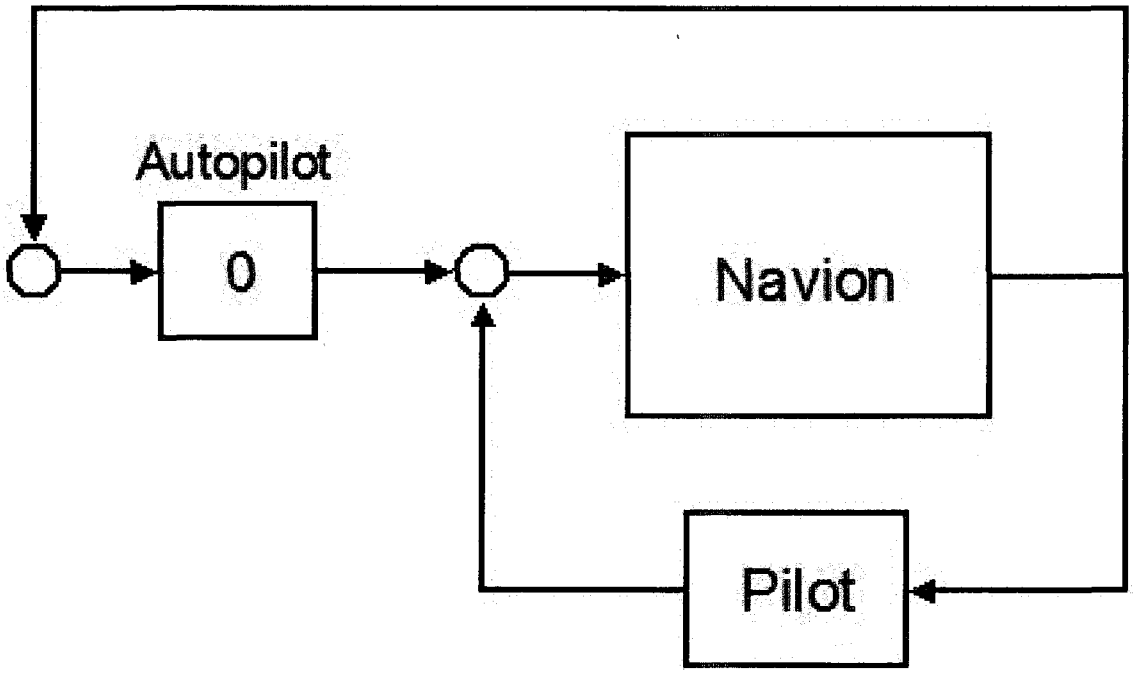


Fig. 4.18 The navion will most likely meet handling qualities requirements without any feedback.

4.4 Wing Damaged Navion Dynamics

In the interests of making a model more accessible to an engineer, the following linear damage model was simplified from Ref. [1.19]. There were just too many numbers. This simplified model is percieved to be reasonably accurate up to about 20-30% wing loss for an aircraft originally trimmed at a low angle of attack. We also assume that the aircraft was well designed and met $I_{xz}/I_{xx} \approx 0.05 \quad \text{to} \quad 0.1$ in its original configuration. Also, an assumption of low flight path angle is made so that $\theta_o \approx 0$. Additionally, we assume that the wind axis is approximately aligned with the body axis. These assumptions are typically good for a cruising flight condition, but usually not a landing flight condition, where the angle of attack can become large.

Of principle interest is the particular aerodynamic force, $\frac{dL}{dw} \approx \frac{dL}{d\alpha}\frac{1}{U_o}$, *which is rolling moment with respect to angle of attack and not lift!*

$$
\begin{aligned}
&dmg < 0.20 \quad or \quad 0.30 \\
&right \quad wing: \quad \frac{dL}{dw} \approx -\frac{dZ}{dw}\frac{b}{2}dmg \\
&left \quad wing: \quad \frac{dL}{dw} \approx \frac{dZ}{dw}\frac{b}{2}dmg
\end{aligned}
\tag{4.7}
$$

With this stability derivative present in the damaged aircraft, the system matrix would be triangular, and stability would not be affected. This is shown in the equations below. The A_{lonlon} matrix would be the longitudinal system matrix. The A_{latlon} matrix would consist primarily of the dL/dw parameter and not much else. The A_{latlat} matrix would consist be the lateral system matrix.

$$
\begin{aligned}
&A_{dmg} = \begin{bmatrix} A_{lonlon} & 0 \\ A_{latlon} & A_{latlat} \end{bmatrix} \\
&\boldsymbol{det}(sI - A_{dmg}) = \boldsymbol{det}\left(\begin{bmatrix} (sI - A_{lon}) & 0 \\ 0 & I \end{bmatrix}\right)\boldsymbol{det}\left(\begin{bmatrix} I & 0 \\ 0 & (sI - A_{lat}) \end{bmatrix}\right)\boldsymbol{det}\left(\begin{bmatrix} I & 0 \\ A_{latlon} & I \end{bmatrix}\right) \\
&\boldsymbol{det}(sI - A_{dmg}) = \boldsymbol{det}(sI - A_{lon})\boldsymbol{det}(sI - A_{lat})
\end{aligned}
\tag{4.8}
$$

The author believes that the system matrix is mostly triangular for mild damage (less than 20 or 30%). Therefore, the author expects the aircraft to be *dynamically* stable in the presence of mild damage (less than 20 or 30%). The author believes that the aircraft might not be *statically* stable in the presence of mild damage. Ailerons might have to be resized to accommodate larger wing loss.

One should note that while the eigenvalues of the system matrix would not change, the eigenvectors of the system matrix would change. Basically, one should expect the longitudinal system variables to affect the lateral system, but not vice versa. This may lead to strange motions of the aircraft.

We will discuss some small things which would make A_{lonlat} non-zero and potentially cause instability.

Another aerodynamic stability derivative of lesser importance is downward force produced by the aileron. For a normal pair of ailerons on a small and lightweight aircraft, the ailerons produce no net Z-axis forces. However, on larger aircraft and some fighter models, an asymmetric flap can be used, which may cause small Z-axis forces. Oftentimes, these forces are neglected in control design. For instance, the USAF DATCOM estimates that the downward force with respect to asymmetric spoiler is zero! This is most likely because these forces actually are quite small. Nonetheless, we introduce an arbitrary number, designated arb, which is left for the engineer to decide on how big or small this gain is.

$$\begin{aligned} right \quad wing: \quad Z_{\delta a} &\approx \frac{L_{\delta a}}{b} dmg\,(arb) \\ left \quad wing: \quad Z_{\delta a} &\approx \frac{-L_{\delta a}}{b} dmg\,(arb) \end{aligned} \tag{4.9}$$

For small and lightweight aircraft, there is the potential for a lateral shift in the center of gravity, Δy. We assume is that Δy is non-zero and Δy^2 is approximately 0. Using schur compliments and assuming diagonal mass properties, one can simplify the equations of motion as follows.

$$f_1 = \frac{1}{m} + \frac{\Delta y^2}{\left(I_{zz} - m\Delta y^2\right)} \approx \frac{1}{m} \quad f_2 = \frac{1}{m} + \frac{\Delta y^2}{\left(I_{xx} - m\Delta y^2\right)} \approx \frac{1}{m} \quad f_3 = \frac{1}{\left(I_{xx} - m\Delta y^2\right)} \approx \frac{1}{I_{xx}}$$

$$f_4 = \frac{1}{\left(I_{zz} - m\Delta y^2\right)} \approx \frac{1}{I_{zz}} \quad f_5 = \frac{-\Delta y}{\left(I_{xx} - m\Delta y^2\right)} \approx \frac{\Delta y}{I_{xx}} \quad f_6 = \frac{\Delta y}{\left(I_{zz} - m\Delta y^2\right)} \approx \frac{\Delta y}{I_{zz}}$$

$$X_v = \frac{dN}{dv} f_6 \quad X_p = \frac{dN}{dp} f_6 \quad X_r = \frac{dN}{dr} f_6$$

$$Z_w = \frac{dZ}{dw} f_2 + \frac{dL}{dw} f_3 \quad Z_v = \frac{dL}{dv} f_5 \quad Z_p = \frac{dL}{dp} f_5 \quad Z_r = \frac{dL}{dr} f_5 \tag{4.10}$$

$Z_u, X_u, X_w, M_u, M_w, M_q$ *are unaffected*

Y_v *is unaffected*

$$L_u = \frac{dZ}{du} f_5 \quad L_w = \frac{dL}{dw} f_3 + \frac{dZ}{dw} f_5 \quad L_v = \frac{dL}{dv} f_3 \quad L_p = \frac{dL}{dp} f_3 \quad L_r = \frac{dL}{dp} f_3$$

$$N_u = \frac{dX}{du} f_6 \quad N_w = \frac{dX}{dw} f_6 \quad N_v = \frac{dN}{dv} f_4 \quad N_p = \frac{dN}{dp} f_4 \quad N_r = \frac{dN}{dr} f_4$$

$$\begin{aligned}
\Delta\dot{u} &= X_u \Delta u + X_w w - g\theta + (X_v U_o)\beta + X_p p + X_r r + X_{\delta T}\delta_T \\
\dot{w} &= Z_u \Delta u + Z_w w + U_o q + (Z_v U_o)\beta + Z_p p + Z_r r + Z_{\dot{\alpha}}\delta_e + Z_{\delta a}\delta_a \\
\dot{q} &= M_u \Delta u + M_w w + M_q q - g\theta + M_{\dot{\alpha}}\delta_e \\
\dot{\theta} &= q \\
\dot{\beta} &= Y_v \beta - r + {}^{g}\!/_{U_o}\,\phi + Y^*_{\delta a}\delta_a \\
\dot{p} &= L_u \Delta u + L_w w + L_\beta \beta + L_p p + L_r r + L_{\delta a}\delta_a + L_{\delta r}\delta_r \\
\dot{r} &= N_u \Delta u + N_w w + N_\beta \beta + N_p p + N_r r + N_{\delta a}\delta_a + N_{\delta r}\delta_r \\
\dot{\phi} &= p
\end{aligned} \tag{4.11}$$

We use the nominal stability derivatives from the previous design examples. We add a lateral cg shift of $\Delta y = -1 \quad ft.$ with $dmg = 0.2$. The system matrices are shown below.

$$A_{lonlon} = \begin{bmatrix} -0.0451 & 0.0361 & 0 & -32.2 \\ -0.3697 & -1.4734 & 176 & 0 \\ 0.0019 & -0.0424 & -2.9857 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$A_{latlon} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0.0301 & 0.6701 & 0 & 0 \\ -0.0011 & 0.0009 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$A_{latlat} = \begin{bmatrix} -0.2543 & 0 & 0 & 0.183 \\ -15.982 & -8.402 & 2.193 & 0 \\ 4.495 & -0.3498 & -0.7605 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$A_{lonlat} = \begin{bmatrix} 4.495 & -0.3498 & -0.7605 & 0 \\ 15.982 & 8.402 & -2.193 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

One should note that the damaged navion is open loop stable, with eigenvalues of $\lambda[A_{dmg}] = (-7.989, -2.42 \pm 2.8j, -0.5 \pm 2.34j, -0.0168 \pm 0.2093j, -0.0088)$.

For now, we will only include the inner loop control inputs of elevator, δ_e, aileron, δ_a, and rudder, δ_r, respectively.

$$B_{latlat} = \begin{bmatrix} 0 & 0.0708 \\ 28.984 & 2.193 \\ -0.2218 & -4.597 \\ 0 & 0 \end{bmatrix} \quad B_{lonlat} = \begin{bmatrix} 0 & 0 \\ 0.0868 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$B_{lonlon} = \begin{bmatrix} 0 \\ -28.17 \\ -11.1892 \\ 0 \end{bmatrix} \quad B_{latlon} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$B_{dmg} = \begin{bmatrix} B_{lonlon} & B_{lonlat} \\ B_{latlon} & B_{latlat} \end{bmatrix}$$

We assume that the outputs will be θ, ϕ and r, respectively. The output matrix is listed below.

$$C_{dmg} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \ldots \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

We use the diagonal controller designed in previous examples, which is repeated below.

$$C_{\delta e\theta}(s)=\frac{-8(s^2+5s+12.5)}{(s+2)(s+20)},\ C_{\delta a\phi}(s)=\frac{7(s+0.009)(s+8.4)}{s(s+40)},$$

$$C_{\delta r r}(s)=\frac{-k_{wo}s}{(s+0.2)}\quad k_{wo}=0.1.$$

With this controller, the closed loop system is still stable, which is good.

$$\lambda[A_{dmg}]=(-34,-13.1,-6.28\pm 2.8j,-2.4\pm 2.62j,-0.85\pm 2.03j$$
$$\ldots -0.9808,-0.0725j,-0.1973,-0.0090)$$

MATLAB code for the damaged Navion is shown below.

Example 4.9 – MATLAB code for Damaged Navion

```
clc
clear all

g=32.2; %mass properties
Uo=176; %ft/sec.
bank=0/180*pi; %bank angle
cbank=cos(bank);

%mass properties
m=2750/g; %slugs
Ixx=1048; %slugs*ft^2
Iyy=3000; %slugs*ft^2
Izz=3530; %slugs*ft^2
Ixz=0;    %slugs*ft^2

S=184; %ft^2
dy=-1;  %ft
b=33.4; %ft

dmg=0.2;

%longitudinal stability derivatives
Yv=-.2543;
Lbeta=-15.982;
Nbeta=4.495;
Lp=-8.402;
Np=-0.3498;
Lr=2.193;
Nr=-0.7605;

Ysda=0;
Lda=28.984;
Nda=-0.2218;
Ysdr=.0708;
Ldr=2.193;
Ndr=-4.597;

%longitudinal stability derivatives
Xw=0.03607;
Xu=-0.0451;
Zw=-2.0244;
Zu=-0.3697;
```

Example 4.9 – MATLAB code for Damaged Navion, continued

```
Zq=0;
Mu=0;
Mw=-0.04997;
Mq=-2.0767;
Mwdot=-0.005165;
Zwdot=0;
Xde=0;
Zde=-28.17;
Mde=-11.1892;
XdT=1/85.4; %lbs/slugs
%weird stability derivatives
dLdw=(-Zw*m)*b/2*dmg; %aerodynamics, strip theory
Zda=Lda/b*dmg*0.5;
%weird mass properties
f2=1/m;f3=1/Ixx;f5=-dy/Ixx;f6=dy/Izz;
f7=2*m*dy/Ixx;
Xv=Nbeta*1/Uo*Izz*f6;
Xp=Np*Izz*f6;
Xr=Nr*Izz*f6;
Zw=Zw*m*f2+dLdw*f3;
Zv=Lbeta*1/Uo*Ixx*f5;
Zp=Lp*Ixx*f5;
Zr=Lr*Ixx*f5;
Lu=Zu*m*f5;
Lw=dLdw*f3+Zw*m*f5;
Nu=Xu*m*f6;
Nw=Xw*m*f6;
Alonlon=[Xu Xw 0 -g;1/(1-Zwdot)*[Zu Zw Zq+Uo 0];
    Mu+Mwdot*Zu/(1-Zwdot) Mw+Mwdot*Zw/(1-Zwdot)...
    Mq+Mwdot*(Zq+Uo)/(1-Zwdot) 0;
    0 0 1 0];
Blonlon=[0 XdT;Zde 0;Mde 0;0 0];
Clonlon=[0 0 0 1;0 -1 0 Uo;1 0 0 0];
Alatlat=[Yv 0 -1 32.2/Uo*cbank;Lbeta Lp Lr 0;
    Nbeta Np Nr 0;0 1 0 0];
Blatlat=[Ysda Ysdr;Lda Ldr;Nda Ndr;0 0];
Clatlat=[0 0 0 1;0 0 1 0;Uo*Yv 0 -Uo*0 32.2*cbank];
Dlatlat=[zeros(2,2);Ysda*Uo    Ysdr*Uo];
Alatlon=[0 0 0 0;Lu Lw 0 0;Nu Nw 0 0;0 0 0 0];
Alonlat=[Xv*Uo Xp Xr 0;Zv*Uo Zp Zr 0;0 0 0 0;
    0 0 0 0];
Admg=[Alonlon Alonlat;Alatlon Alatlat];
Cdmg=[0 0 0 1 zeros(1,4);...
    zeros(1,7) 1;zeros(1,6) 1 0];
Blonlat=[0 0;Zda 0;0 0;0 0];
Bdmg=[Blonlon(:,1) Blonlat;zeros(4,1) Blatlat];
C11=zpk([-2.5+2.5j -2.5-2.5j],[-2 -20],-8);
C22=zpk([-0.009 -8.4],[0 -40],7);
C33=zpk([0],[-0.2],-.1);
sysdmg=ss(Admg,Bdmg,Cdmg,zeros(3));
syscl=feedback(sysdmg*[C11 0 0;0 C22 0;0 0 C33],eye(3)); %stable
```

4.5 Inner Loop M-Delta Stability Analysis of Damaged Navion

M-Delta stability analysis is always a difficult thing to do when singular value inequalities are not met. Remind yourself that if $\bar{\sigma}(M_A(j\omega)\Delta_A(j\omega))<1$, then stability is guaranteed. However, if $\bar{\sigma}(M_A(j\omega)\Delta_A(j\omega))>1$, then stability is unknown.

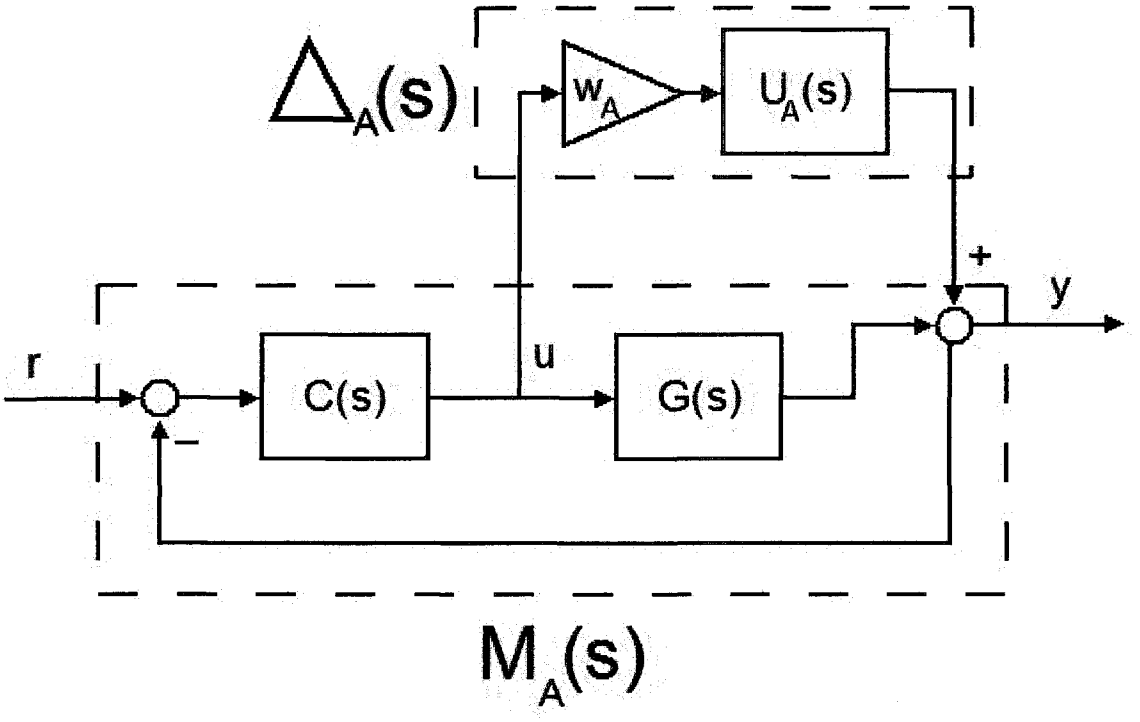


Fig. 4.19. M-Delta form for additive input uncertainty

We use the controller which was designed in previous design examples.

$$C(s)=\begin{bmatrix} \dfrac{-8(s^2+5s+12.5)}{(s+2)(s+20)} & 0 & 0 \\ 0 & \dfrac{7(s+0.009)(s+8.4)}{s(s+40)} & 0 \\ 0 & 0 & \dfrac{-0.1s}{(s+0.2)} \end{bmatrix}$$

In this case, we will view the uncertainty as $G^p(s)=G^{dmg}(s)$ and $\Delta_A(s)=G^p(s)-G(s)$.

$$\begin{aligned} M_A(s)&=-(I+C(s)G(s))^{-1}C(s)=-C(s)(I+G(s)C(s))^{-1}=S_{nI}(s) \\ \Delta_A(s)&=G^p(s)-G(s) \end{aligned} \tag{4.16}$$

As can be seen from Fig. 4.20, the sufficient condition of $\overline{\sigma}(M_A(j\omega)\Delta_A(j\omega))<1$ has not been met, therefore we must perform a search of perceived additive uncertainties and check for instability. Many feedback control books over-emphasize or misrepresent the singular value.

We propose the following solution to find a destabilizing additive uncertainty.

Initialization: Set Δy and $\_dmg$ equal to some reasonable values.

$$\text{Set } \Delta_A(s)=G^p(s)-G(s),\ G^p(s)=G^{dmg}(s)$$

$$U_A(s)=\Delta_A(s) \quad w_A=1$$

Step 1.0: $\Delta_A(s)=w_A U_A(s)$,

Calculate closed loop eigenvalues: $\boldsymbol{det}(I+(G(s)+w_A U_A(s))C(s))$

Step 2.0: If : $w_A < \boldsymbol{max}(w_A)$ then exit

Else : Increase w_A return to step 1.0

$\Delta y=-1 \quad ft.$ $dmg=0.2$ $w_A=1$	$\Delta y=-1 \quad ft.$ $dmg=0.2$ $w_A=2$	$\Delta y=-1 \quad ft.$ $dmg=0.2$ $w_A=3$	$\Delta y=-1 \quad ft.$ $dmg=0.2$ $w_A=4$
Closed loop Stable Effective "damage" = 0.2	Closed loop Stable Effective "damage" = 0.4	Closed loop Unstable s = 0.27 Effective "damage" = 0.6	Closed loop Unstable s = 0.79 Effective "damage" = 0.8

Table. 4.1. Stability results by increasing the w_A parameter. The author believes that the spiral root is destabilized by increasing w_A.

Accurate M-delta analysis is always a difficult thing to do. One has to simultaneously know the model accurately and introduce inaccuracies into the model. Simply making up a bunch of numbers may not represent physics very well. If too many numbers are fictitiously introduced, false instability will be indicated. This is a low fidelity model, so the instability with w_A might be false.

According to these computer models of wing damage, the author believes that it is reasonable to expect that almost any reasonable and non-adaptive feedback controller will provide *dynamic* stability in the presence of mild wing damage. However, the author is more concerned about *static* stability in the presence of mild wing damage. The author expects that aircraft whose total roll control surfaces occupy

around 2% of the total wing area and have deflection limitations less than 10 degrees will not be capable of balancing rolling moments at landing speeds with 20 to 30% wing loss. Resizing ailerons would be the most obvious solution, however, this may cause many more problems than it solves. Too much roll control authority may cause upsets for the undamaged aircraft.

Example 4.10 – MATLAB code for M-Delta analysis with Damaged Navion, continued from previous Example 4.9

```
%%%%%% M delta %%%%%%
sysdmg=ss(Admg,Bdmg,Cdmg,zeros(3));
Cs=[C11 0 0;0 C22 0;0 0 C33];
syscl=feedback(sysdmg*Cs,eye(3));

Anom=[Alonlon zeros(4);zeros(4) Alatlat];
Bnom=[Blonlon(:,1) 0*Blonlat;zeros(4,1) Blatlat];
Cnom=Cdmg;
sysnom=ss(Anom,Bnom,Cnom,zeros(3));

delta_add=minreal(sysdmg-sysnom,1e-1);

figure(1);
Ma=-feedback(Cs*sysnom,eye(3))*Cs;
sigma(-Ma*delta_add,'k',{.01,10});
title('Singular values of M_A(s)\Delta_A(s)','FontSize',14);


w_add=linspace(1,4,4); %4 points

figure(2);
for k=1:length(w_add)
    Gp=balreal(sysnom+w_add(k)*delta_add);
    clp=feedback(Gp*Cs,eye(3));
    eigs(:,k)=eig(clp);
    Xeigs(:,k)=real(eigs(:,k));
    Yeigs(:,k)=imag(eigs(:,k));
    scatter(Xeigs(:,k),Yeigs(:,k),k,'k^');
    hold on
    grid on
end
hold off

Gp=sysnom+ultidyn('whatever',[3 3],'Type',...
    'GainBounded','Bound',3)*delta_add;
clp=feedback(Gp*Cs,eye(3));
[STABMARG,DESTABUNC,REPORT,INFO]=robuststab(clp);
%robuststab did not indicate instability
```

4.6 Improving Performance of Wing Damaged Aircraft

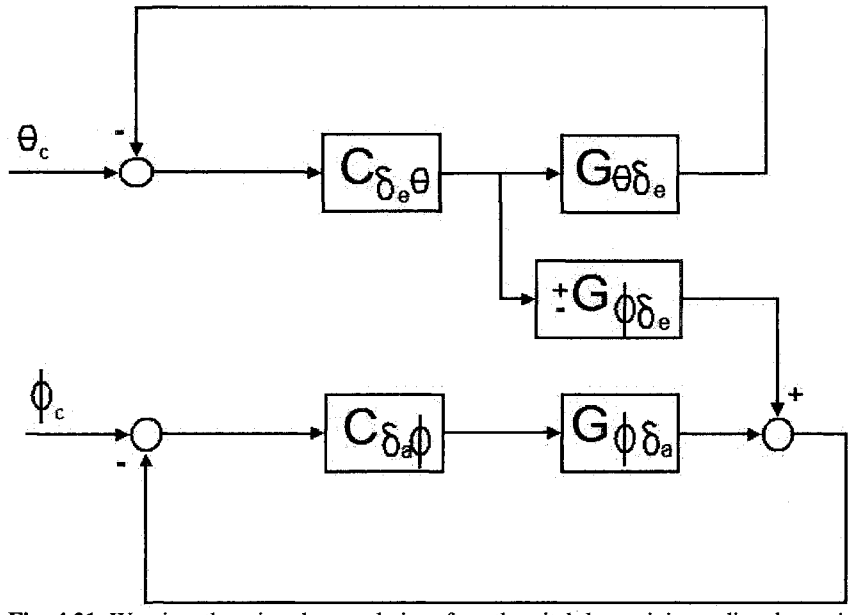


Fig. 4.21. We view the wing damaged aircraft as the pitch loop giving a disturbance into the bank loop. This then just becomes a disturbance rejection problem and we make the bank angle controller more aggressive. This will give an improvement in performance.

We will view the wing damaged aircraft as a triangular plant with an uncertain element, $G_{\phi\delta e}(s)$, which has an uncertain sign because either the left or right wing can be damaged. In this event, one cannot decouple with either technique shown in chapters 5 and 6.

We assume the plant is triangular and described by the following uncertain transfer function.

$$G(s)=\begin{bmatrix} G_{\theta\delta e}(s) & 0 \\ \pm G_{\phi\delta e}(s) & G_{\phi\delta e}(s) \end{bmatrix} \tag{4.17}$$

We will be using a diagonal controller as shown below.

$$C(s)=\begin{bmatrix} C_{\delta e\theta}(s) & 0 \\ 0 & C_{\delta a\phi}(s) \end{bmatrix} \tag{4.18}$$

The complimentary sensitivity is listed below.

$$T(s)=\begin{bmatrix} \dfrac{L_\theta(s)}{(1+L_\theta(s))} & 0 \\ \dfrac{\pm G_{\phi\delta e}(s)C_{\delta e\theta}(s)}{(1+L_\theta(s))(1+L_\phi(s))} & \dfrac{L_\phi(s)}{(1+L_\phi(s))} \end{bmatrix} \tag{4.19}$$

$$L_\theta(s)=G_{\theta\delta e}(s)C_{\delta e\theta}(s) \quad L_\phi(s)=G_{\phi\delta a}(s)C_{\delta a\phi}(s)$$

We can see that if $L_{\phi}(s)$ is large, then $\frac{\pm G_{\phi\delta e}(s)C_{\delta e\theta}(s)}{(I+L_{\theta}(s))(I+L_{\phi}(s))}$ is small. Also, we could reduce $C_{\delta e\theta}(s)$ to make the second element smaller.

4.7 Switched Controllers for Wing Damaged Aircraft

In this section, we will suggest that one could use switched linear controllers between the longitudinal and lateral subsystems to achieve control of the aircraft.

$$G(s)=\begin{bmatrix} G_{lonlon}(s) & G_{lonlat}(s) \\ G_{latlon}(s) & G_{latlat}(s) \end{bmatrix} \tag{4.20}$$

We suppose the controller is block decentralized as follows in eq. 4.21.

$$C(s)=\begin{bmatrix} C_{lon}(s) & 0 \\ 0 & C_{lat}(s) \end{bmatrix} \tag{4.21}$$

We want the return difference, $(I+G(s)C(s))$, to have all of its transmission zeros in the LHP of the jw axis. This equivalent to saying that the poles of $(I+G(s)C(s))^{-1}$ are stable.

$$\boldsymbol{det}(I+G(s)C(s))=0 \quad \forall\, \boldsymbol{Re}\{s\}<0 \tag{4.22}$$

We expand the terms out to give the expression below.

$$\boldsymbol{det}\left(\begin{bmatrix} (I+G_{lonlon}(s)C_{lon}(s)) & G_{lonlat}(s)C_{lat}(s) \\ G_{latlon}(s)C_{lon}(s) & (I+G_{latlat}(s)C_{lat}(s)) \end{bmatrix}\right)=0 \tag{4.23}$$

Using schur complements, we can decompose the determinant to read the following below.

$$\begin{aligned} &\boldsymbol{det}(I+G_{lonlon}(s)C_{lon}(s))\ldots \\ &\boldsymbol{det}\left(I+G_{latlat}(s)C_{lat}(s)-G_{latlon}(s)C_{lon}(s)(I+G_{lonlon}(s)C_{lon})^{-1}G_{lonlat}(s)C_{lat}(s)\right)=0 \end{aligned} \tag{4.24}$$

Assuming that $C_{lon}(s)=0$, the closed loop plant has closed loop poles corresponding to following expression below.

$$\boldsymbol{det}(I+G_{latlat}(s)C_{lat}(s))=0 \tag{4.25}$$

Provided that the other transfer functions, $G_{lonlon}(s), G_{latlon}(s), G_{lonlat}(s)$, are stable, one can expect stability.

Assuming that $C_{lat}(s) = 0$, the closed loop plant has the plant has closed loop poles corresponding to following expression below.

$$\boldsymbol{det}(I + G_{lonlon}(s)C_{lon}(s)) = 0 \tag{4.26}$$

Provided that the other transfer functions are stable, $G_{latlat}(s), G_{lonlon}(s), G_{latlon}(s)$, are stable, one can expect stability.

If one switches between two controller modes, $[C_{lon}(s) \quad off, \quad C_{lat}(s) \quad on]$, and $[C_{lon}(s) \quad on, \quad C_{lat}(s) \quad off]$, then one can expect stability when in either of those modes. Further discussion of stability of switched linear systems is discussed in Chapter 8.

Chapter 5 – Case Studies in Feedforward Action with Decentralized Control

Feedforward action in decentralized control [2.16], is not a generalizable control strategy. It is not difficult to generate counterexamples where this technique can unnecessarily create controllers with unstable poles. Conditions on the solvability and scope of this approach can be found in Ref. [5.1].

Despite its shortcomings, this technique has legitimate value because it allows engineers to think about the control design in a palatable way. It allows engineers the important ability to selectively decouple at different frequencies by simply looking at bode plots. This technique is revisited in section 6.8 because it is sometimes a very bad idea to decouple at all frequencies.

This chapter avoids generality and pursues case studies in aircraft control and deviates once to design a controller using the model from example 2.11.

5.1 A Simple Decoupling Controller Design for the VZ-4 Doak

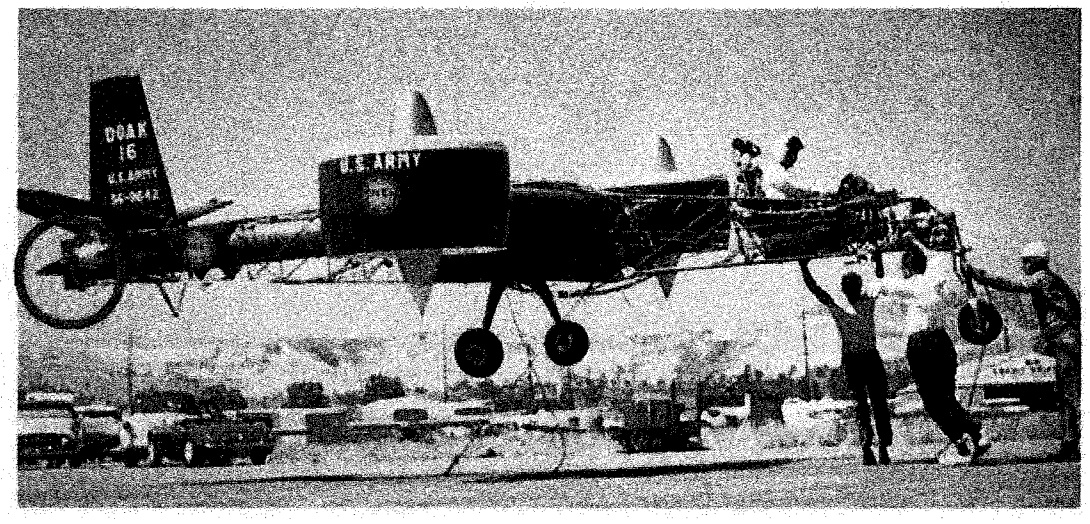


Fig. 5.1. The VZ4 doak was an experimental VTOL (Vertical Take-Off And Landing) aircraft developed in 1956 for the Army. Only one of these was ever produced because the project was cancelled after helicopters showed more promise. Vertical thrust is generated by the two turbofans on the wings and it looks as though pitching moment is generated by vectored thrust at the tail which is circled in red. This will be referred to as "elevator".

This section serves as a very simple and introduction to the idea of designing decoupling controllers. The state space is listed below in equations 5.1-5.3. The state vector is comprised of forward velocity, Δu , downward velocity, Δw, pitch rate, Δq, and pitch angle, $\Delta\theta$. For the first loop closure, we will consider the outputs of pitch angle and altitude rate $\dot{h}$ and the inputs of "elevator" δ_e and thrust, δ_T.

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{5.1}$$

$$A = \begin{bmatrix} -0.137 & 0 & 0 & -32.2 \\ 0 & -0.137 & 0 & 0 \\ 0.0136 & 0 & -0.0452 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 1.08 & 1 \\ 1 & 0 \\ 0 & 0 \end{bmatrix} \quad C = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 \end{bmatrix} \tag{5.2}$$

$$y = \begin{bmatrix} \Delta\theta \\ \dot{h} \end{bmatrix} \quad u = \begin{bmatrix} \delta_e \\ \delta_T \end{bmatrix} \quad x = \begin{bmatrix} \Delta u \\ \Delta w \\ \Delta q \\ \Delta\theta \end{bmatrix} \tag{5.3}$$

5.1.1 Pre-compensator Design

By inspecting the *A* matrix, one can see that the up-down motion of the aircraft is a simple first order decay with no influence over the pitching and tipping dynamics. By inspecting the *B* matrix one can see that an increase in "elevator" δ_e not only affects the pitching dynamics, but it also affects the altitude as well. This is due entirely to the B_{21} value of 1.08.

If this B_{21} value of 1.08 could be made to be 0, then there would be no cross coupling between the pitch and altitude rate channels. While it is impossible to change the true *B* matrix of the aircraft

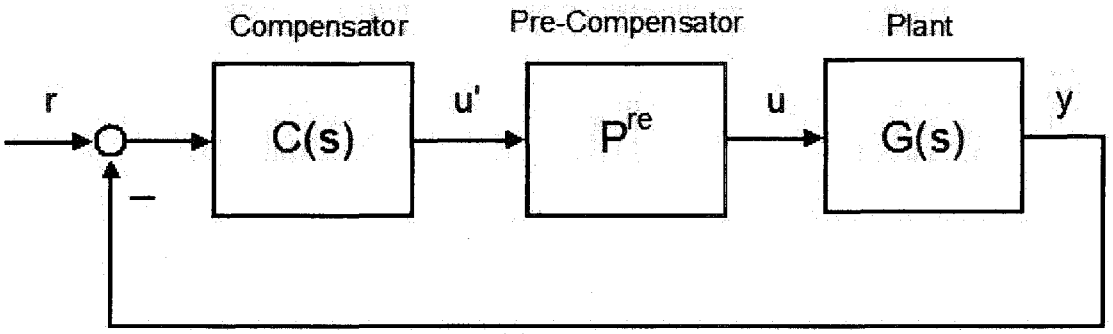


Fig. 5.2. Prior to designing the compensator $C(s)$, a pre-compensator , P^{re} , is designed to approximately decouple or approximately diagonalize the plant. Then the compensator is designed with *u'* as the input.

without physically re-designing the aircraft, it is possible to set up a pre-compensator scheme so that the compensator will see a state space where the B_{21} value is 0 and the outputs will be decoupled.

$$\begin{aligned} \dot{x} &= Ax + BP^{re}u' \quad u = P^{re}u' \\ y &= Cx \end{aligned} \tag{5.4}$$

A picture of such a pre-compensator, P^{re}, is shown in Fig. 5.2.

We can calculate the pre-compensator P^{re}, so that the effective plant, *G(s)* P^{re}, which the compensator sees is completely diagonal and decoupled.

$$P^{re} = \begin{bmatrix} 1 & 0 \\ -1.08 & 1 \end{bmatrix} \quad BP^{re} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 0 \end{bmatrix} \tag{5.5}$$

To understand what this pre-compensator means physically, one looks at the pre-compensator P^{re},, to understand what the inputs *u'* mean.

$$u = P^{re}u' \quad P^{re} = \begin{bmatrix} 1 & 0 \\ -1.08 & 1 \end{bmatrix}$$
$$\delta_e = u_1' \quad \delta_T = -1.08u_1' + u_2' \tag{5.6}$$

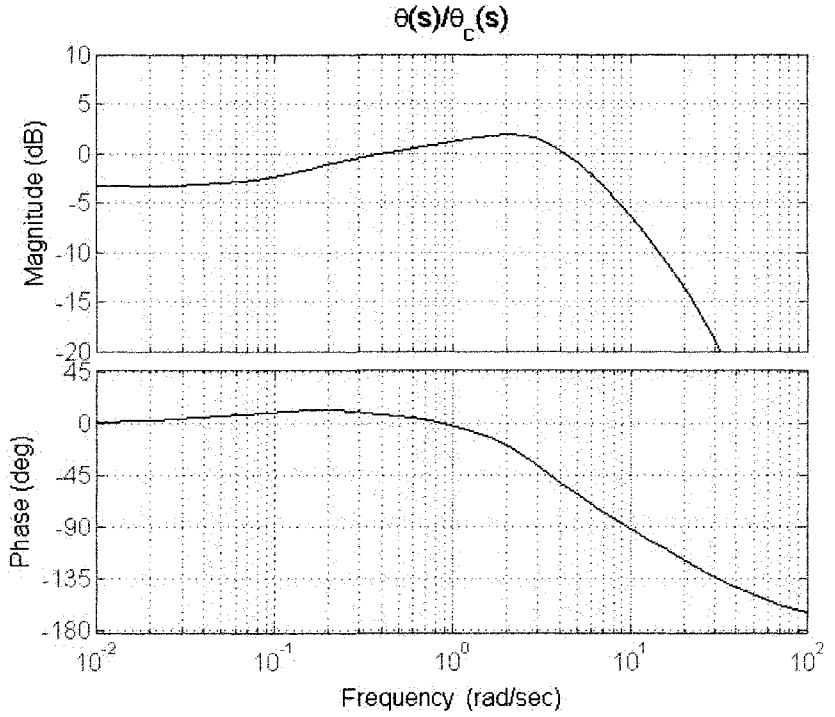


Fig. 5.3. Closed loop bode diagram after the first loop closure with elevator. This achieves handling qualities level 1 using the method described in Ref. [2.17].

The first pseudo-control, u_1', is thrust together with "elevator". This means that controller will be designed to pair the pitch channel $\delta_e = u_1'$ which would seem to be the most logical pairing without this pre-compensator. But also there is the additional thrust generated from this pseudo-control. If $u_2' = 0$, then $\delta_T = -1.08u_1'$ as well. This first pseudo-control will automatically move the elevator and the thrust at the same time.

The second pseudo-control just contributes to thrust. So if $u_1' = 0$, then $\delta_T = u_2'$ and $\delta_e = 0$.

5.1.2 Initial Decentralized Control Design

Now that the pre-compensator step is done, the next step is to design the actual compensator. The outputs and inputs will be paired as $\theta \rightarrow u_1'$ and $\dot{h} \rightarrow u_2'$.

Since the pitching dynamics of the aircraft are open-loop unstable and are readily stabilized by feeding back pitch angle into the elevator, this $y_1 \rightarrow u_1'$ loop will be closed first with a relatively high bandwidth.

$$\theta/u_1'(s) = \frac{(s+0.137)}{(s+0.8223)(s^2 - 0.6401s + 0.5326)} \tag{5.7}$$

Using MATLAB's "sisotool" function, the following compensator is designed to have a closed-loop bandwidth of about 6.5 rad/sec and a percent overshoot to a step less than 20. The resulting gain margin and phase margin was 20dB and 55 deg., respectively. Since this will be a first loop closure, integral action will not be pursued.

$$C_{u_1\theta}(s) = 135\frac{(s+1.5)}{(s+30)} \tag{5.8}$$

Despite the poor performance at low frequency, this simple compensator achieves handling qualities level 1 (which is good) using the method described in Ref. [2.20]. Of course, this good handling is somewhat a product of assuming that there is not a significant time delay with the actuators. A different actuator model would most likely give different handling qualities.

The next thing to do next is close the altitude rate loop, $\dot{h} \rightarrow u_2'$. The previous loop closure involved a pseudo-control input, $u_1' = \delta_e$, which was exactly a real control, δ_e. This second closure will involve a loop closure which is actually a pseudo-control, $u_2' = 1.08\delta_e + \delta_T$ as determined by the pre-compensator. Since the altitude rate is basically a first order decay, a simple PI compensator is sufficient to achieve a good loop-shape.

$$\left.\dot{h}/u_2'(s)\right|_{\theta \rightarrow u_1'} = \dot{h}/u_2'(s) = \frac{-1}{(s+0.137)} \tag{5.9}$$

Therefore, the obvious choice for a compensator is listed below in eq. 5.10 which achieves a bandwidth of 2 rad/sec.

$$C_{u_2\dot{h}}(s) = -2\frac{(s+0.1)}{s} \tag{5.10}$$

5.1.3 Outer Loop Closures

For hovering vehicles, it is common to control forward velocity by tipping the aircraft down gently to achieve some forward velocity. The outer loop PI controller in Fig. 5.4 was designed this way.

It is also obvious that if one wants to control altitude, one can just give commands to the altitude rate controller. Because of the natural integrator and slow bandwidth of this loop closure, one only needs to use a gain to design this loop.

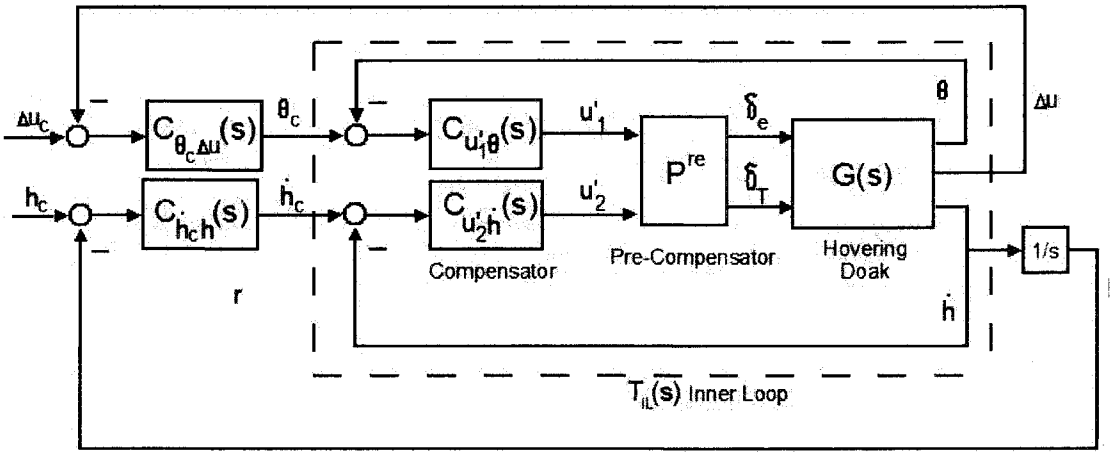


Fig. 5.4. Outer loop closures are achieved by controlling forward velocity, *u*, by giving commands to tip down and altitude, *h*, is controlled by giving commands to the altitude rate controller

5.1.3 Transient Response

Some simple simulations were run to display what was achieved with this control design, particularly by the decoupling pre-compensator. No actuator dynamics or non-linearities are used in this simulation. Only the state space listed at the beginning of this section is used and it was the exact same state space which was used to design the controller. These simulations were run purely to display what can be achieved under ideal conditions with a decoupling pre-compensator.

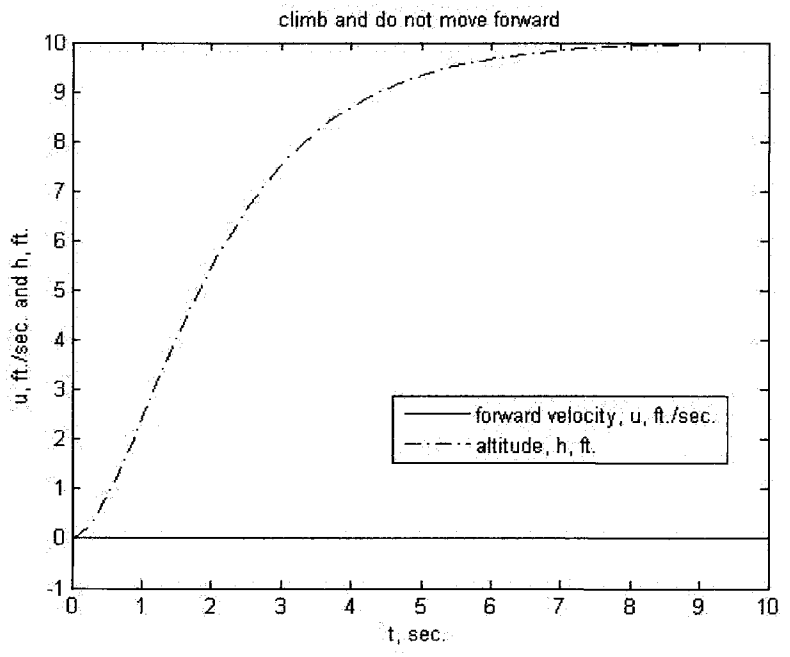


Fig. 5.5. When the command is given to climb and not to move forward, no forward velocity is generated.

In Fig. 5.5, one can see that the same behavior is exhibited when giving a command to climb; the other output does not change at all. In Fig. 5.6, one can see that the altitude output is not affected by the forward velocity command.

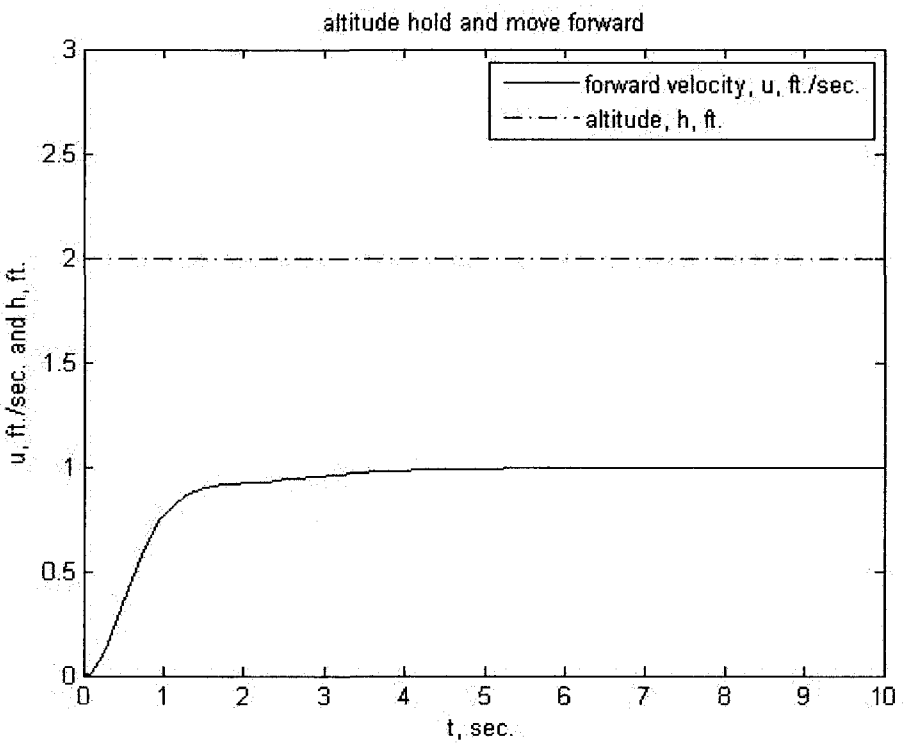


Fig. 5.6. Notice that there is absolutely no loss in altitude when given a step command to move forward at 1 ft./sec. Hence, there is no cross-coupling present in this response.

5.1.4 Control Design Summary

Table 5.1 is given below to summarize the design procedure and sequence of loop closures. Because the θ loop and $\dot{h}$ loops are decoupled, one could actually do step 2 before step 1, but this was not done in the design procedure.

Table 5.1. Summary of Control Design for VZ-4 Doak in hover

Design Step	Bandwidth	Handling Qualities	Notes
1 - Pre-compensator	NA	NA	Objective is to decouple or diagonalize plant dynamics
2 – attitude loop closure, $\theta \rightarrow u_1'$	6.5 rad./sec.	Level 1	Wanted a simple compensator, integral component omitted.
3 – altitude rate loop	2 rad./sec.	Unknown	The bandwidth of this closure still

closure, $\dot{h} \rightarrow u_2'$			could be fairly high. Time delays in thrust can create significant problems.
4 – forward velocity loop closure, $u \rightarrow \theta_c$	2 rad./sec.	Unknown	Rule of thumb - outer loop closure bandwidth $\leq$ 1/3 inner loop
5 – altitude loop closure, $h \rightarrow \dot{h}_c$	0.6 rad./sec.	Unknown	Rule of thumb - outer loop closure bandwidth $\leq$ 1/3 inner loop

5.1.5 Effect of Thrust Time Delay

During the design phase, actuator dynamics were not taken into account. Thrust is typically the slowest actuator on an aircraft and other actuators such as an elevator are typically much faster.

The first loop closure was made with a high bandwidth. It should be noted that the first pseudo control consists of both "elevator" and thrust. This causes concern because the first loop closure was made with a high bandwidth yet it also used the slow thrust input.

The approximation thrust will be a time delay in the range between 0.4 seconds to 1 second. Using the first order Pade approximation to a time delay, we will initially consider thrust to be generated with a 0.4 second delay and "elevator" to have a 0.1 second delay. Actuator dynamics are thus represented below.

$$Actuators(s) = \begin{bmatrix} -(s-20)/(s+20) & 0 \\ 0 & -(s-5)/(s+5) \end{bmatrix} \tag{5.11}$$

The first loop closure will now be reconsidered and we will also consider a parametric variation in the B_{22} value, noted ε. This would mean that thrust could also some pitching moment. The modified state space is shown in eq. 5.12 and it is shown in the Laplace domain in eq. 5.13. Note that in eq. 5.13 $G_{\theta\delta_T} = \varepsilon G_{\theta\delta_e}$.

$$A_\varepsilon = \begin{bmatrix} -0.137 & 0 & 0 & -32.2 \\ 0 & -0.137 & 0 & 0 \\ 0.0136 & 0 & -0.0452 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad B_\varepsilon = \begin{bmatrix} 0 & 0 \\ 1.08 & 1 \\ 1 & \varepsilon \\ 0 & 0 \end{bmatrix} \quad C_\varepsilon = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 \end{bmatrix} \tag{5.12}$$

$$G^{\varepsilon}(s)=\begin{bmatrix} G_{\theta\delta_e}(s) & \varepsilon G_{\theta\delta_e}(s) \\ G_{\dot{h}\delta_e}(s) & G_{\dot{h}\delta_T}(s) \end{bmatrix} \tag{5.13}$$

Now the precompensator and actuator dynamics are connected up in eq. 5.14.

$$G'^{\varepsilon}(s)=\begin{bmatrix} G_{\theta\delta_e}(s) & \varepsilon G_{\theta\delta_e}(s) \\ G_{\dot{h}\delta_e}(s) & G_{\dot{h}\delta_T}(s) \end{bmatrix}\begin{bmatrix} -(s-20)/(s+20) & 0 \\ 0 & -(s-5)/(s+5) \end{bmatrix}\begin{bmatrix} 1 & 0 \\ -1.08 & 1 \end{bmatrix} \tag{5.14}$$

Performing the necessary matrix multiplication from equation 5.14, we can look at the (1,1) entry of equation 5.14 in equation 5.15.

$$G'^{\varepsilon}_{\theta u_1}(s)=G_{\theta\delta_e}(s)\left(\frac{-(s-20)}{(s+20)}+(-1.08\varepsilon)\frac{(s-5)}{(s+5)}\right) \tag{5.15}$$

From eq. 5.15 we can see that the time delays create *additional* SISO zeros defined by the roots of eq. 5.16.

$$-(s-20)(s+5)-x(s-5)(s+20)=0 \tag{5.16}$$

In eq. 5.16, x is defined as $x \equiv (-1.08\varepsilon)$. One also notices that the roots of eq. 5.16, which are the additional SISO zeros created by the time delays, can also be represented by the feedback system shown in Fig. 5.7.

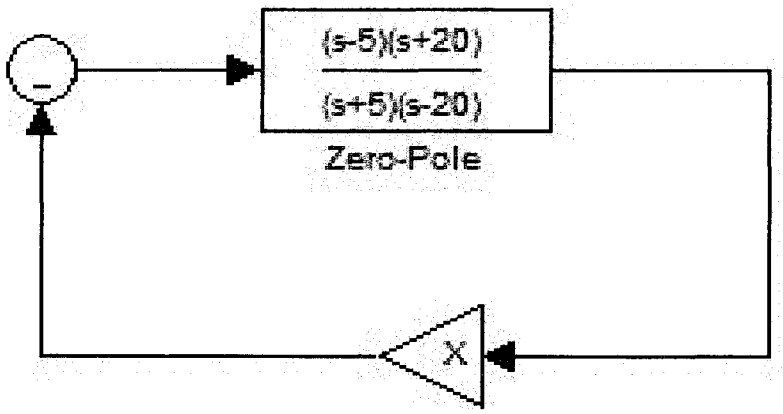


Fig. 5.7. To solve for the additional zeros created by the time delay, one can create a root locus based upon the above feedback system

The positive root locus to solve eq. 5.16 is shown in Fig. 5.8.

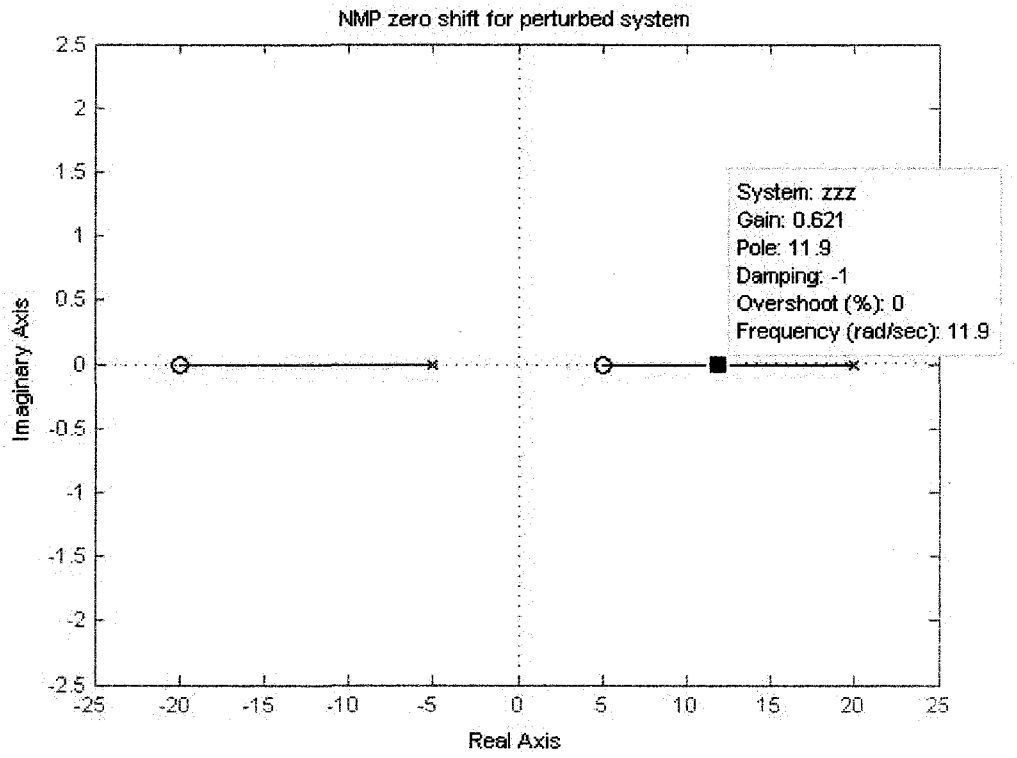


Fig. 5.8. From this locus plot, we can see that if x = 0.621, the NMP zero in the $G_{11}^{\prime\varepsilon}(s)$ transfer function will move from 20 to 11.9. This would correspond to an $\varepsilon = -0.575$

One can see that a fairly large gain of x = 0.621 cuts the unperturbed NMP in half. A more practical perturbation would most likely be something closer x = 0.05. Even though the $\theta \rightarrow u_1'$ loop was closed with a high bandwidth, the NMP zero mixing is a result of the pre-compensator and the $G_{\theta\delta_T}$ transfer function, which is expected to be small. A value of roughly x = 4 is large enough to shift the NMP to about s = 6.5, which would make the bandwidth of 6.5 rad/sec impossible with the existing pre-compensator.

It is the expectation that ε be small, it could be positive or negative. One also needs to look at the negative root locus of the system described in Fig. 5.7 which is displayed in Fig. 5.9.

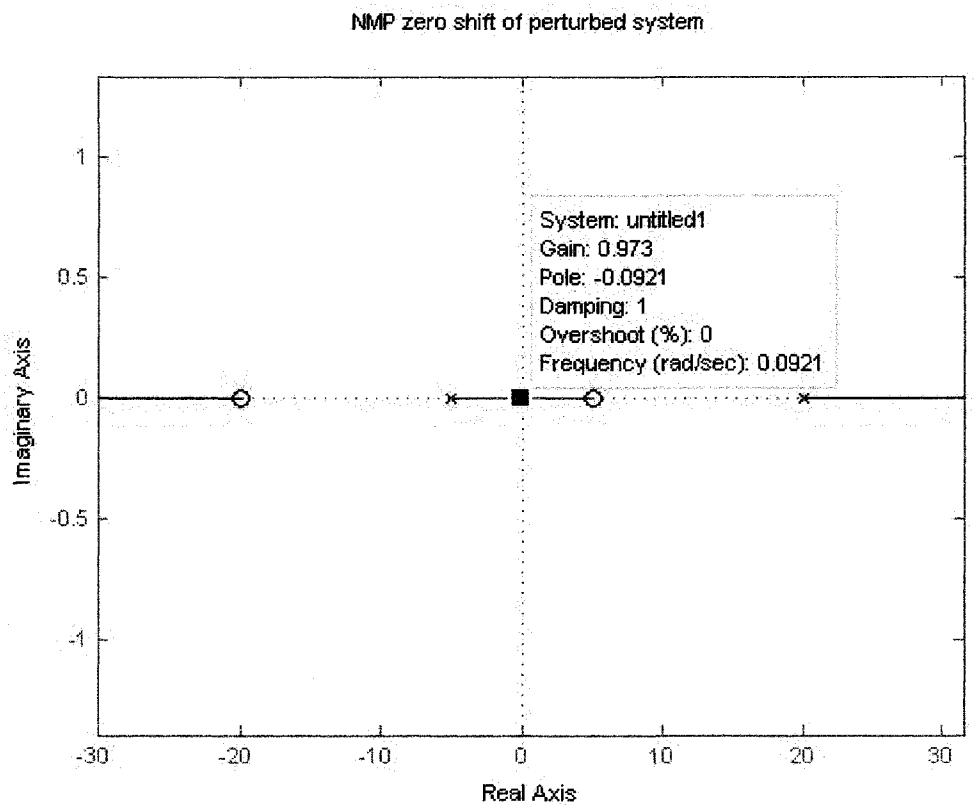


Fig. 5.9. From this negative locus plot, we can see that if x = -1, a very slow NMP zero will appear in the $G_{11}^{\prime\varepsilon}(s)$ transfer function. This would correspond to a value of $\varepsilon = 0.9259$. This would make the first loop closure impossible with the current pre-compensator.

This indicates that a value of $\varepsilon = 0.9259$ would make this loop closure impossible and the pre-compensator would need to be redesigned completely! But, this is unlikely, because a more reasonable value of $\varepsilon = 0.05$ or 0.1. Notice that only a large $\varepsilon = 0.9$ makes this pre-compensator infeasible, indicating that the first loop closure is fairly robust.

Table 5.2. Influence of Thrust Time delay on SISO NMP zero in 1st loop closure

Thrust Time Delay	"Elevator" Time Delay	Smallest positive perturbation ε to require redesign of pre-compensator	Smallest negative perturbation ε to make bandwidth of 6.5 rad./sec. impossible
0.4 sec.	0.1 sec.	0.9259	-3.61
0.6 sec.	0.1 sec.	0.9259	-1.46
0.8 sec.	0.1 sec.	0.9259	-1.06

Even though the first loop closure uses thrust and "elevator", the time delay only affects the first loop closure through the $G_{\theta\delta_T}$ transfer function which is nominally 0 for the hover condition. In other words, the thrust time delay has a very small affect on the first loop closure. This is not an immediately obvious result, but it is what results from the mathematics of the problem.

5.1.6 Effect of "Elevator" Time Delay and Redesign

The preceding sections used the assumption that the "elevator" on this VZ-4 Doak was as fast as a typical elevator on a CTOL (Conventional Take-Off and Landing) aircraft. It is entirely possible that this is not the case at all.

From Fig. 5.1, it looks the vane at the tail might redirect constant thrust. If this is the case, the vane should be fast and there should be no problem with time delay. This might unfortunately produce small forward/backward thrust.

It might also be that the vane is fixed downward and it is the magnitude of thrust coming out of that long column which is in fact the "elevator". If this is the case, then there is a huge problem with this control system, because thrust is so slow. We reconsider the actuators as follows in eq. 5.17.

$$Actuators(s) = \begin{bmatrix} -(s-5)/(s+5) & 0 \\ 0 & -(s-5)/(s+5) \end{bmatrix} \tag{5.17}$$

Now, it will become completely impossible under any circumstance to achieve the bandwidth of 6.5 rad./sec. Therefore, the compensator needs to be redesigned with lower bandwidths in mind. The pre-compensator does not need to change at all though.

The first loop closure is quite tough because it is unstable and non-minimum phase. Thus, there is only a small window of possible bandwidths in the range of about 1 rad./sec. to 5 rad./sec. The redesigned compensator for the first loop closure is shown below in eq. 5.18. From Fig. 5.10, one can see just how very small the margin for error is.

$$C_{\theta i_1'} = \frac{40.195(s+0.825)(s+0.671)(s+0.028)}{(s+0.123)(s+1.05)(s+30)} \tag{5.18}$$

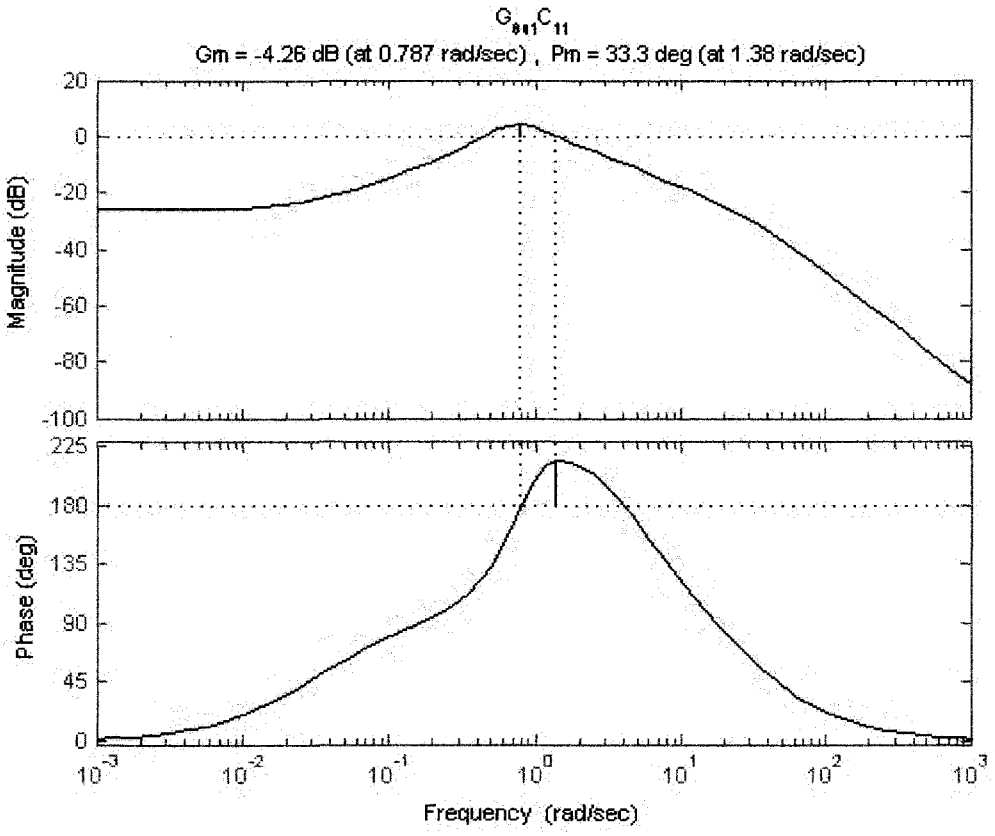


Fig. 5.10. Bode diagram with redesigned compensator, $G_{\theta u_1'}(s)C_{u_1'\theta}(s)$. Due to the slower "elevator" time delay and the unstable hover mode, the first loop closure is barely stable.

One should note that with this compensator, if the unstable hover mode becomes any faster, like greater than 1 rad./sec., then this loop closure will most likely be unstable.

Of course, because the sub-systems are decoupled, there is no need to close the first loop, $\theta \rightarrow u_1'$, before the second, but it was done anyway.

$$\left.\frac{\dot{h}}{u_2'}(s)\right|_{\theta \rightarrow u_1'} = \frac{\dot{h}}{u_2'}(s) = \frac{-1}{(s+0.137)} \tag{5.19}$$

Taking into account the time delay in thrust, we can simply scale back the second loop closure, so that it is closed at a bandwidth of 0.75 rad./sec

$$C_{u_2\dot{h}}(s) = -0.75\frac{(s+0.1)}{s} \tag{5.20}$$

Completing the $u \rightarrow \theta_c$ loop closure at a bandwidth of 0.5 rad./sec. unfortunately gives a complicated transfer function seen in eq. 5.21.

$$C_{u_1'\theta} = \frac{40.195(s+0.116)(s^2+0.6940s+0.5815)}{s(s+0.028)(s+20)(s+20)} \tag{5.21}$$

The very last loop closure to be redone is the altitude rate loop $h \rightarrow \dot{h}_c$, this is just done with a gain to achieve a bandwidth of 0.25 rad./sec.

$$C_{\dot{h}_c h} = 0.25 \tag{5.22}$$

Table 5.3. Summary of Redesign for slow "elevator" time delay

Design Step	Bandwidth	Handling Qualities	Notes
1 - Pre-compensator	NA	NA	No pole zero cancellation present
2 – attitude loop closure, $\theta \rightarrow u_1'$	2-3 rad./sec.	Level 2 by Ref. [1]	Very small gain and phase margin, difficult loop to close
3 – altitude rate loop closure, $\dot{h} \rightarrow u_2'$	0.75 rad./sec.	Unknown	
4 – forward velocity loop closure, $u \rightarrow \theta_c$	0.5 rad./sec.	Unknown	Rule of thumb - outer loop closure bandwidth $\leq$ 1/3 inner loop
5 – altitude loop closure, $h \rightarrow \dot{h}_c$	0.6 rad./sec.	Unknown	Rule of thumb - outer loop closure bandwidth $\leq$ 1/3 inner loop

5.2 Approximate Roll-Yaw Decoupling for the T-38

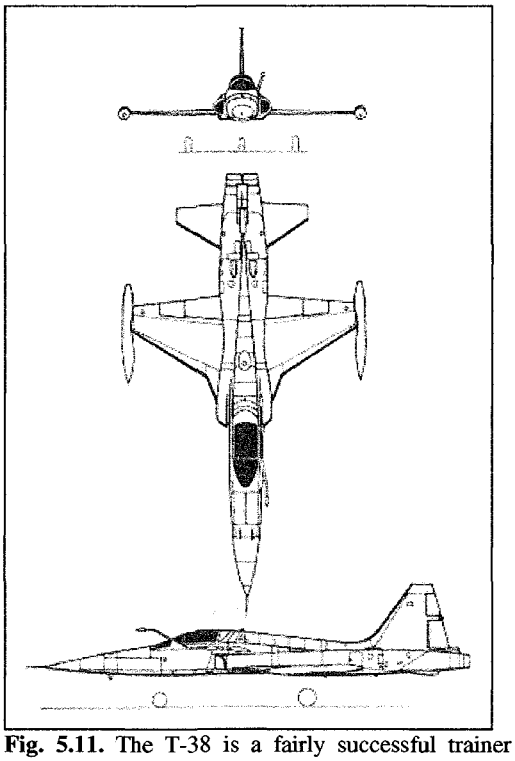


Fig. 5.11. The T-38 is a fairly successful trainer aircraft which has been used by the US air force since the 1960's

Fighter aircraft are typically required to roll without yawing. Frequently, they are required to simply execute a constant roll without yawing. Most fighter aircraft enter into sharply banked turns which are achieved by first rolling over very quickly and then producing a large amount of normal acceleration with the elevator. It is therefore of interest to reduce the amount of yaw generated during these roll maneuvers.

Just as in the previous section, a pre-compensator will be used in an attempt to minimize coupling between outputs. The particular pre-compensator will consist of an Aileron to Rudder Interconnect (ARI) and Rudder to Aileron Interconnect (RAI). The goal of designing this ARI is to make it possible to roll without yawing too much.

Before jumping into this case study, the idea of Feedforward Action in Decentralized Control

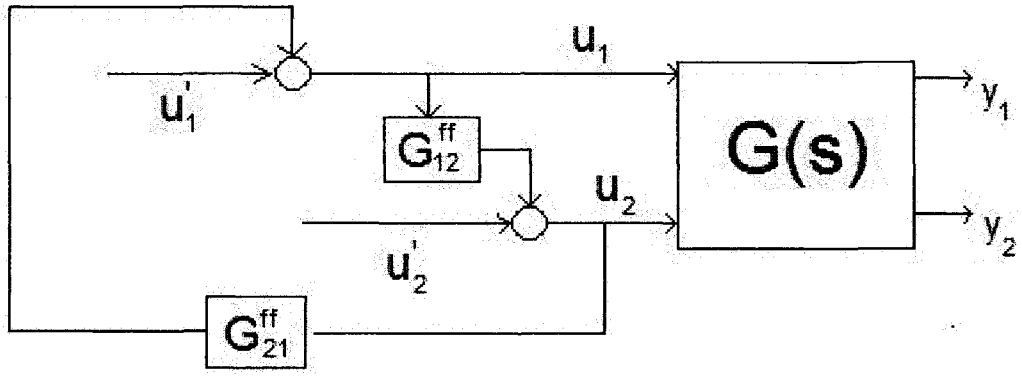


Fig. 5.12. Prior to designing any closed loop compensator, one attempts to approximately decouple the plant, $G(s)$, by using the interconnects of G_{12}^{ff} and G_{21}^{ff}.

[2.16] will be introduced.

With this in mind, one can design G_{21}^{ff} and G_{12}^{ff} in the manner described in eq. 5.23.

$$G_{22}(s)G_{12}^{ff}(s)+G_{21}(s)\approx 0 \tag{5.23}$$

Manipulation of eq. 5.23 leads to eq. 5.24.

$$G_{12}^{ff}(s)\approx -[G_{22}(s)]^{-1}G_{21}(s) \tag{5.24}$$

This can be further generalized past 2x2 systems [2.16].

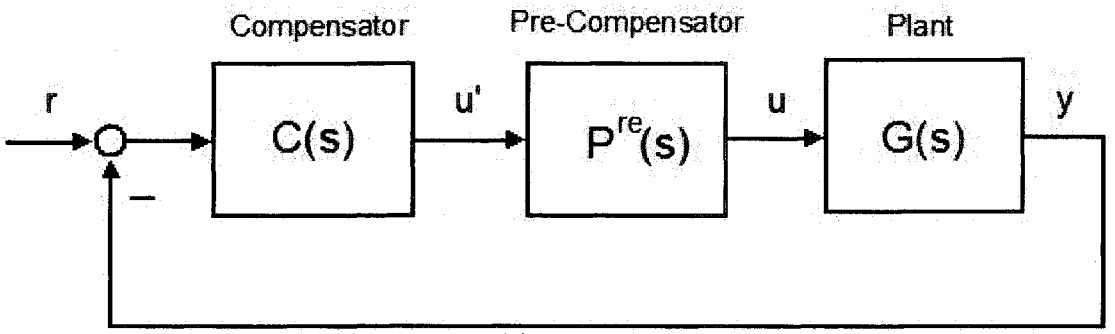


Fig. 5.13. Prior to designing the compensator $C(s)$, a pre-compensator , $P^{re}(s)$, is designed to approximately decouple or approximately diagonalize the plant. Then the compensator is designed with *u'* as the input.

$$G_{ji}^{ff}(s)\approx -[G_{ii}(s)]^{-1}G_{ij}(s) \tag{5.25}$$

The pre-compensator matrix $P^{re}(s)$ can be defined as follows in eq. 5.26. *It should be noted that it may look as though the entries in eq. 5.26 are mixed up, however, they are not. A decision was made to maintain the fairly awkward notation from Ref. [2.16].*

$$[P^{re}(s)]^{-1}=\begin{bmatrix} 1 & -G_{21}^{ff} & \cdots & -G_{m1}^{ff} \\ -G_{12}^{ff} & \ddots & -G_{ij}^{ff} & \vdots \\ \vdots & -G_{ji}^{ff} & \ddots & -G_{m(m-1)}^{ff} \\ -G_{1m}^{ff} & \cdots & -G_{(m-1)m}^{ff} & 1 \end{bmatrix} \tag{5.26}$$

The inverse of $P^{re}(s)$ is not symmetric unless the plant is symmetric. One should note that inverse dynamics are basically being used here, so one has to be wary of unstable pole-zero cancellations.

First, we will only consider one feed-forward need be considered which is $G_{12}^{ff} \equiv ARI(s)$.

Fig. 5.14 shows the T-38 regulatory Stability Augmentation System (SAS) from Ref. [2.19]. Notice that there is a gain, $K_{\delta a}$, which is more commonly referred to as an Aileron to Rudder Interconnect (ARI).

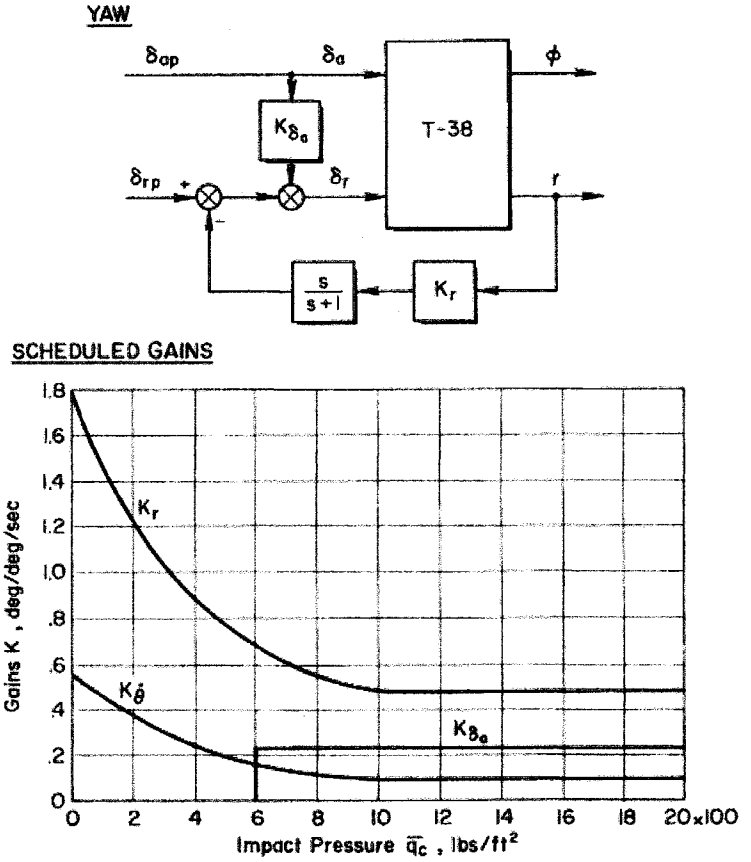


Fig. 5.14. Schematic of T-38 control system with ARI, $K_{\delta a}$, and washout feedback to the rudder. Taken from Ref. [2.19] and used with permission from NASA.

In eq. 5.27, the state space model for the lateral dynamics of an aircraft is given.

$$A=\begin{bmatrix} Y_v & 0 & -1 & g/U_o \cos\phi_o \\ L'_\beta & L'_p & L'_r & 0 \\ N'_\beta & N'_p & N'_r & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad B=\begin{bmatrix} Y^*_{\delta a} & Y^*_{\delta r} \\ L'_{\delta a} & L'_{\delta r} \\ N'_{\delta a} & N'_{\delta r} \\ 0 & 0 \end{bmatrix} \quad C=\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{5.27}$$

The states, outputs and inputs are listed below in eq. 5.28. The outputs are roll and yaw rate, respectively and the inputs are aileron and rudder, respectively.

$$y=\begin{bmatrix}\phi\\ r\end{bmatrix}\quad u=\begin{bmatrix}\delta_a\\ \delta_r\end{bmatrix}\quad x=\begin{bmatrix}\beta\\ p\\ r\\ \phi\end{bmatrix} \tag{5.28}$$

U_o	g	Y_v	L'_β	L'_p	L'_r
893	32.2	-0.737	-58.29	-4.316	1.242
N'_β	N'_p	N'_r	$Y^*_{\delta a}$	$Y^*_{\delta r}$	$L'_{\delta a}$
37.71	0.132	-0.736	0	0.1	27.75
$L'_{\delta r}$	$N'_{\delta a}$	$N'_{\delta r}$			
16.65	1.712	-11.01			

Table 5.4. Stability derivatives for the T-38 at Mach = 0.8 at Sea-Level

The stability derivatives for the T-38 are listed to the left for the flight condition of Mach = 0.8 at a sea level (SL) in Table 5.4. One should note that the sign of $L'_{\delta a}$ is positive. Positive aileron means left wing trailing edge down and right wing trailing edge up.

From eq. 5.24, one can calculate the ARI as shown in eq. 5.29.

$$ARI(s)=G^{ff}_{\partial r\delta a}(s)\approx -[G_{r\delta r}(s)]^{-1}G_{r\delta a}(s) \tag{5.29}$$

If $G_{r\delta r}(s)=N^r_{\delta r}(s)/\Delta(s)$ and $G_{r\delta a}(s)=N^r_{\delta a}(s)/\Delta(s)$, then an ARI can be calculated below.

$$ARI(s)=-N^r_{\delta a}(s)/N^r_{\delta r}(s) \tag{5.30}$$

Similarly, the RAI can be calculated in the same manner.

$$RAI(s)=G^{ff}_{\partial a\delta r}(s)\approx -[G_{\phi\delta a}(s)]^{-1}G_{\phi\delta r}(s) \tag{5.31}$$

$$RAI(s)=-N^\phi_{\delta r}(s)/N^\phi_{\delta a}(s) \tag{5.32}$$

With the data from table 5.4, we can calculate dynamic ARI(s). The bode plot of the dynamic ARI is shown in Fig. 5.15.

$$ARI(s)=\frac{0.1555(s^2+0.3592s+2.789)}{(s+0.3804)(s+0.0001356)}\quad\Rightarrow\quad ARI=0.1555 \tag{5.33}$$

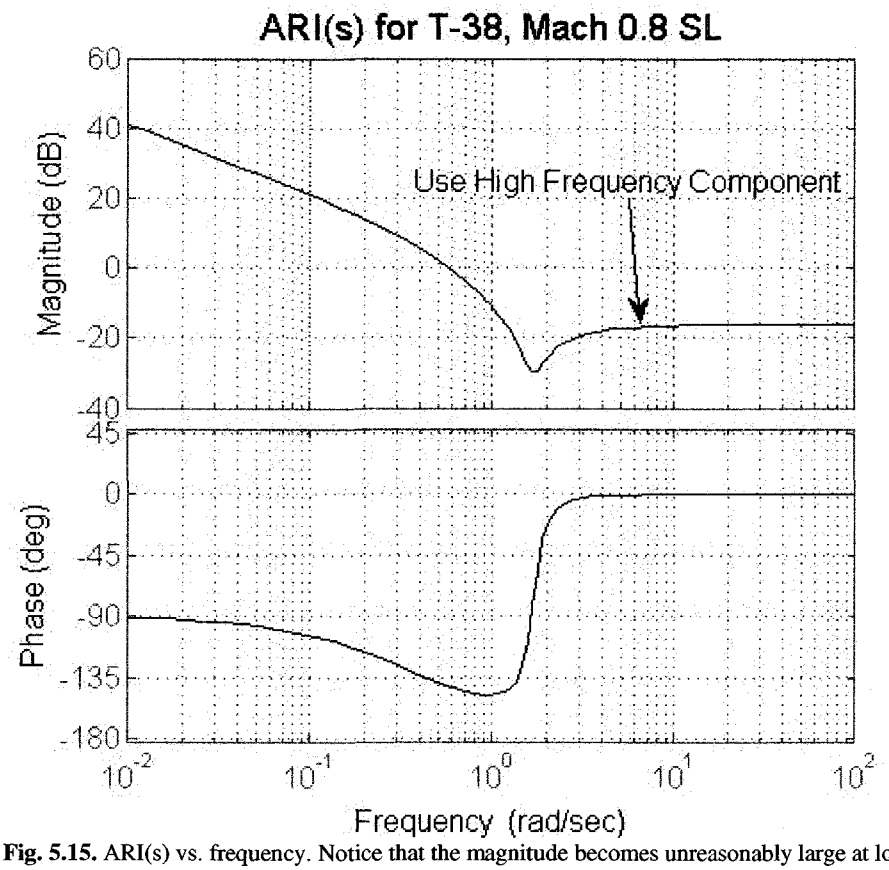


Fig. 5.15. ARI(s) vs. frequency. Notice that the magnitude becomes unreasonably large at low frequency.

From Fig. 5.15 and eq. 5.33, we can see that the ARI has a very large low frequency component due to the very slow pole which is almost an integrator.

Imagine that a pilot were to fly the aircraft with this ARI and leave the aileron in a small, yet non-zero position. After some time, a very large rudder position would result which would attempt to counteract a steady turn. This is not such a good idea, so we simply use the high frequency component of the ARI which is very close to the original design of ARI = 0.2, as seen in Fig. 5.14.

Next, we calculate the RAI from eq. 5.32 and produce the bode plot in Fig. 5.16. The redesigned RAI(s) is on the right side of the arrows in eq. 5.34.

$$RAI(s) = \frac{-0.6(s-0.6964)(s+0.9896)}{(s^2+1.549s+41.9)} \quad \Rightarrow \quad RAI(s) = \frac{-0.6(s-0.7)(s+1)}{(s^2+4s+42.44)} \tag{5.34}$$

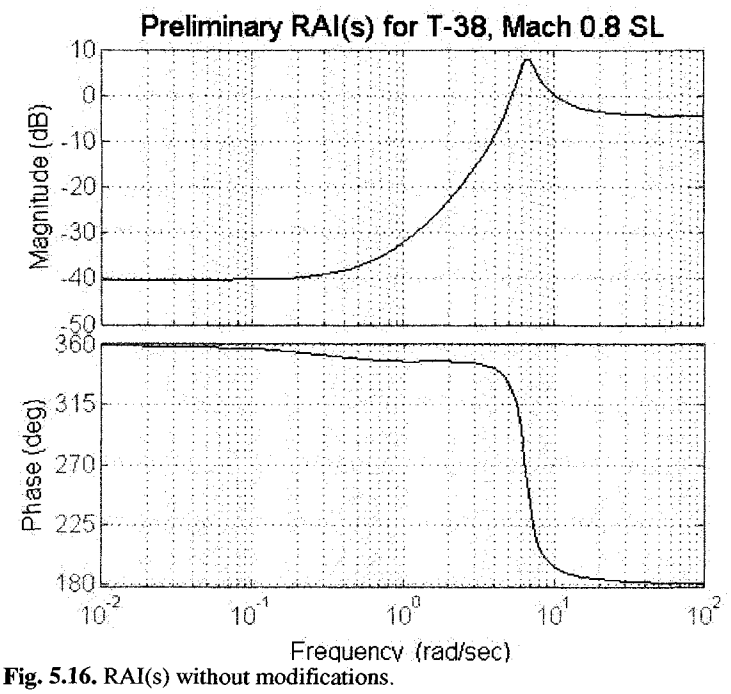


Fig. 5.16. RAI(s) without modifications.

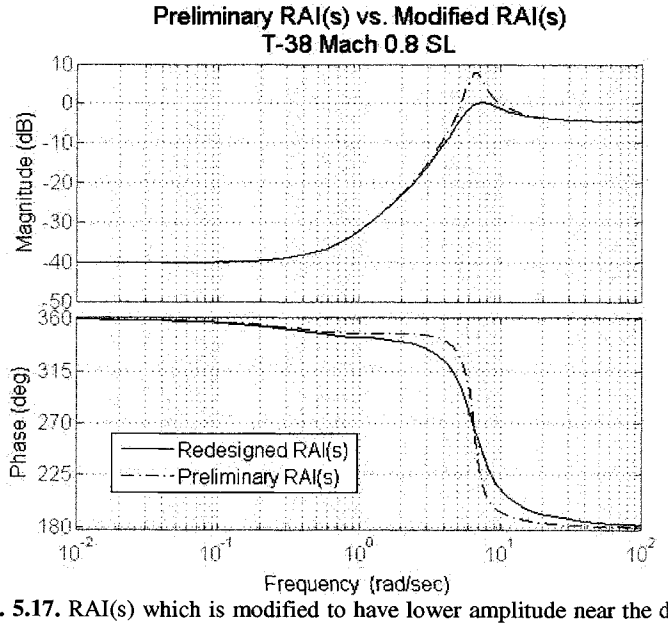


Fig. 5.17. RAI(s) which is modified to have lower amplitude near the dutch roll frequency.

From Fig. 5.16, we can see that at low frequency, the RAI(s) has a very small low frequency component magnitude of about 0.01. The small magnitude at low frequency is a very small issue. It is acceptable to use this low frequency component, which was not the case with the ARI.

Do notice that there is a large spike near the dutch roll frequency at 6 rad./sec. due to the under-damped oscillatory poles in the RAI(s). We choose to modify the RAI(s) so as to add some damping to the RAI(s). This makes sure we don't make pole-zero cancellations close the jw axis with our pre-compensator.

Generally, one should expect that the ARI to be less than 1, 2 or 3. An ARI of 10, 100 or 1000 is pretty unreasonable. Large ARI gains typically appear at low frequency because roll-yaw decoupling at low frequency is unreasonable.

From eq. 5.26, one can solve precisely for the pre-compensator which is listed in eq. 5.35.

$$P^{re}(s)=\begin{bmatrix} 1 & -RAI(s) \\ -ARI(s) & 1 \end{bmatrix}^{-1} \tag{5.35}$$

However, in this case, we will approximate the pre-compensator as shown in eq. 5.36.

$$P^{re}(s)\approx\begin{bmatrix} 1 & \dfrac{-0.6(s-0.7)(s+1)}{(s^2+4s+42.44)} \\ 0.1555 & 1 \end{bmatrix} \tag{5.36}$$

Closing the $\phi \rightarrow u_1'$ loop first with a bandwidth of about 5 rad./sec. gives the PID-like compensator listed in eq. 5.37.

$$C_{u_1'\phi}(s)=\frac{3.5(s+0.001)(s+4)}{s(s+20)} \tag{5.37}$$

The rudder to yaw rate transfer function after this loop closure is listed in eq. 5.38.

$$G'_{ru_2'}(s)\Big|_{\phi\rightarrow u_1'} \approx \frac{-12.03(s+0.39)}{(s^2+1.46s+37.35)} \tag{5.38}$$

We close the $r \rightarrow u_2'$ loop using the same washout design from Fig. 5.14 as in eq. 5.39.

$$C_{u_2'r}(s)=\frac{-0.5s}{(s+1)} \tag{5.39}$$

Table 5.5. Summary of Design Procedure for T-38

Design Step	Bandwidth	Handling Qualities	Notes
1 – Generate ARI(s)	NA	NA	Try to minimize cross-coupling along bandwidth of interest, Low frequency components using outputs of ϕ and r should be made to be small
2 – Check to make sure unstable pole-zero	NA	NA	If there are unstable (NMP) transmission zeros, be careful, do

cancellations are not made with G(s) and $P^{re}(s)$			not make unstable pole-zero cancellations of G(s) and $P^{re}(s)$. Check det(G(s) $P^{re}(s)$) and det(G(s))
3 – Roll rate loop closure, $\phi \rightarrow u_1'$	5 rad./sec.	Nice	Pilots enjoy decoupled response and high gain
4 –Yaw rate loop closure, $r \rightarrow u_2'$	High	Nice	

5.3 Dealing with Unstable Interconnects for the HL-10

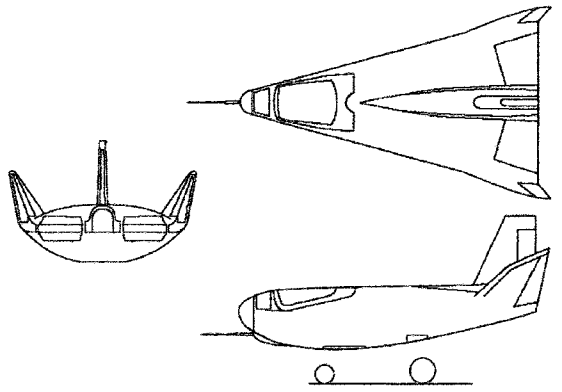


Fig. 5.18 The HL-10 generated lift from its body alone. It was used as a proof of concept for the space shuttle design.

In section 5.2, an active *RAI(s)* was used for roll-yaw decoupling. However, instances do occur where $N_{\delta r}^{r}(s)$ has SISO zeros which are unstable, which make for an unstable *ARI(s)*. This can occur even if the plant has stable transmission zeros.

Performing a fully diagonal decoupling procedure with an unstable *ARI(s)* using eq. 5.26 can sometimes introduce unstable poles in $P^{re}(s)$, which is undesirable. *There is no guarantee that using eq. 5.26 will work properly, even if the plant is open loop stable, and has stable*

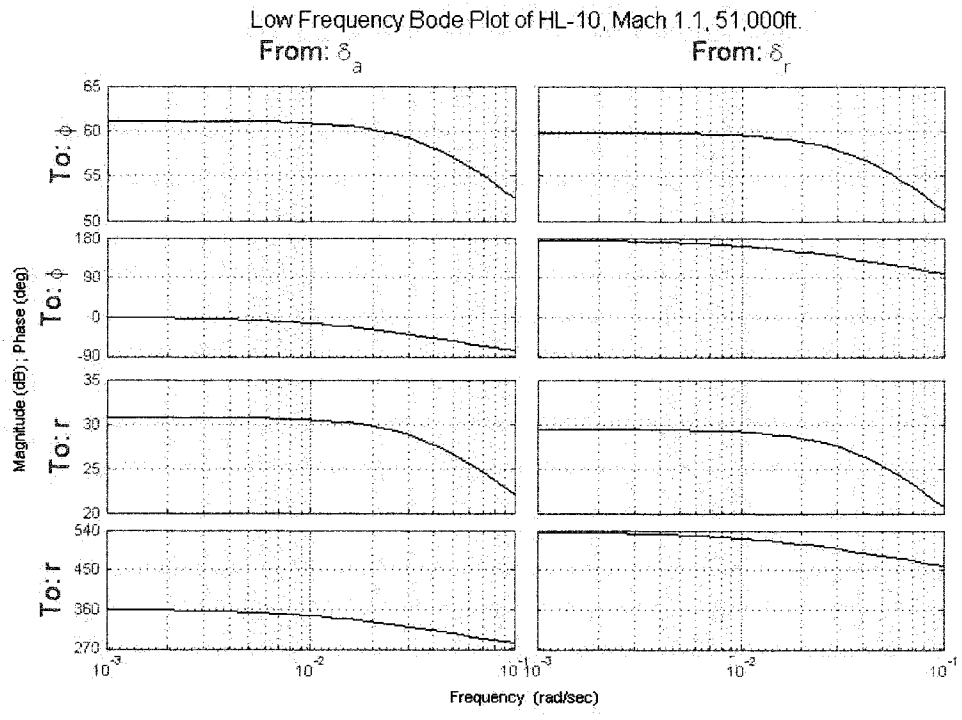


Fig. 5.19 Near the spiral mode, most aircraft form a singular transfer function with the outputs of ϕ and r. This is true for the HL-10 as well. A reasonable low frequency approximation for the HL-10 in this frequency range is given by $G(s) \approx \begin{bmatrix} 36.58 & -31.45 \\ 1.10 & -0.95 \end{bmatrix} \frac{1}{s+0.028}$

transmission zeros.

For the outputs of ϕ and r (or the outputs of p and r), an overwhelming majority of aircraft exhibit a largely singular behavior at low frequency. This singular behavior is shown in Fig. 5.19. Therefore, we will not invert with the low frequency content of $P^{re}(s)$. We will simply use the higher frequency content of $P^{re}(s)$ for this reason.

We first use eq. 5.25 to show that the *ARI(s)* is unstable. Fig. 5.20 shows the bode plot of the unstable *ARI(s)*.

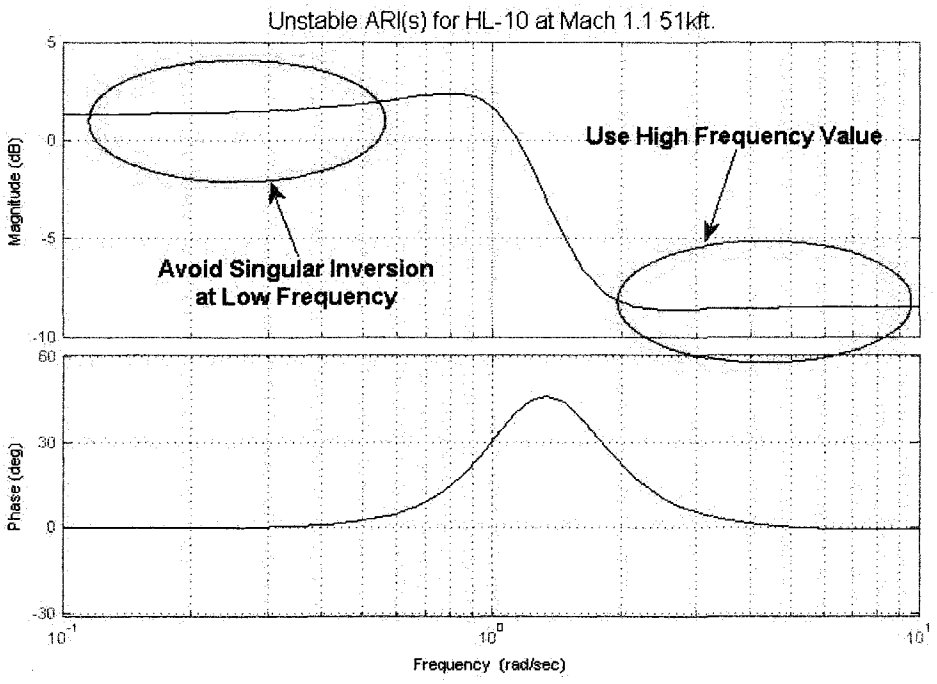


Fig. 5.20 The HL-10 has an unstable *ARI(s)* at Mach 1.1 and 51,000 ft. The *ARI(s)* can be calculated from eq. 5.29 and is given by $ARI(s)=\dfrac{0.376(s+1.99)(s^2-1.25s+2.58)}{(s+1.42)(s^2-0.78s+1.17)}$. *ARI = 0.38* will be used.

U_o	g	Y_v	L'_β	L'_p	L'_r
1064	32.2	-0.173	-69.3	-0.524	0.351
N'_β	N'_p	N'_r	$Y^*_{\delta a}$	$Y^*_{\delta r}$	$L'_{\delta a}$
11.0	0.00877	-0.351	-0.00590	0.0202	16.7
$L'_{\delta r}$	$N'_{\delta a}$	$N'_{\delta r}$			
6.50	1.83	-4.87			

Table 5.6. Stability derivatives for the HL-10 at Mach = 1.1 at 51kft.

From section 5.2, eq. 5.25, assuming there are no hidden modes in either $G_{p\delta r}(s)$ or $G_{p\delta a}(s)$, then a Rudder-to-Aileron Interconnect (RAI) can be calculated with the following in eq. 5.40.

$$RAI(s) = -N^{p}_{\delta r} / N^{p}_{\delta a} \tag{5.40}$$

Using eq. 5.25 with the constant values obtained from Fig.'s 5.20 and 5.21 gives the following pre-compensator in eq. 5.41.

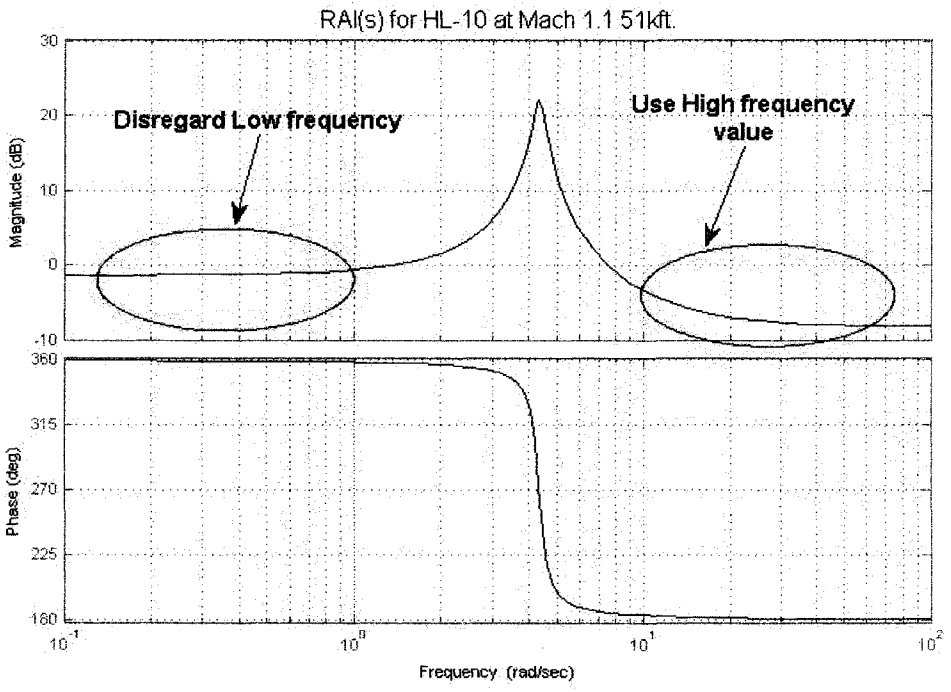


Fig. 5.21 The HL-10 has an unstable *RAI(s)* at Mach 1.1 and 51,000 ft. The *RAI(s)* can be calculated from eq. 5.34. We will use the approximate value of *RAI* = -0.4

$$\left[P^{re}\right]^{-1} = \begin{bmatrix} 1 & 0.4 \\ -0.38 & 1 \end{bmatrix}$$
$$P^{re} = \begin{bmatrix} 0.868 & -0.35 \\ 0.33 & 0.868 \end{bmatrix} \tag{5.41}$$

With the pre-compensator done, we can see the immediate benefit by comparing the root loci of $p \rightarrow u'_1$ with and without the pre-compensator. This is shown in Figs. 5.22 and 5.23.

From Fig. 5.22, we see the branches of the root locus move into the RHP. With the pre-compensator of eq. 5.41, we can see from Fig. 5.23 that the branches of the root locus do not move into the RHP, which is nice.

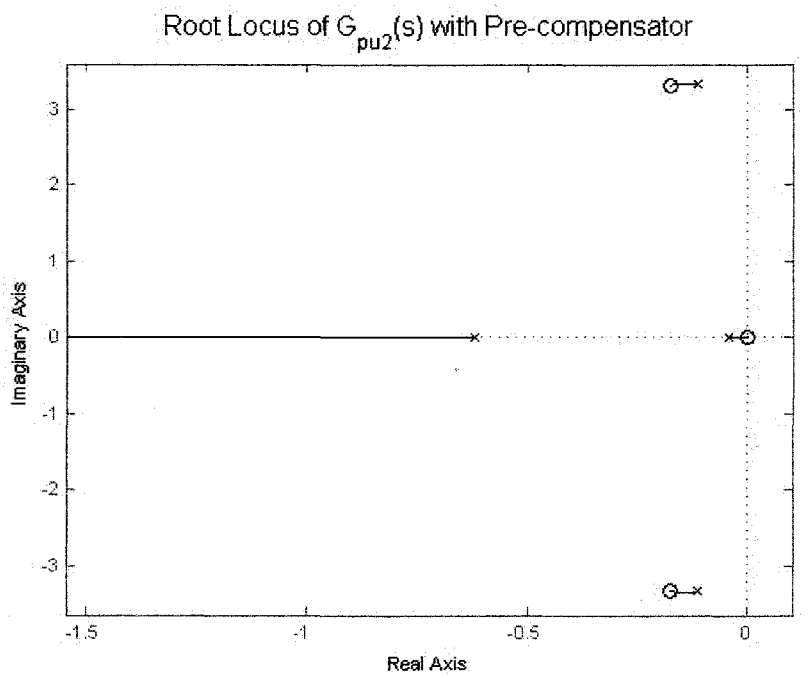


Fig. 5.23 With the pre-compensator of eq. 5.41, the branches of the root locus stay within the LHP, which is nice. This is a pairing of $p \rightarrow u'_1$ without any other loop closures.

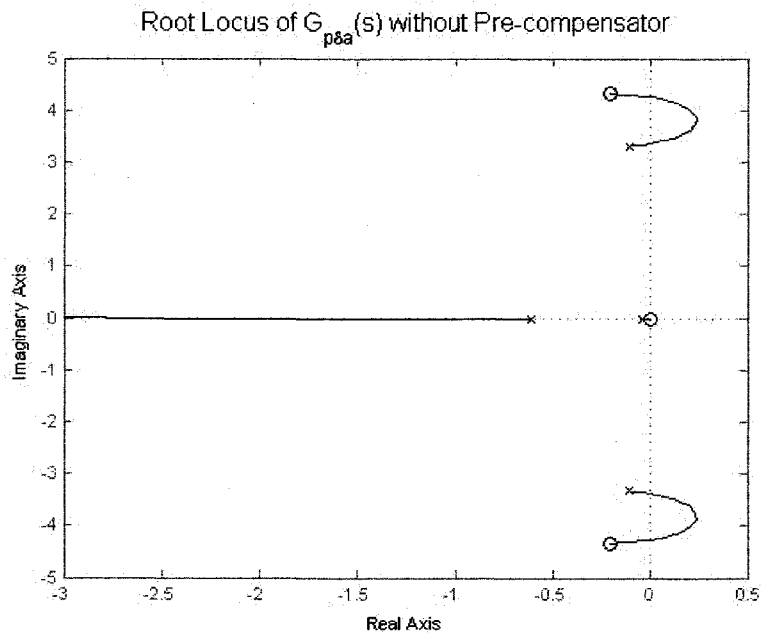


Fig. 5.24 The HL-10 has branches of the root locus which extend into the RHP without the pre-compensator of eq. 5.41. This is a pairing $p \rightarrow \delta_a$ without any other loop closures

There can sometimes be a stability benefit from using an *ARI* and/or a *RAI*, but one has to be careful not to make unstable pole-zero cancellations by blindly using the feedforward elements in eq. 5.25 and eq. 5.26.

Oddly enough, when it came to implementing a Stability Augmentation System (SAS) for the HL-10, no pre-compensator was used at all. From Ref. [5.2], it seems as though some high gains were used and cross over frequencies of up to 20 rad./sec. were exploited to suppress some strange limit cycles. Of course, this was possible because the structural modes of the HL-10 showed up above 24Hz (120 rad./sec.) and there were also some very fast servomechanisms.

5.4 Vehicles with Front and Rear Steering

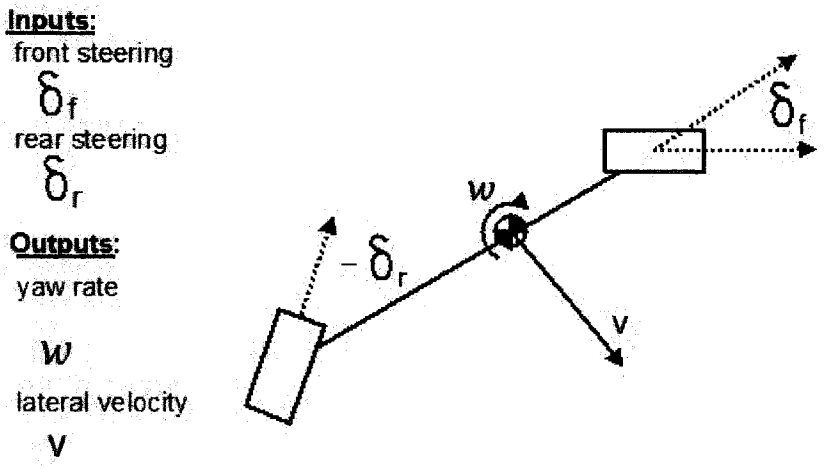


Fig. 5.25. The bicycle model of a vehicle from example 2.11. Inputs are front steering angle δ_f and rear steering angle, δ_r.

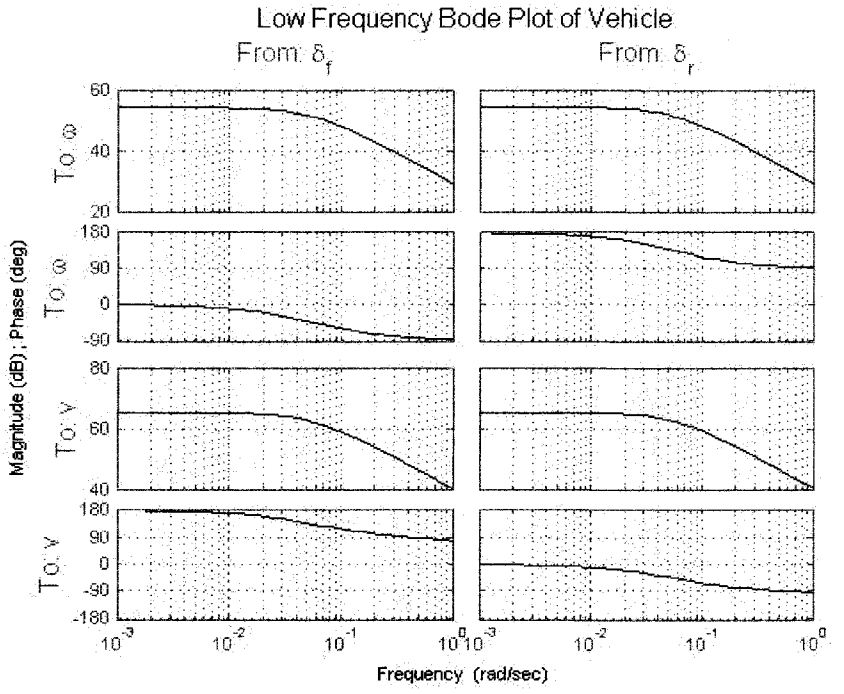


Fig. 5.26 The low frequency dynamics of a vehicle with front and rear steering are pretty close to being singular across a broad frequency range.

$$G(s) \approx \begin{bmatrix} 25.47 & -26.18 \\ -90.55 & 93.07 \end{bmatrix} \frac{1}{s+0.046}$$

be seen in Fig. 5.25, at really low frequency, things are approximately singular.

Low frequency singularity does not exist exclusively on aircraft with outputs p and r (or ϕ and r).

Low frequency singularity primarily manifests itself in systems where there are several actuators that do very nearly the same thing at low frequency. A car with both front and rear wheel steering is one of those things. These dynamics say that if you turn the rear wheels very slowly or if you turn the front wheels very slowly, you will turn in pretty much the same manner. As can

It is not a good idea to invert this singular behavior. Therefore, we will pursue the same strategy of section 5.4, only we will apply it to a car this time. We will use equations 5.23 through 5.26 and arrive at the following interconnects.

From Fig. 5.27, one can see that we will be using the interconnects of $G^{ff}_{\delta f \delta r}(s) \approx -1$, which will 180 degrees out of phase with the low frequency value. In the case of both this vehicle and the HL-10, this does not cause instability. Vehicle dynamics with data from Ref. [5.3] are shown below in eq. 5.42. There is a transmission zero at 0.

$$G(s) = \begin{bmatrix} 28.8(s+5.124) & -29.28(s+5.122) \\ 46.875(s-11.16) & 46.875(s+11.45) \end{bmatrix} \frac{1}{(s+0.046)(s+5.122)} \tag{5.42}$$

The first output of the transfer function in eq. 5.42 is yaw rate, ω, and the second output is lateral velocity, v. The first input is front steering, δ_f, and the second input is rear steering δ_r.

$$\begin{aligned} G^{ff}_{\delta f \delta r}(s) &= -(G_{v\delta r}(s))^{-1} G_{v\delta f}(s) = \frac{-(s-11.16)}{(s+11.45)} \\ G^{ff}_{\delta r \delta f}(s) &= -(G_{\omega \delta f}(s))^{-1} G_{\omega \delta r}(s) = 1.0167 \frac{(s+5.122)}{(s+5.124)} \approx 1 \end{aligned} \tag{5.43}$$

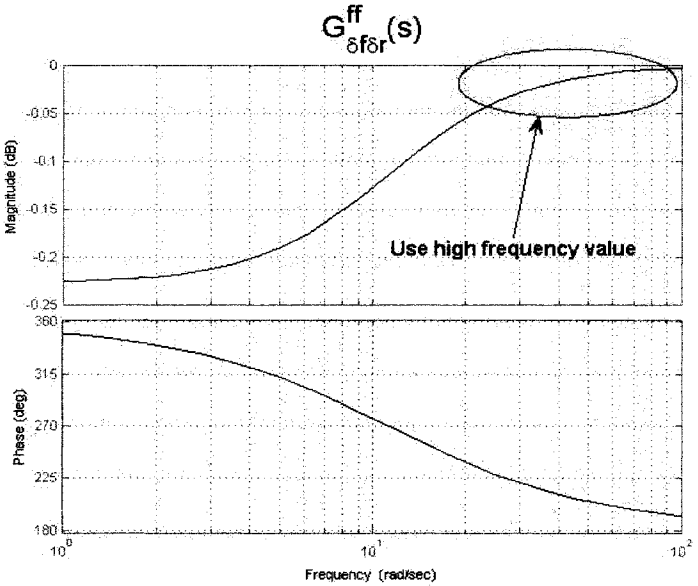


Fig. 5.27 The low frequency dynamics of a vehicle with front and rear steering are pretty close to being singular across at low frequency, so we will once again disregard the low frequency value of $G^{ff}_{\delta f \delta r}(s)$. $G^{ff}_{\delta f \delta r}(s) \approx -1$ will be used for pre-compensator design.

Hence, we propose the following pre-compensator using eq. 5.43 through 5.44.

$$\begin{aligned} \left[P^{re}\right]^{-1} &= \begin{bmatrix} 1 & -(1) \\ -(-1) & 1 \end{bmatrix} \\ P^{re} &= \begin{bmatrix} 0.5 & 0.5 \\ -0.5 & 0.5 \end{bmatrix} \end{aligned} \tag{5.44}$$

We can see that the first pseudo-control, defined by the first column of P^{re}, will consist of positive front steering and negative rear steering. The second pseudo-control, defined by the second

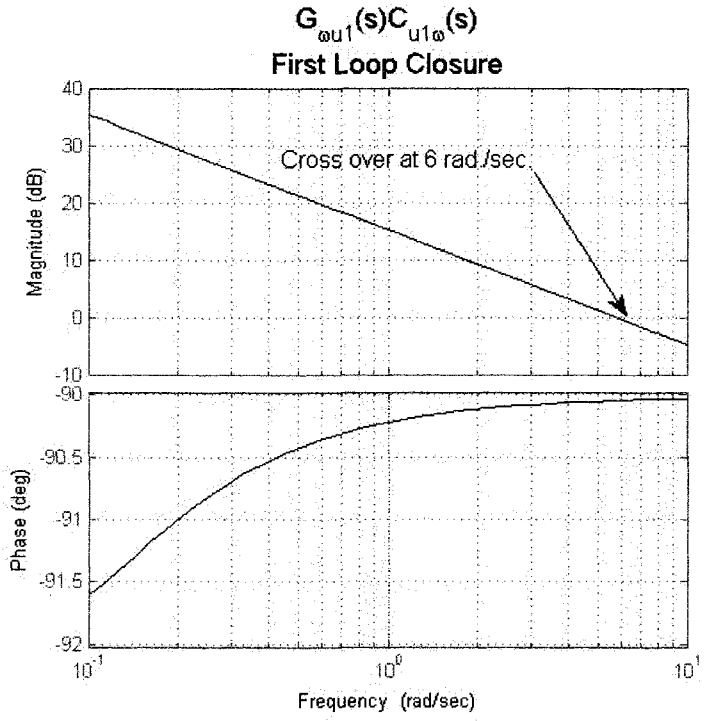


Fig. 5.28 First loop closure where yaw rate is paired with the first pseudo-control, $\omega \rightarrow u'_1$. An aggressive cross over of 6 rad./sec. was chosen.

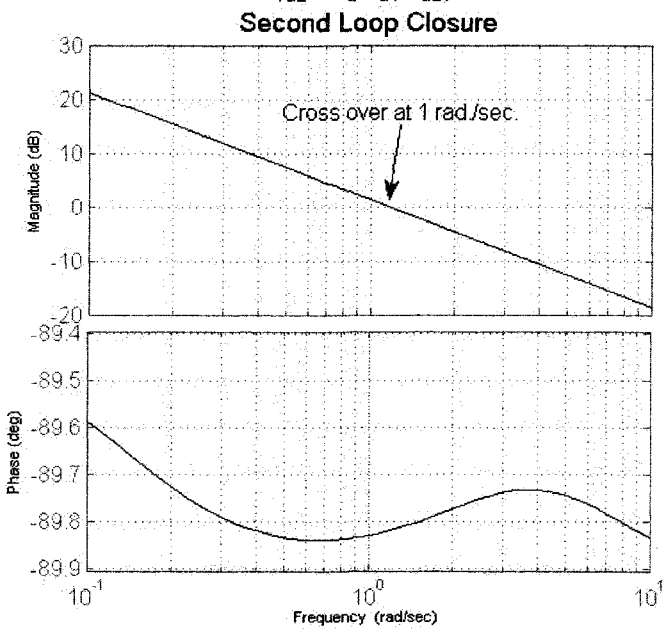


Fig. 5.29 The second loop closure is more gentle and has a cross over frequency of 1 rad./sec.

column of P^{re}, will consist of positive front steering and positive rear steering.

We will decide to close the first loop of $\omega \rightarrow u'_1$ with a bandwidth of about 6 rad./sec. Excluding actuator/sensor dynamics, this gives a nearly perfect $1/s$-like loop shape, shown in Fig. 5.27.

After closing the first loop, we move on to closing the second loop which is paired as $v \rightarrow u'_2$, and we decide to cross over gently at 1 rad./sec. This choice was made because turning maneuvers which involve ω are likely to be more important than maneuvers that require direct lateral velocity, v. Most people are accustomed to simply turning their vehicle.

5.5 "Staggered" Decoupling Procedure for F-104 Lateral system

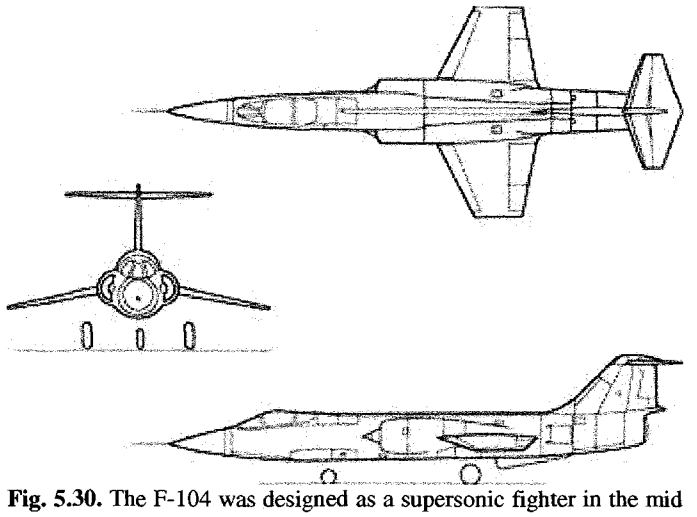


Fig. 5.30. The F-104 was designed as a supersonic fighter in the mid 1950's

It is not always true that one needs an ARI for an aircraft or that one has to use yaw rate, *r*, to calculate it. In some cases, one may want to design a Rudder-to-Aileron Interconnect (RAI).

It is also not universally true that one has to design all elements of a decoupling precompensator before closing any loops. One can design a single decoupling feedforward, close a loop, then look at a different loop closure with a different decoupling feedforward. The following example of the F-104 illustrates how such a design procedure could be carried out.

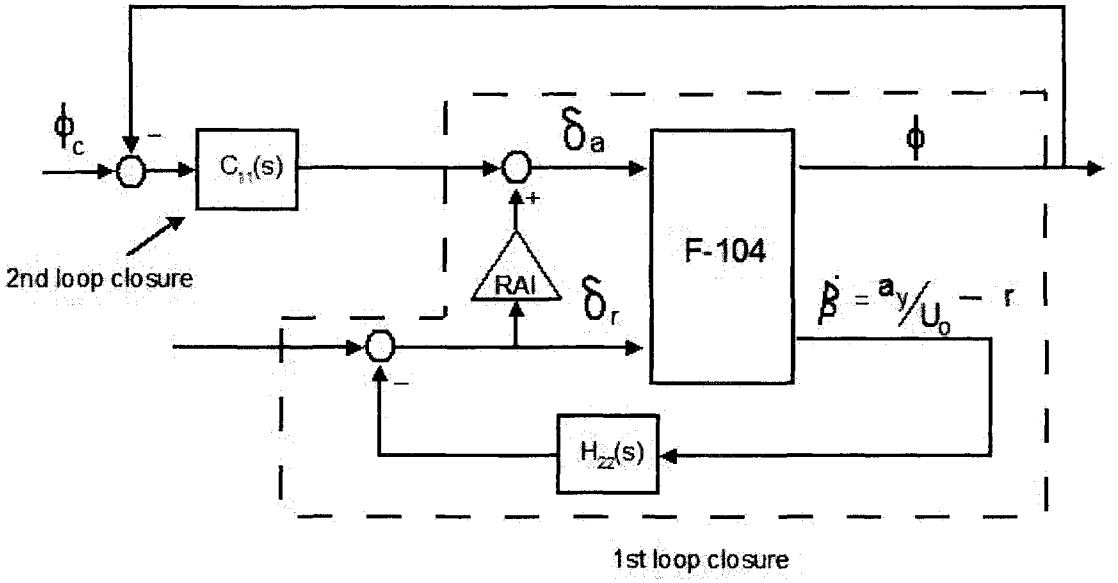


Fig. 5.31 Initial design strategy for the F-104. We will design a regulator to damp out an unstable dutch roll mode. Then we will design a bank angle compensator.

The F-104 has a very odd behavior in that $\left|L'_{\delta a}/L'_{\delta r}\right| \approx 1$ across most flight conditions [2.19].

This means that the rudder causes just as much rolling moment as the aileron, which is truly strange. The state space for the F-104 at Mach 0.257, SL is listed below.

U_o	g	Y_v	L'_β	L'_p	L'_r
287	32.2	-0.178	-20.9	-1.38	1.16
N'_β	N'_p	N'_r	$Y^*_{\delta a}$	$Y^*_{\delta r}$	$L'_{\delta a}$
2.68	-0.00993	-0.157	0	0.0317	4.76
$L'_{\delta r}$	$N'_{\delta a}$	$N'_{\delta r}$	ϕ_o		
5.35	0.266	-0.923	0		

Table 5.7. Stability derivatives for the F-104 at Mach = 0.257 at SL

$$A=\begin{bmatrix} Y_v & 0 & -1 & g/U_o\cos\phi_o \\ L'_\beta & L'_p & L'_r & 0 \\ N'_\beta & N'_p & N'_r & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad B=\begin{bmatrix} Y^*_{\delta a} & Y^*_{\delta r} \\ L'_{\delta a} & L'_{\delta r} \\ N'_{\delta a} & N'_{\delta r} \\ 0 & 0 \end{bmatrix} \quad C=\begin{bmatrix} 0 & 0 & 0 & 1 \\ Y_v & 0 & -1 & g/U_o\cos\phi_o \end{bmatrix}$$

$$D=\begin{bmatrix} 0 & 0 \\ Y^*_{\delta a} & Y^*_{\delta r} \end{bmatrix} \quad y=\begin{bmatrix}\phi & \dot\beta\end{bmatrix}^T=\begin{bmatrix}\phi & a_y/U_o - r\end{bmatrix}^T \quad u=[\delta_a \quad \delta_r]^T$$

5.5.1 Initial RAI Design

Figure 5.31 shows the initial design strategy we will employ. We will first get the RAI using the method in section 5.2, but we will only use RAI as a constant gain.

We will be looking at the output of bank angle ϕ as a control variable and feeding back $\dot\beta$ simply to damp out an unstable dutch roll mode.

With the RAI as a constant, we will then feed back $\dot\beta$ into the psuedocontrol of $u_2' = \delta_r + RAI\delta_a$. From section 5.2, eq. 5.25, the RAI can be designed as follows in eq. 5.45.

$$RAI(s) = G^{ff}_{\partial r\delta a}(s) \approx -\left[G_{\phi\delta a}(s)\right]^{-1} G_{\phi\delta r}(s) \tag{5.45}$$

Assuming there are no hidden modes in either $G_{\phi\delta r}(s)$ or $G_{\phi\delta a}(s)$, then an *RAI* can be calculated with the following in eq. 5.46.

$$RAI(s) = -N^\phi_{\delta r}/N^\phi_{\delta a} \tag{5.46}$$

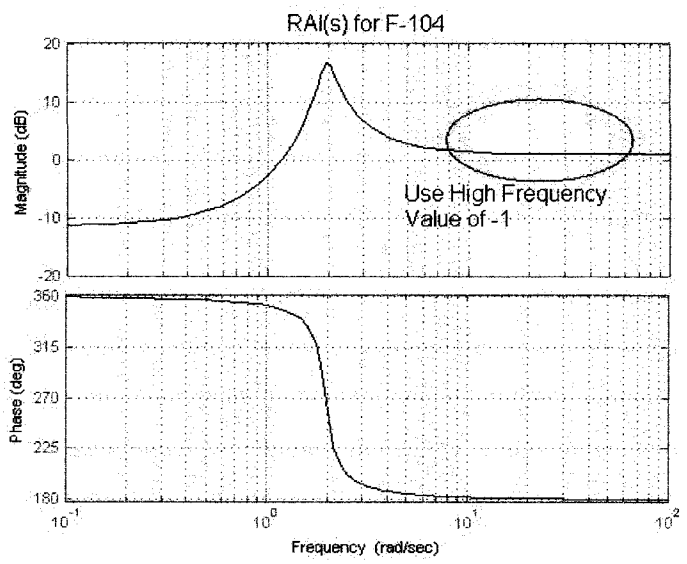


Fig. 5.32. *RAI(s)* for the F-104 at Mach 0.257. With the exact RAI(s) value of $RAI(s) = \dfrac{-1.1239(s+0.9722)(s-0.9612)}{s^2+0.3998s+3.887}$, unstable pole-zero cancellations would be made when trying to close the $\dot{\beta} \rightarrow u_2^{'}$ loop. We will use the high frequency content of the RAI, which is -1.

$$P^{re}_{\dot{\beta}\rightarrow u_2^{'}} = \begin{bmatrix} RAI \\ 1 \end{bmatrix} \tag{5.47}$$

Figure 5.32 shows the bode plot of the *RAI(s)* at this particular flight condition. Notice that *RAI(s)* has a small value at low frequency, a large peak around the dutch roll frequency and then tapers off to a value of approximately -1.

If one used the exact *RAI(s)* across all frequencies, it would make unstable pole zero cancellations for the $\dot{\beta} \rightarrow u_2^{'}$ loop closure. Obviously, this undesirable, since the whole purpose of the $\dot{\beta} \rightarrow u_2^{'}$ loop closure was to damp out an unstable dutch roll mode! Thus, we will use the high-frequency content of *RAI(s)* which is just -1, and design a regulator in a washout fashion. Of course, using the value of -1 is not so great for low frequency behavior because *RAI(0)=0.27* and the two are 180 degrees out of phase. But this should not matter too much since we will be pursuing a washout-like regulator as seen in Fig. 5.33.

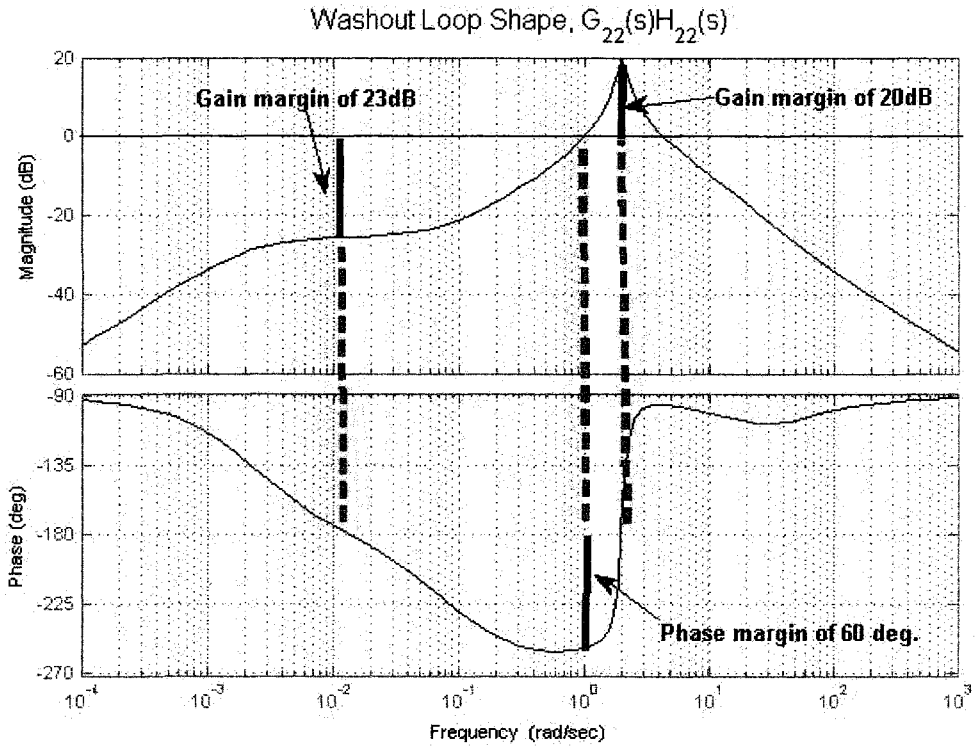


Fig. 5.33. Washout-like loop shape for design with $RAI=-1$, $H_{22}(s)=3\dfrac{20}{(s+20)}$ and $G_{22}(s)=G_{\hat{\beta}u'_2}(s)\approx\dfrac{.032s(s-.077)(s+1.57)(s+37.6)}{(s+2.05)(s+0.002)(s^2-0.34+3.98)}$.

The next thing to do is to design the *ARI(s)* and close the bank angle loop. However, we must keep in mind that that this will not be the original *ARI(s)* of the open loop system. This will be different thing, $ARI(s)_{\hat{\beta}\to u'_2}$, that is the *ARI(s)* will be calculated with the first loop closed. It is unfortunate that $ARI(s)_{\hat{\beta}\to u'_2}$ is unstable. Note that it is unstable at very low frequency. Also, recall that *RAI(0)=0.27* and the *RAI = -1* were 180 degrees out of phase from Fig. 5.32. This is most likely the cause of the problem.

$$ARI(s)_{\hat{\beta}\to u'_2}\approx\frac{8.39(s-2.58)(s+0.173)}{(s+37.55)(s+1.57)(s-0.077)} \tag{5.48}$$

We will have to finish off this design using a simple PID design using ARI = 0.

$$C_{11}(s)\big|_{ARI=0}=C_{\phi\delta a}(s)=\frac{24(s+0.35)(s+0.0027)}{s(s+20)} \tag{5.49}$$

5.5.2 RAI Redesign

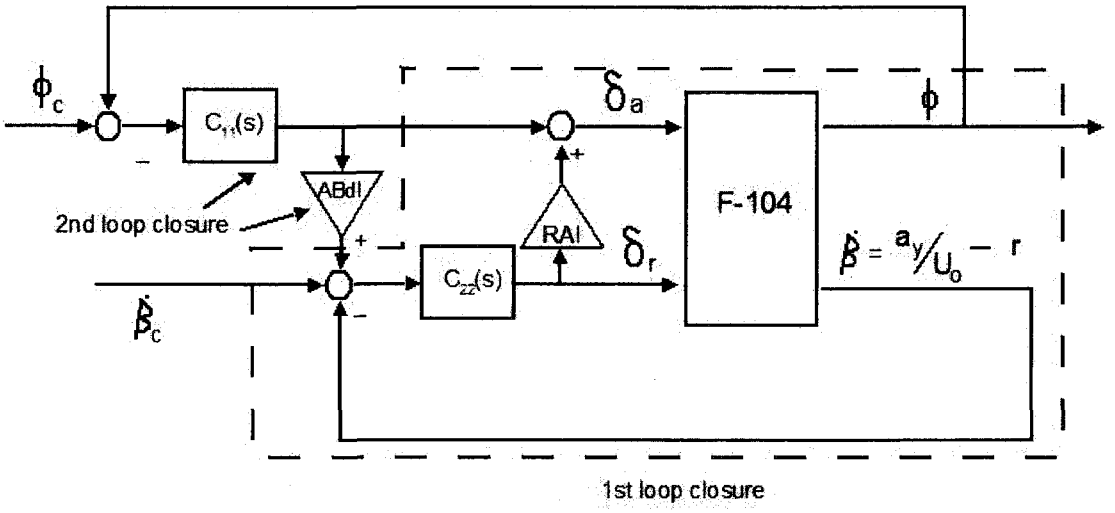


Fig. 5.34 Redesign strategy with "staggered" interconnects. We will going for integral action this time with the $\dot{\beta}$ loop after we redesign the *RAI(s)* to be active. Next, we will design a Aileron to Beta-dot Interconnect (ABdI).

As mentioned earlier, that *RAI(0)=0.27* and the *RAI = -1* are 180 degrees out of phase as can be seen from Fig. 5.32. Also, recall that if we used the exact RAI calculated directly from eq. 5.45 or 5.46, we would make unstable pole-zero cancellations in the first loop closure. Thus, we will use an active RAI, only we will do it "wrong" to avoid the unstable pole-zero cancellations. Fig. 5.35 shows the redesigned RAI to avoid this unstable pole-zero cancellation. Basically, all that was done was to add damping to the denominator.

With this active *RAI(s)* in mind, we will do the $\dot{\beta} \rightarrow u_2'$ loop closure this time and we will go for integral action. Figure 5.36 shows the loop shape for the $\dot{\beta} \rightarrow u_2'$ loop closure. Since it is fairly strange due to the unstable dutch roll mode, stability can be verified via a nyquist plot.

The next thing to do is to design the *ABdI(s)*. To do this, we simply use eq. 5.25 after closing the loop $\dot{\beta} \rightarrow u_2'$ with the "feedback( )" subroutine.

The exact *ABdI(s)* is unfortunately 9th order. Therefore, a model order reduction was done to omit some states and *ABdI(s)* was reduced to 5th order.

$$ABdI(s)\Big|_{\dot{\beta}\rightarrow u_2'} \approx \frac{-0.0014s(s-13.16)(s-2.751)(s+0.5174)}{(s+0.2011)(s+0.0944)(s^2+0.9s+4.864)(s+10)} \tag{5.50}$$

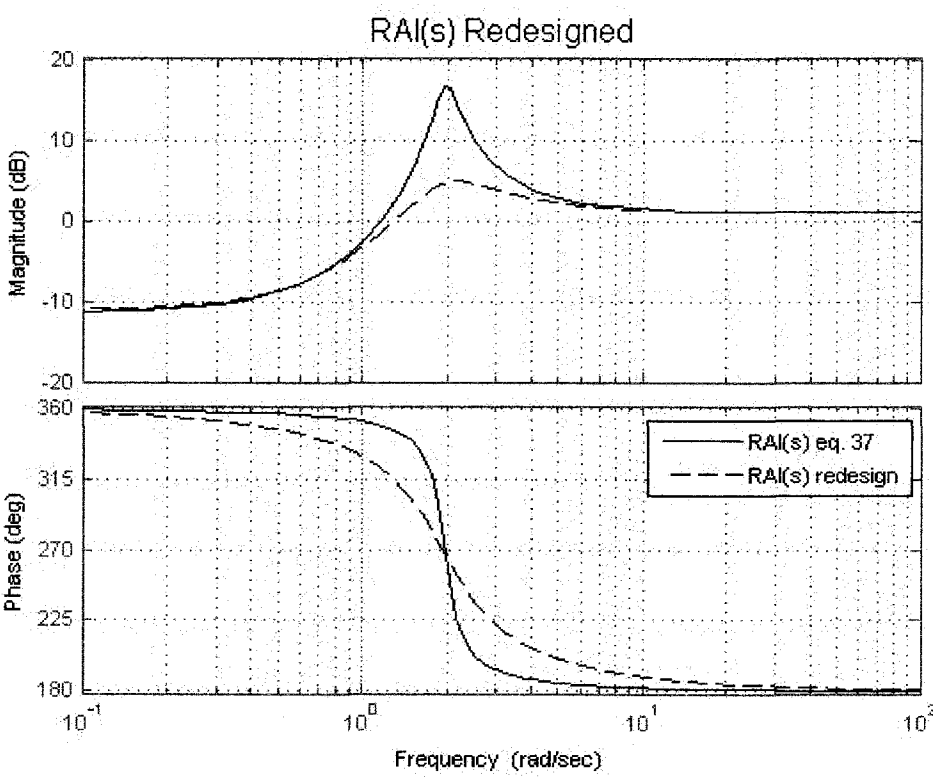


Fig. 5.35. The *RAI* was redesigned to be $RAI(s) \approx \dfrac{-1.124(s+0.972)(s-0.961)}{(s^2+1.6s+3.756)}$ to avoid unstable pole-zero cancellations for the $\dot{\beta}$ loop closure.

With the *ABdI(s)* complete, one can close off the final loop with $u'_1 = \delta_a + ABdI(s)\dot{\beta}_c$. This loop shape is rather odd looking for a bank compensator, but we will close this $\phi \rightarrow u'_1$ loop with a

typical PID compensator nonetheless.

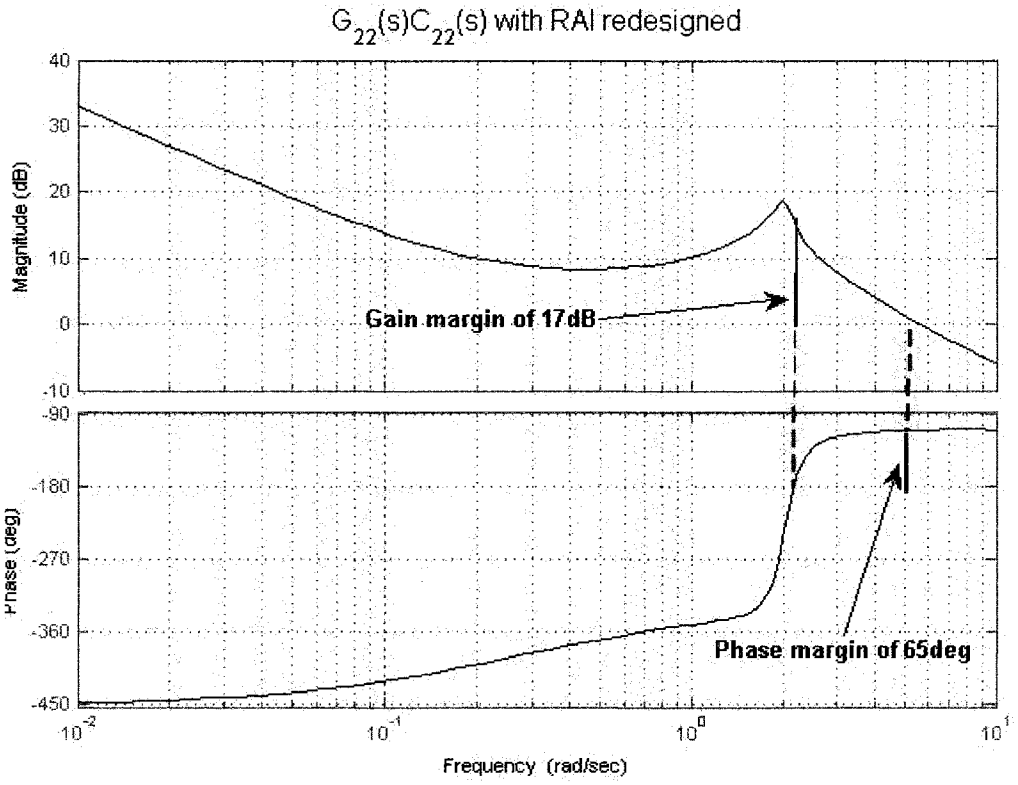


Fig. 5.36. Loop shape for $\dot{\beta} \rightarrow u_2^{'}$ with the redesigned *RAI(s)*. The active *RAI(s)* allows for integral tracking of this loop. $C_{22}(s) = \dfrac{84(s+0.2)(s+1.66)}{s^2(s+20)}$

Figure 5.37 shows the loop shape for the final $\phi \rightarrow u_1'$ closure.

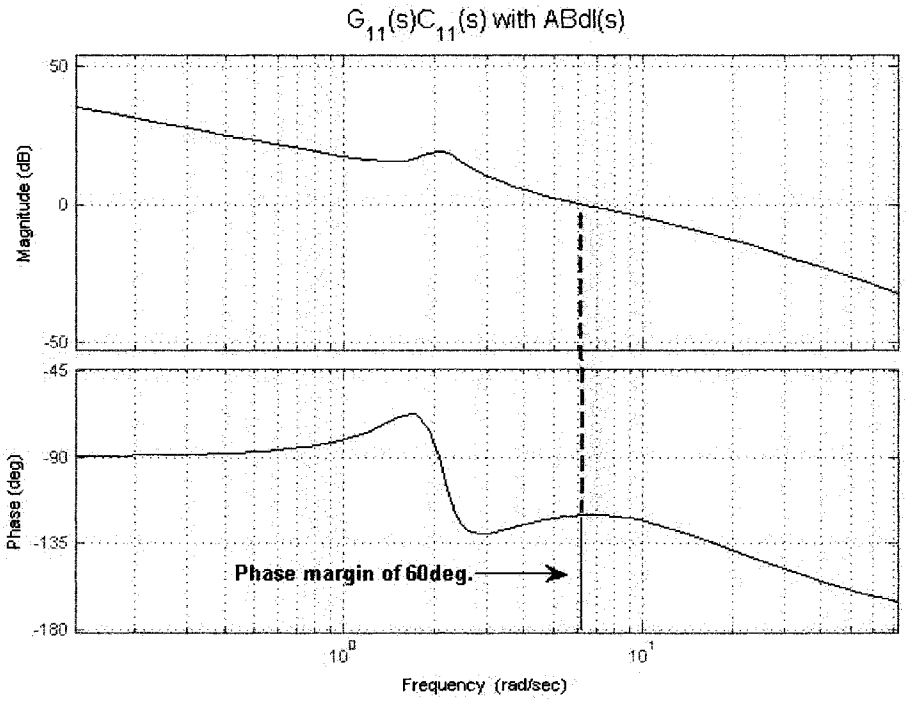


Fig. 5.37. Loop shape for $\phi \rightarrow u_1'$ with the *ABdI(s)*. The active *RAI(s)* allows for integral tracking of this loop. $C_{11}(s) = \dfrac{28(s+2.35)(s+0.0035)}{s(s+20)}$

A bandwidth of approximately 6 rad./sec. was sought for both loop closures. One should note that in Fig. 5.37, there is a significant rise and dip in phase that would not be seen for a typical aircraft. This means that the dutch roll mode inevitably found its way into the bank angle output, despite the decoupling attempts.

This does not mean that it is impossible though. This problem is revisited in Chapter 6 using the regulator we designed in section 5.5.1.

5.6 Difficulties of Roll-Yaw Decoupling for the C-5 Transport

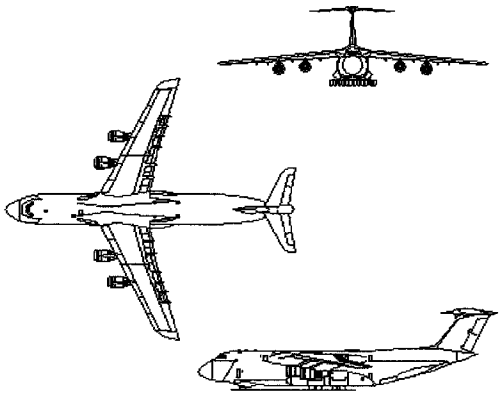


Fig. 5.38. The C-5 is a very large transport aircraft. It can carry a payload of up to 120,000 kg.

This section is intended to show an instance in which one may have difficulties with roll-yaw decoupling controllers. Not only is it frequently unnecessary for transport aircraft, but one might actually get increased cross-coupling due to some sign changes in the parameter $N'_{\delta a}$, which is Yaw moment with respect to aileron deflection. For the T-38 in section 5.2, $N'_{\delta a}$ maintains a positive value throughout its flight envelope. However, for the C-5 from Ref. [2.21], $N'_{\delta a}$ switches sign very frequently.

$$A=\begin{bmatrix} Y_v & 0 & -1 & {}^{g}/_{U_o}\cos\phi_o \\ L'_\beta & L'_p & L'_r & 0 \\ N'_\beta & N'_p & N'_r & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad B=\begin{bmatrix} Y^*_{\delta a} & Y^*_{\delta r} \\ L'_{\delta a} & L'_{\delta r} \\ N'_{\delta a} & N'_{\delta r} \\ 0 & 0 \end{bmatrix} \quad C=\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{5.51}$$

U_o	g	Y_v	L'_β	L'_p	L'_r
335	32.2	-0.0995	-0.863	-0.997	0.399
N'_β	N'_p	N'_r	$Y^*_{\delta a}$	$Y^*_{\delta r}$	$L'_{\delta a}$
0.150	-0.150	-0.187	$-9.47\text{x}10^{-5}$	0.0181	0.321
$L'_{\delta r}$	$N'_{\delta a}$	$N'_{\delta r}$	ϕ_o		
0.0852	-0.0126	-0.2820	0		

Table 5.8. Stability derivatives for the C-5 at Mach = 0.3 at Sea-Level

U_o	g	Y_v	L'_β	L'_p	L'_r
500	32.2	-0.153	-1.60	-1.36	0.344
N'_β	N'_p	N'_r	$Y^*_{\delta a}$	$Y^*_{\delta r}$	$L'_{\delta a}$
0.560	-0.113	-0.31	$-1.42\text{x}10^{-4}$	0.0271	0.516
$L'_{\delta r}$	$N'_{\delta a}$	$N'_{\delta r}$	ϕ_o		
0.229	0.050	-0.639	0		

Table 5.9. Stability derivatives for the C-5 at Mach = 0.45 at Sea-Level

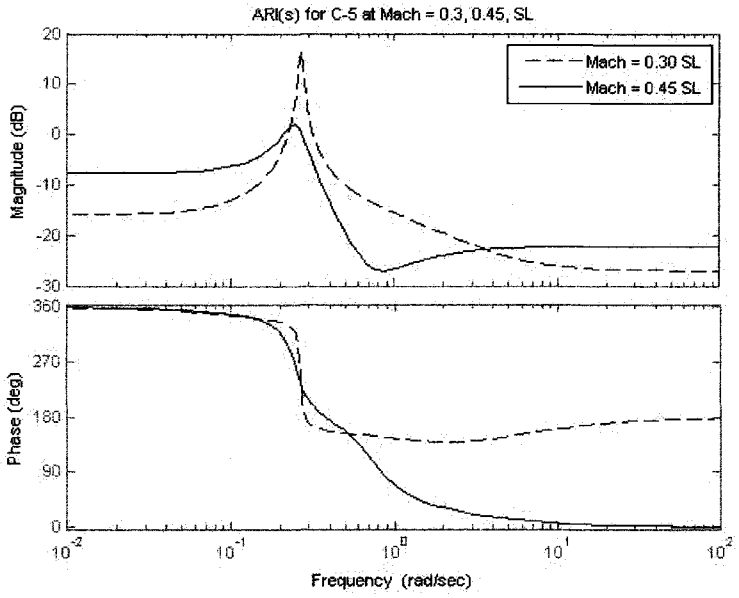


Fig. 5.39. C-5 ARI(s) bode plot for C-5 at Mach = 0.3 and 0.45 at sea-level (SL) and a bank angle of 0 deg. Notice that there is a sign reversal at high frequency between two very similar flight conditions.

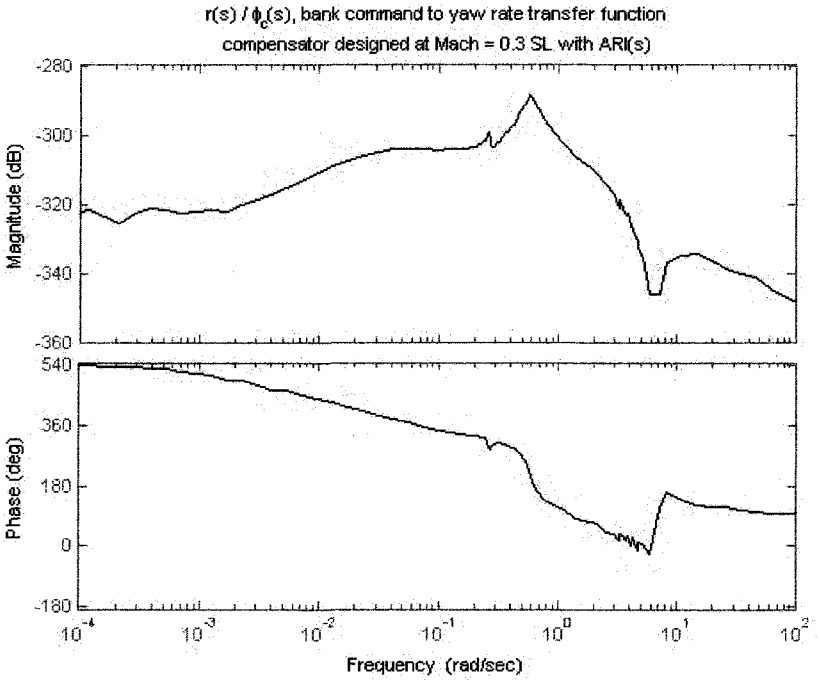


Fig. 5.40. Perfect decoupling is achieved when using the compensator. Here, the compensator designed for the Mach = 0.3 SL condition is designed for the Mach = 0.3 SL condition.

At Mach 0.3 at Sea Level (SL), $N_{\delta a} = -0.0126$, and at Mach 0.45 SL, $N_{\delta a} = 0.0500$. One notices that the Aileron to Rudder Interconnect (ARI), described in section 5.2, has a sign change between two very similar flight conditions which is shown in Fig. 5.39. It is reasonable to suggest that this small number $N_{\delta a}$ will shift at some unknown point between the two flight conditions.

What will be done in this section is to design a decoupling compensator at Mach 0.3 SL using ARI(s) at Mach 0.3 SL. Also, a completely decentralized design with ARI = 0 will also be designed at Mach 0.3 SL. Then, both the compensators will be applied to the C-5 at Mach 0.3 SL with the value of $N_{\delta a} = 0.05$.

First, the transmission zeros of the open-loop system are checked. There is only one and it is stable (listed in Table 5.8). Thus, design goals should be easily met. The fully dynamic ARI, described in section 5.2, will be used and is listed in equation 5.52.

$$ARI_{0.3SL}(s)=\frac{-0.044681(s+4.807)(s+0.3059)(s-0.1934)}{(s+1.118)(s^2+0.01388s+0.07028)} \tag{5.52}$$

The first loop closure will be to close $\phi \rightarrow u_1'$ with a bandwidth of 3 rad./sec. The compensator is listed in equation 5.53.

$$C_{u_1'\phi}(s)=\frac{66.062(s+1.118)(s+0.2642)}{s(s+10)} \tag{5.53}$$

The second loop closure, $r \rightarrow \delta_r$, is closed with a bandwidth of 1 rad./sec. The compensator is listed in equation 5.54.

$$C_{\delta r r}(s)=\frac{-3.3263(s+0.02566)(s^2+0.1708s+0.3302)}{s(s^2+0.02139s+0.07262)} \tag{5.54}$$

One can see that perfect decoupling is effective in Fig. 5.40.

The next thing to do is to design a purely decentralized compensator for the mach 0.3 SL case. The first loop closure of $\phi \rightarrow \delta_a$ is done to achieve a bandwidth of 3 rad./sec., this compensator is listed below in eq. 5.55.

$$C_{\delta a\phi}(s)=\frac{65(s+1.118)(s+0.02642)}{s(s+10)} \tag{5.55}$$

The purely decentralized design for Mach = 0.3 SL is finished by closing the $r \rightarrow \delta_r$ loop with a bandwidth of 1 rad./sec. The compensator is listed below in equation 5.56.

$$C_{\delta r r}(s)=\frac{-2.33(s+0.3784)(s+0.3301)}{s(s+0.1365)} \tag{5.56}$$

There two compensators designed for Mach = 0.3 SL, one with an active ARI(s), described by eqs. 5.52, 5.53 and 5.54. There is also one which is purely decentralized, described by eqs. 5.55 and 5.56. Now, the value of $N_{\delta a} = -0.0126$ will be changed to $N_{\delta a} = 0.0500$. The result shown in Fig. 5.41

is that across most frequencies, the decentralized controller actually shows less cross-coupling than the controller with the active ARI.

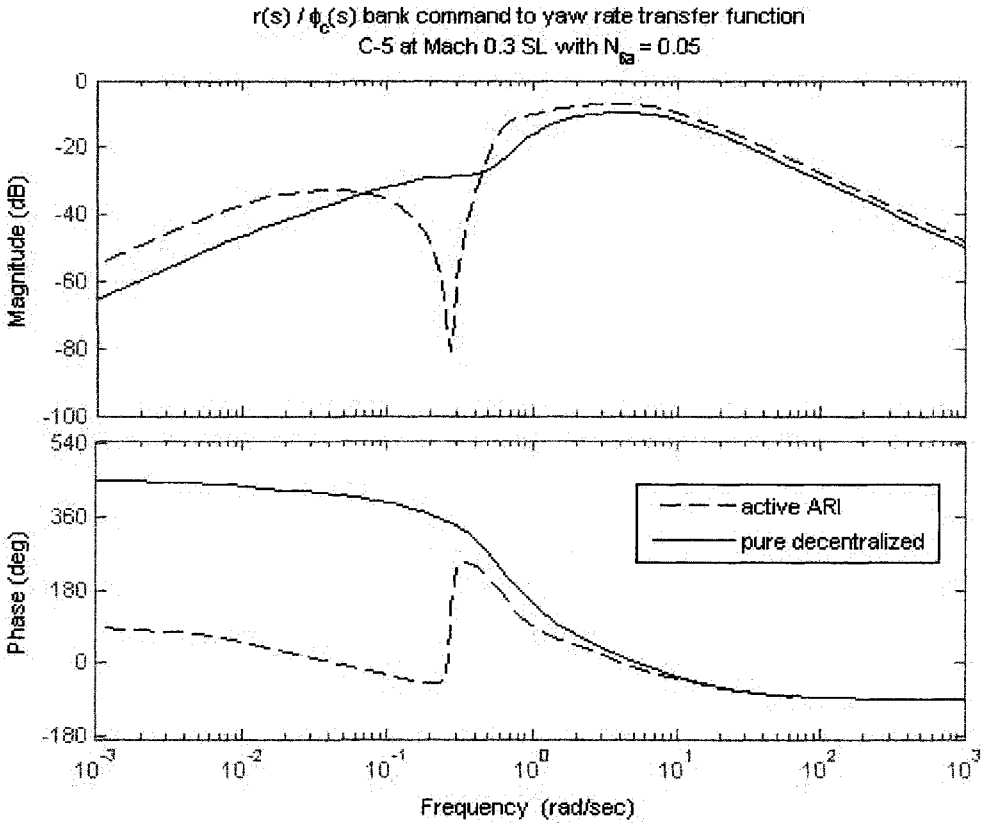


Fig. 5.41. With a change in the parameter $N_{\delta a}$ from -0.0126 to 0.05, cross coupling is increased at most frequencies when using the ARI designed around $N_{\delta a}$ = -0.0126. One can see that the purely decentralized controller actually gets slightly better behavior.

One can see that it doesn't really make much sense to make the extra effort of decoupling, since it won't be successful anyway. In this way, we can see that the plant must allow the control designer to decouple.

Example 5.1 – MATLAB code for C-5 ARI

```
Uo=335;
bank=0/180*pi;
cbank=cos(bank);
Yv=-.0995;
Lbeta=-0.863;
Nbeta=0.150;
Lp=-0.997;
Np=-0.150;
Lr=0.399;
Nr=-0.187;
Ysda=-0.947e-4;
Lda=0.321;
Nda=-0.0126;
Ysdr=.0181;
Ldr=0.0852;
Ndr=-0.282;
Alat=[Yv 0 -1 32.2/Uo*cbank;Lbeta Lp Lr 0;Nbeta Np Nr 0;0 1 0 0];
Blat=[Ysda Ysdr;Lda Ldr;Nda Ndr;0 0];
Clat=[0 0 0 1;0 0 1 0];
syslat=ss(Alat,Blat,Clat,zeros(2));
zero(syslat) %Check for unstable transmission zeros
s=tf('s');
G12ff=-inv(s*syslat(2,2))*syslat(2,1); %Dynamic ARI
[z,p,k]=ss2zp(G12ff.a,G12ff.b,G12ff.c,G12ff.d);
G12ff=minreal(s*zpk(z,p,k));
invPre=[1 0;-G12ff 1];
Pre=inv(invPre);
GP=syslat*Pre;
bode(GP(2,1))
```

5.7 Decoupling Design for the Damaged P17

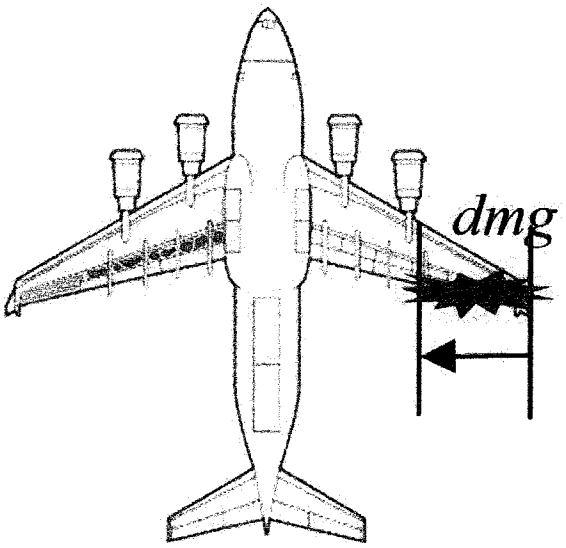


Fig. 5.42. The dynamics in this section will be called the P17. The aileron is highlighted in green and the spoiler section is highlighted in blue. The aircraft in this section is cruising at Mach = 0.6 and the altitude is 3000m

This section will show the design of a decoupling compensator for the P17 shown in Fig. 5.42. One particular aspect of this model is that there is a new stability derivative, $\partial L/\partial \alpha$, which is *rolling moment with respect to angle of attack.*

Assuming the aircraft can be restored to steady, wings-level trim, any change in angle of attack would cause the aircraft to roll. Hence, $G_{\phi\delta e}(s) \neq 0$. Normally, $G_{\phi\delta e}(s) = 0$, so this is a fairly abnormal flight condition.

What will be done in this section is to do a purely decentralized control design for the undamaged aircraft. Then, the transient response of the damaged aircraft will be observed and a controller structure will be suggested.

0% of right wing missing

$$A = \begin{bmatrix} -0.0068 & 0 & 0.0214 & 0 & 0 & 0 & -9.8200 & 0 \\ 0 & -0.3558 & 0 & 0.4536 & 0 & -208.1210 & 0 & 9.8200 \\ -0.1268 & 0 & -0.9596 & 0 & 195.5386 & 0 & 0 & 0 \\ 0 & -0.0807 & 0 & -4.0039 & 0 & 0.6557 & 0 & 0.0000 \\ 0.0004 & 0 & -0.0070 & 0 & -0.9997 & 0 & 0 & 0 \\ 0 & 0.0268 & 0 & -0.2380 & 0 & -0.7469 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.0000 & 0 & 0 & 0 & 0 \end{bmatrix}$$

0% of right wing missing

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.0043 & -7.7051 \\ -14.3602 & 0 & 0 \\ 0 & -2.5943 & 0.5650 \\ -4.7189 & 0 & 0 \\ 0 & -0.1462 & -1.4445 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

$$x = [u \quad v \quad w \quad p \quad \ldots \quad q \quad r \quad \theta \quad \phi]^T$$

$$u = [\delta_e \quad \delta_a \quad \delta_r]^T$$

20% of right wing missing, dmg = 0.20

$$A = \begin{bmatrix} -0.0069 & -0.0163 & 0.0190 & 0.1208 & 0.0031 & 0.4987 & -9.8200 & -0.0000 \\ -0.0000 & -0.3401 & 0.0005 & 0.2674 & -0.0031 & -208.6802 & 0.0000 & 9.8200 \\ -0.1361 & -0.0539 & -0.8945 & -2.0068 & 196.0958 & 0.3438 & 0.0000 & 0.0000 \\ -0.0016 & -0.0881 & 0.0189 & -3.2827 & -0.0905 & 0.5612 & 0.0000 & 0.0000 \\ 0.0015 & -0.0014 & -0.0009 & -0.0545 & -0.9739 & 0.0081 & 0.0000 & 0.0000 \\ -0.0000 & 0.0262 & 0.0008 & -0.1942 & -0.0051 & -0.8022 & 0.0000 & 0 \\ 0 & 0 & 0 & 0 & 1.0000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.0000 & 0 & 0 & 0 & 0 \end{bmatrix}$$

20% of right wing missing, dmg = 0.20

$$B = \begin{bmatrix} 0.0108 & 0.0755 & 0.8436 \\ -0.0109 & -0.0757 & -8.2060 \\ -11.1076 & -1.0564 & 0.4529 \\ -0.2969 & -2.2730 & 0.7388 \\ -4.1552 & -0.0399 & 0.0103 \\ -0.0174 & -0.1215 & -1.3570 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

The transfer function representation is listed below in eq. 5.57.

$$G(s) = \begin{bmatrix} G_{\theta\delta e}(s) & G_{\theta\delta a}(s) & G_{\theta\delta r}(s) \\ G_{\phi\delta e}(s) & G_{\phi\delta a}(s) & G_{\phi\delta r}(s) \\ G_{r\delta e}(s) & G_{r\delta a}(s) & G_{r\delta r}(s) \end{bmatrix} \tag{5.57}$$

The Relative Gain Array, (RGA) [2], for the undamaged plane at s = 0 is shown below in eq. 5.58.

$$RGA\big|_{s=0}^{0\%} = \begin{bmatrix} 1.00 & 0 & 0 \\ 0 & 21.1 & -20.1 \\ 0 & -20.1 & 21.1 \end{bmatrix} \tag{5.58}$$

The Relative Gain Array, (RGA), for the damaged plane at s = 0 is shown below in eq. 5.59.

$$RGA\Big|_{s=0}^{20\%} = \begin{bmatrix} 1.00 & -0.0017 & 0.0 \\ -0.027 & 16.65 & -15.63 \\ 0.025 & -15.65 & 16.63 \end{bmatrix} \tag{5.59}$$

One can see that RGA(1,1) is pretty much 1.0, thus it is a very good choice to pair the elevator, δ_e, with pitch attitude, θ. However, it would be a bad choice to pair bank angle, ϕ, with the aileron, δ_a because 16 is a large number and not close to 1.0. The same goes for pairing the rudder, δ_a, with yaw rate, *r*. At very low frequency, lateral control is best suited as a SISO problem.

5.7.1 Undamaged Control Design for P17

Transmission zeros, *TZ*, of G(s)
TZ = -0.9352, -0.0098,-0.5198

Table 5.9. MIMO Transmission zeros of G(s), which are all stable

The transmission zeros of the plant, shown in table 5.9, are all stable. One can proceed without too many problems. To start things off, $\theta \rightarrow \delta_e$ and a bandwidth of 3 rad./sec. will be the target. This leads to a compensator listed below in eq. 5.60.

$$C_{\delta e\theta}(s) = \frac{-9.25(s+0.1)(s^2+2s+1)}{s(s+1)(s+15)} \tag{5.60}$$

Next, $\phi \rightarrow \delta_a$ and a bandwidth of 3 rad./sec. will be the target. This leads to a compensator listed below in eq. 5.61.

$$C_{\delta a\phi}(s) = \frac{-16(s+0.015)(s+3.75)}{s(s+15)} \tag{5.61}$$

The final loop closure will be paired with $r \rightarrow \delta_r$ and a bandwidth of 1 rad./sec.

$$C_{\delta rr}(s) = \frac{-0.574(s^2+0.966s+6.48)}{s(s+0.575)} \tag{5.62}$$

Also, a traditional washout filter design for the yaw channel will be pursued as well.

$$C_{\delta rr}(s) = \frac{-2s}{(s+0.5)} \tag{5.63}$$

Of course, with a washout filter, one would want to set the commanded *r=0*, such that the washout filter would mostly be a regulator type design.

5.7.2 Damaged Control Design for P17 with Elevator to Aileron Interconnect

The rolling moment created by a change in angle of attack, $\partial L/\partial \alpha$ can become quite large. What will be investigated is a Elevator to Aileron Interconnect (EAI) to counteract the rolling moment produced by changes in the elevator (which causes a change in angle of attack). This is shown in Fig. 5.43. Approximate EAI(s) in the 10^{-1} to the 1 rad./sec. regime are listed in eq. 5.65.

Here the EAI(s) is designed according to the process described in section 5.2 and eq. 5.64 results.

$$EAI(s) = G^{ff}_{\delta a \delta e}(s) \approx \left[G_{p\delta a}(s)\right]^{-1} G_{p\delta e \backslash}(s) \tag{5.64}$$

No unstable pole-zero cancellations are made by adding the EAI(s) in the precompensator.

$$\begin{aligned}
EAI(s)\big|_{dmg=0.1} &\approx \frac{-0.1\left(s^2+s+6\right)}{\left(s^2+1.8s+1.1\right)} \\
EAI(s)\big|_{dmg=0.2} &\approx \frac{-7}{\left(s^2+1.8s+1.1\right)} \\
EAI(s)\big|_{dmg=0.3} &\approx \frac{-14}{\left(s^2+1.8s+1.1\right)} \\
EAI(s)\big|_{dmg=0.4} &\approx \frac{-25}{\left(s^2+1.8s+1.1\right)}
\end{aligned} \tag{5.65}$$

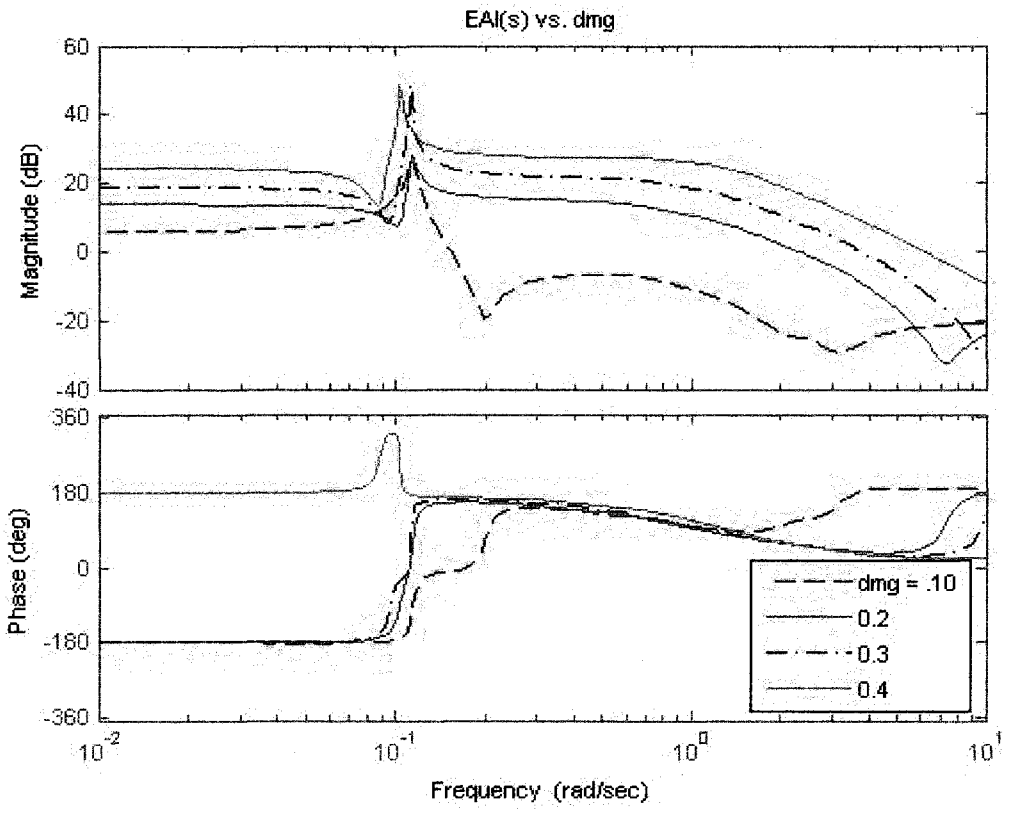


Fig. 5.43. Elevator to Aileron Interconnect EAI(s) for P17. Notice how large a jump it makes from dmg = 0.1 to dmg = 0.2 and above.

Notice how very large the EAI(s) is in eq. 5.65. For 40% wing damage, this means that if the elevator moves down 1 deg. then the ailerons will move 25 deg., this is a huge imbalance. It immediately becomes obvious that ailerons alone will not be enough to deal with this massive force imbalance. Therefore, one must get more rolling moment. This will be achieved by adding asymmetric spoiler and taileron.

5.7.3 P17 Mega Aileron Design and Elevator to Mega Aileron Interconnect

In section 5.4.1, it was noticeable how large EAI(s) became, and it was deemed necessary to get more rolling moment out of the available hardware on the aircraft. Thus, differential tail (i.e. taileron), and differential spoilers were added. The new B matrix is shown below.

0% of right wing missing

$$B=\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0.0043 & 0.0052 & -0.0023 & -7.7051 \\ -14.3602 & 0 & 0 & 0 & 0 \\ 0 & -2.5943 & -0.8421 & -0.2669 & 0.5650 \\ -4.7189 & 0 & 0 & 0 & 0 \\ 0 & -0.1462 & -0.1756 & 0.0782 & -1.4445 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$u=\begin{bmatrix}\delta_e & \delta_a & \delta_{sp} & \delta_{ta} & \delta_r\end{bmatrix}^T$$

Now, aileron, differential spoiler and tail will be added together to create a new pseudo-control, a "mega-aileron", $\delta_{ma}=\delta_a+\delta_{sp}+\delta_{ta}$. Updating the B matrix with the "mega-aileron" yields the following B matrix below.

0% of right wing missing

$$B=\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.0072 & -7.7051 \\ -14.3602 & 0 & 0 \\ 0 & -3.7033 & 0.5650 \\ -4.7189 & 0 & 0 \\ 0 & -0.2435 & -1.4445 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$u=\begin{bmatrix}\delta_e & \delta_{ma} & \delta_r\end{bmatrix}^T$$

Because a more powerful aileron was created, one must reduce the gain in the bank angle to aileron channel. The update is shown in eq. 5.66.

$$C_{\delta na\phi}(s)=\frac{-10(s+0.015)(s+3.75)}{s(s+15)} \tag{5.66}$$

The increase in available rolling moment is not that significant, which is unfortunate. Nonetheless, the re-design will proceed with calculating Elevator to Mega-Aileron Interconnect, EMAI(s).

From fig. 5.44, one can see that the counteracting the rolling moment produced by elevator movement is still very large. It is so large that it is difficult to believe that the aircraft is even salvageable at damage levels of 40% or greater. But from these plots, it seems that the possibility of successfully maintaining wings-level flight while descending/ascending is quite small.

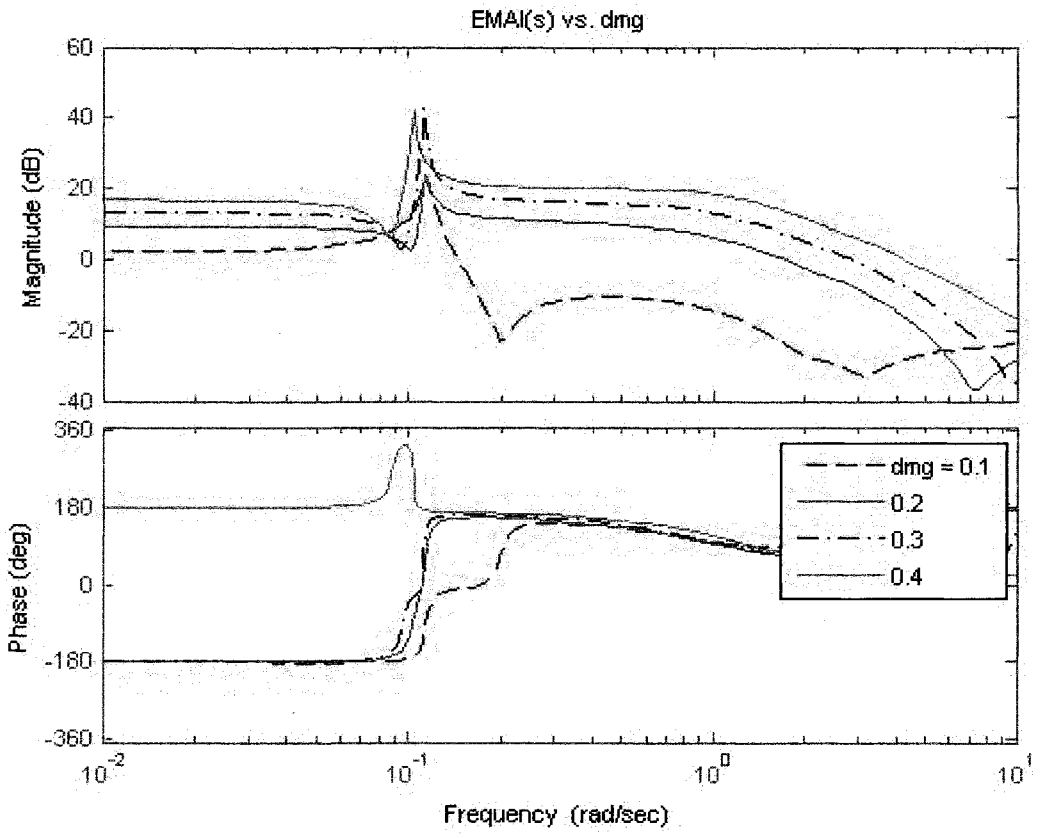


Fig. 5.44. Elevator to Mega-Aileron Interconnect EMAI(s) for P17. Here, mega aileron consists of aileron, differential spoilers and differential tail.

$$
\begin{aligned}
EMAI(s)\big|_{dmg=0.1} &\approx \frac{-0.07\left(s^2+s+6\right)}{\left(s^2+1.8s+1.1\right)} \\
EMAI(s)\big|_{dmg=0.2} &\approx \frac{-3}{\left(s^2+1.8s+1.1\right)} \\
EMAI(s)\big|_{dmg=0.3} &\approx \frac{-10}{\left(s^2+1.8s+1.1\right)} \\
EMAI(s)\big|_{dmg=0.4} &\approx \frac{-14}{\left(s^2+1.8s+1.1\right)}
\end{aligned}
\tag{5.67}
$$

Approximate EMAI(s) in the 10^{-1} to the 1 rad./sec. realm are listed in eq. 5.67.

Also, a Elevator to Rudder Interconnect, ERI(s), will be pursued, which is shown in Fig. 5.45. Approximate ERI(s) in the 10^{-1} to the 1 rad./sec. realm are listed in eq. 5.68.

$$
\begin{aligned}
ERI(s)\big|_{dmg=0.1} &\approx \frac{-0.02\left(s^2-4s+6\right)}{\left(s^2+0.15s+0.1\right)} \\
ERI(s)\big|_{dmg=0.2} &\approx \frac{-0.7\left(s^2-1.55s+1.08\right)}{\left(s^2+0.45s+0.18\right)\left(s^2+2s+1\right)} \\
ERI(s)\big|_{dmg=0.3} &\approx \frac{-1\left(s^2-1.55s+1.08\right)}{\left(s^2+0.45s+0.18\right)\left(s^2+2s+1\right)} \\
ERI(s)\big|_{dmg=0.4} &\approx \frac{-2\left(s^2-1.55s+1.08\right)}{\left(s^2+0.45s+0.18\right)\left(s^2+2s+1\right)}
\end{aligned}
\tag{5.68}
$$

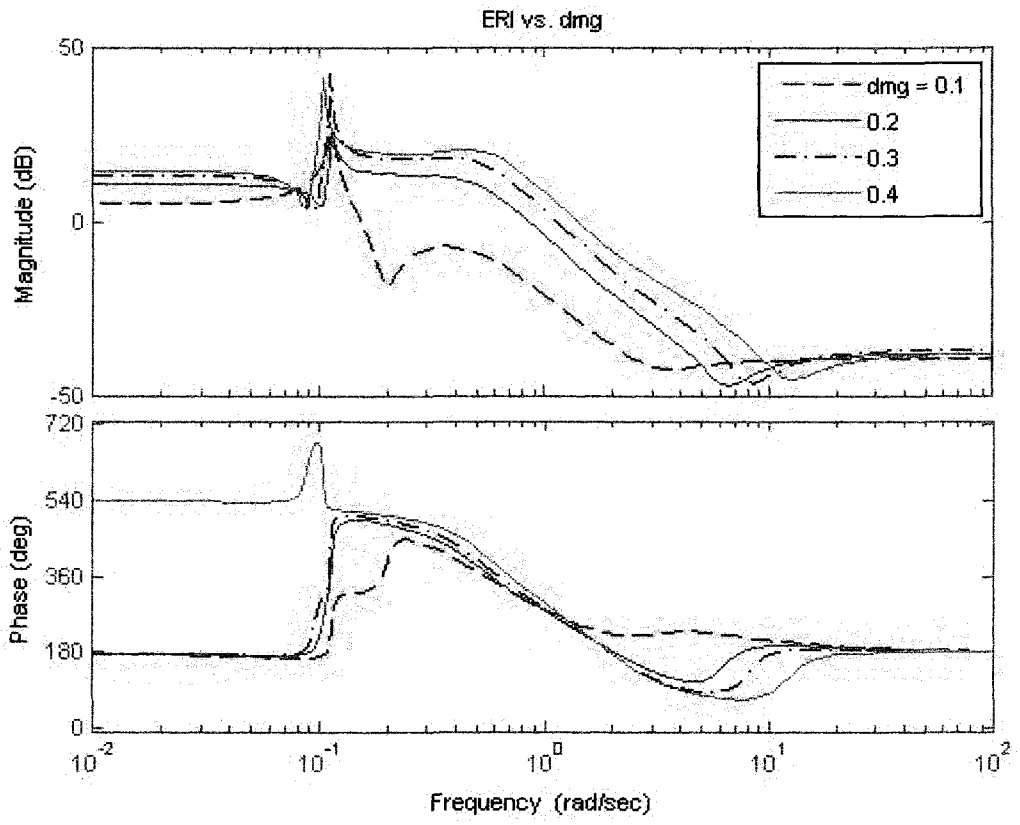


Fig. 5.45. Elevator to Rudder Interconnect ERI(s) for P17.

The total pre-compensator structure is listed in eq. 5.69.

$$
P(s)=\begin{bmatrix} 1 & 0 & 0 \\ EMAI(s) & 1 & 0 \\ ERI(s) & 0 & 1 \end{bmatrix} \tag{5.69}
$$

One should note that the emphasis on decoupling in this section has been to focus primarily on low frequency dynamics. Previous sections have focused on emphasizing decoupling at higher

frequencies to avoid inverting ill conditioned matrices at low frequency. With the particular interconnects used here, there is no concern of inverting an ill conditioned matrix.

5.7.4 P17 Transient Response Behavior

Figure 5.46 shows the transient response with the pre-compensator at 20% of the right wing gone. The bank angle deviation is reduced, yet still takes a long time to settle down to 0. The yaw rate deviation is actually increased, which might be a result of inaccurate model reduction of the ERI(s).

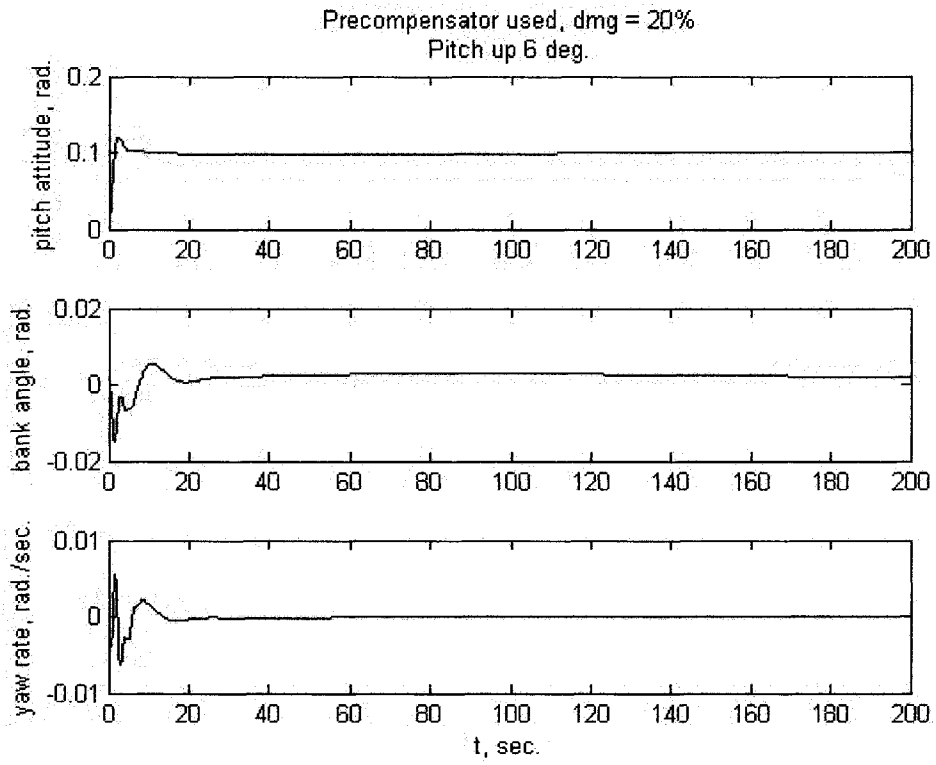


Fig. 5.46. Transient response with pre-compensator is shown here. Notice that the bank angle deviation is reduced. However, the yaw rate deviation is actually increased because the high frequency content of the ERI is inaccurate.

Figure 5.47 shows the transient response without the pre-compensator at 20% of the right wing gone. A step command to pitch up 6 deg. (approximately 0.1 rad) is given. The bank angle takes a long time to settle down.

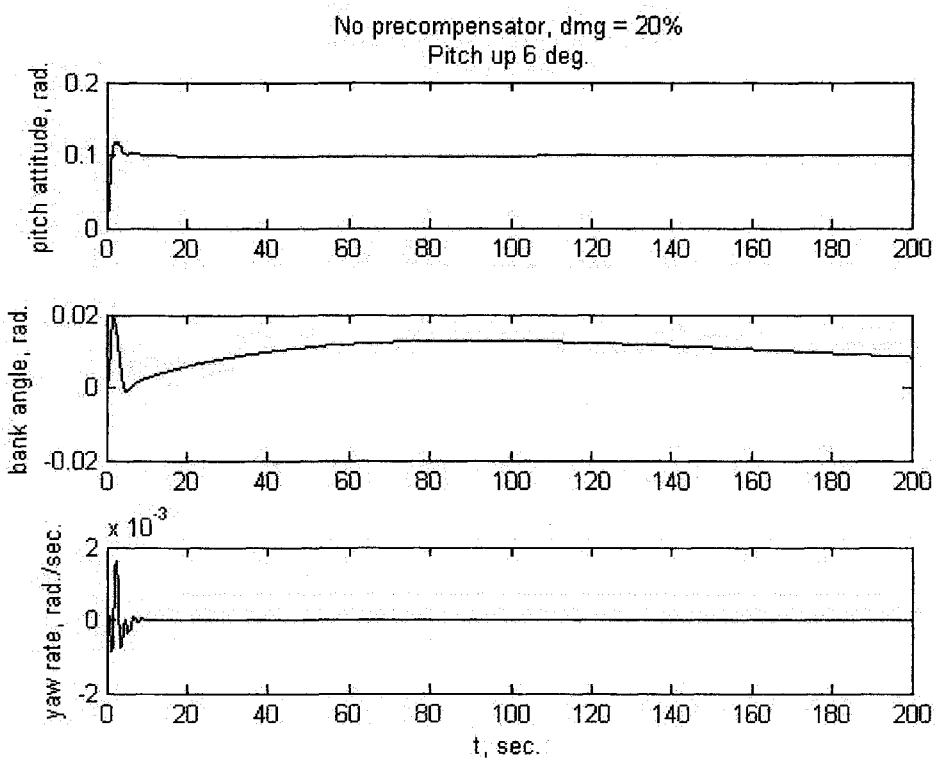


Fig. 5.47. Transient response without pre-compensator is shown here. Notice that it takes a very long time for the bank angle to settle down. Actuator dynamics are included in this simulation. No disturbances from lift or roll are included.

Although there is potential for very mild improvements in decoupling performance, it would probably not greatly affect the survivability of the damaged aircraft. Recall from chapter 1 that perfect situational awareness is not possible. In a reconfigurable control system, the EMAI(s) and ERI(s) would inevitably misapplied to the undamaged aircraft. In this way, the presence of the EMAI(s) and ERI(s) would be a threat to the undamaged aircraft.

It should be fairly intuitive that a pre-compensator designed for the right wing damage could be detrimental if it were applied to an undamaged aircraft. For an undamaged aircraft, it makes absolutely no sense to have an EAI(s) or ERI(s). In fact, putting an EAI(s) or ERI(s) which is designed for right wing damage on an undamaged aircraft is a lot like having left wing damage.

5.7.5 Wing Damaged P17 Design Conclusion

Perfect decoupling requires a perfect model of the plant. Approximate decoupling (if this is not a contradiction in terms) requires an approximate model of the plant. Also, the use of pre-compensators in this case represents a threat to the undamaged aircraft.

The disturbance rejection properties of the bank angle loop can be exploited for wing damaged aircraft. This can be achieved without using pre-compensators. Refer to section 4.6 for ideas on decentralized techniques for improving performance of wing damaged aircraft.

Table 5.10. Suggested design procedure for small to moderate wing damage on transport aircraft

Design Step	Bandwidth	Handling Qualities	Notes
1 – Assume outputs θ, ϕ, r, check that zeros are stable	NA	NA	Use matlab command "zero" or find roots of det(G(s)) = 0, with G(s) square.
2 – Make use of any available control surfaces for roll	NA	NA	Try to create a "mega-aileron"
3 – Close bank angle to "mega-aileron" $\phi \rightarrow \delta_{ma}$ for undamaged aircraft	As high as can reasonably be reached	Unknown	Disturbance rejection is a high priority
4 – Yaw rate loop closure $r \rightarrow \delta_r$ on undamaged aircraft	A reasonable quantity, 1 to 3 rad./sec.	Unknown	
5 – Pitch attitude closure $\theta \rightarrow \delta_e$ on undamaged aircraft	Slowing the pitch attitude loop will help minimize cross coupling	Unknown	
6 – If wing damage information is to be available via *perfect* fault isolation, design EMAI(s) and ERI(s)	NA	Unknown	Applying EMAI(s) or ERI(s) is a lot like damaging the opposite wing! *Do not apply EMAI(s) or ERI(s) to an undamaged aircraft.*

Chapter 6 - Decoupling Control by Inverse Dynamics

Achieving transient specifications set out by either complimentary sensitivity, $T(s)$, or sensitivity, $S(s)$ can be met by a fairly complicated H_∞ model matching problem. Model matching also leads to a singular LQR problem [6.1]. This means that one cannot use optimal control techniques to exactly achieve transient specifications set out in the frequency domain. However, exact model matching can be met very simply by using inverse dynamics to generate a controller.

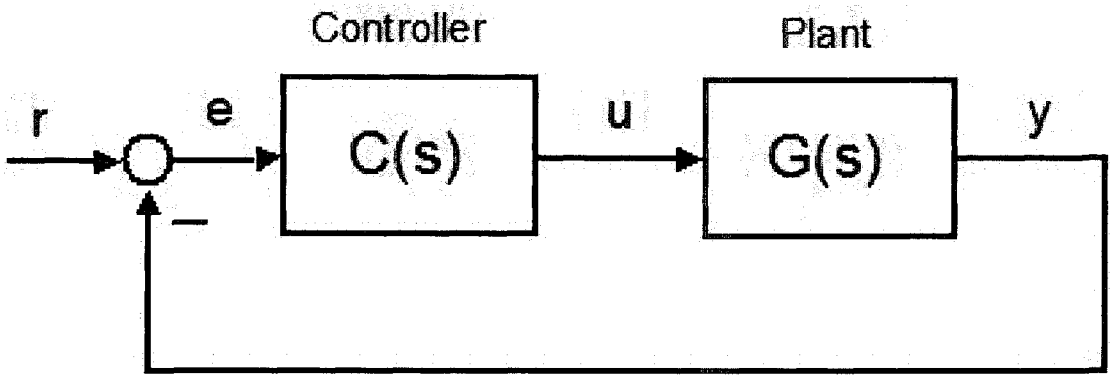


Fig. 6.1. A single degree of freedom feedback configuration.

The idea is that a desired loop shape, $L(s)$, is achieved by a design of a controller, $C(s)$.

$$G(s)C(s) = L(s)$$

In order to synthesize inversion-based controllers, we need to be able to invert the plant, $G(s)$. This is no problem if $G(s)$ is already biproper (i.e. has an invertible *D* matrix). However, it is frequently the case that $G^{-1}(s)$ is an improper transfer function, for which there is no state space description. The solution is to use an "interactor matrix", $\varepsilon_L(s)$, to achieve inversion.

A very simple example of the interactor matrix is that if the outputs to be followed are velocities and the inputs are forces, then there is one differentiation to get accelerations, $\varepsilon_L(s) = s$. This way, $ma = F$ (Force = mass x acceleration) which is an invertible system. We say that $m\dot{v} = F$ is not an invertible system because the state space realization with v as an output does not have a *D* matrix.

Aside from this, we will basically be making pole-zero cancellations for controller synthesis. This is good when poles are stable and located far away from the $j\omega$ axis. However, if poles are

unstable or located close to the $j\omega$ axis, then it is best to add damping by designing a regulator. One can use direct output feedback or one can use pole placement with an observer. The author personally prefers direct output feedback because it is easier to implement and it encourages the control designer to better understand the system dynamics.

This design procedure is called the $G^{reg}(s)-C^{reg}(s)$ design procedure and is shown below.

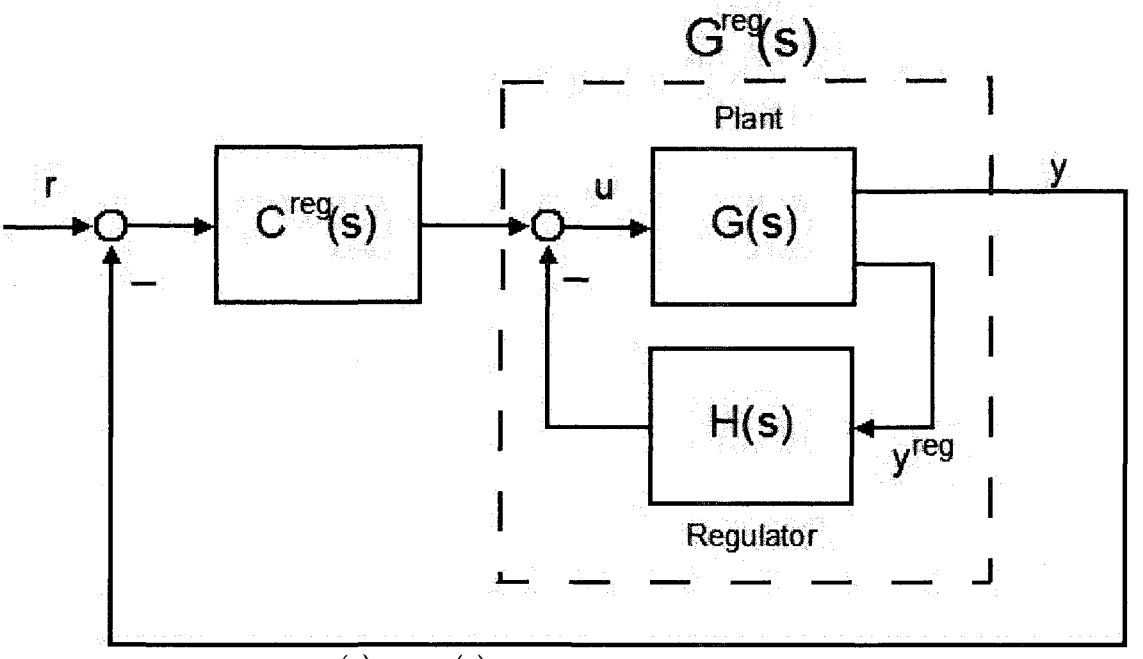


Fig. 6.2. A two-step design, $G^{reg}(s)-C^{reg}(s)$, is shown here. The first step is to design a regulator, *H(s)*, for the plant. The next step is to use a decoupling compensator $C^{reg}(s)$ to decouple the outputs, y, which may not be the same outputs used for the regulator, y^{reg}.

The design steps are given below.

Step 1 – Design a reasonable and stablilizing regulator, $H(s)$, for the plant, $G(s)$

Step 2 – Invert $G^{reg}(s)=(I+G(s)H(s))^{-1}G(s)$ to obtain the desired loop shape, $L(s)$, with $C^{reg}(s)$.

With this control scheme, there are robustness issues. Basically, if the plant is poorly conditioned across a broad frequency range, then it is a bad idea to do decoupling or doing inversion based control. One can do this by either looking at a frequency domain plot of the RGA, as in section 4.2. Also, one could look at an "optimally scaled" singular value plot. Two simple examples are shown below.

$$G(s)=\begin{bmatrix}1 & 1\\ 1 & 1.0001\end{bmatrix}\frac{1}{s+2}\Rightarrow \textit{poorly conditioned} \Rightarrow \textit{do not invert}$$

$$G(s)=\begin{bmatrix}1 & -1\\ 1 & 1.0001\end{bmatrix}\frac{1}{(s+2)(s+4)}\Rightarrow \textit{well conditioned} \Rightarrow \textit{invert}$$

Many design examples are given in this chapter. Most design examples are lateral aircraft control problems. One design is a longitudinal aircraft control problem. An alternative discussion of diagonal decoupling for lateral aircraft control can be found in Ref. [6.2]. There are many different styles of decoupling control techniques [6.3, 2.3, 2.6, 2.16].

6.1 Interactor Matrices and Inversion of Square Systems

In order to synthesize inversion-based controllers, we need to be able to invert the plant, $G(s)$. This is possible if $G(s)$ is already biproper (i.e. has an invertible *D* matrix). However, it is frequently the case that $G^{-1}(s)$ is an improper transfer function, for which there is no state space description.

The idea is that a diagonal loop shape, $L(s)$, is achieved by a design of a compensator, $C(s)$.

$$G(s)C(s) = L(s) \tag{6.1}$$

But it is difficult to do $C(s)=G^{-1}(s)L(s)$ because $G^{-1}(s)$ would be an improper transfer function. So, we introduce the "left interactor matrix" $\varepsilon_L(s)$, which makes things biproper.

$$\begin{aligned}&\varepsilon_L(s)G(s)C(s)=\varepsilon_L(s)L(s)\\ &C(s)=(\varepsilon_L(s)G(s))^{-1}\varepsilon_L(s)L(s)=\Lambda_G^{-1}(s)\varepsilon_L(s)L(s)\end{aligned} \tag{6.2}$$

The irony is that the left interactor [6.3], $\varepsilon_L(s)$, is itself an improper transfer function.

If a square transfer function is not invertible, then finding an interactor matrix will not be successful. It is not hard to see that if $G(s)$ is not invertible, then the determinant is 0 for all *s*, i.e. ($\det(G(s))=0 \quad \forall \quad s$).

Necessary and sufficient condition for un-invertibility/functional un-controllability of square plants

$$\det(G(s))=0 \quad \forall \quad s$$

6.1.1 "Classical" Left Interactor Matrices

The left interactor matrix is sometimes called the "differentiator matrix", but it is basically any linear combination of s^0, s^1, s^2, s^3,s^4, etc., that makes the plant, $G(s)$, into a biproper transfer function, $\Lambda_G(s)$. The same is true in the digital domain, only one would be use z^0, z^1, z^2, z^3,z^4, etc. Any of the following algorithms can be extended to the digital domain by simply using the forward shift, *z*, in place of *s*.

A really simple example of the interactor matrix would be a dc motor with an input of armature voltage and output of angular velocity. The interactor matrix for this plant would just be *s*. If the output were angular position, then the interactor matrix would just be s^2.

The definition provided by Wolovich in 1976 [6.4] specified that the left interactor matrix, $\varepsilon_L(s)$, was lower triangular (though this is not absolutely necessary). Wolovich's solution only works for 2x2,2x4 or 4x2 transfer functions.

$$\lim_{s\to\infty} \varepsilon_L(s)G(s) = \lim_{s\to\infty} \Lambda_G(s) = D_\Lambda \quad \det(D_\Lambda) \neq 0 \quad or \quad \infty \tag{6.3}$$

A quick solution to this left interactor problem is to look across the rows of $G(s)$ and make each row proper independently. If it works, then this will form a diagonal interactor $\varepsilon_L(s)$.

Example 6.1:

$$G(s) = \begin{bmatrix} \dfrac{2(s+2)}{(s+1)^2} & \dfrac{0.2(s-0.25)}{(s+1)^2} \\ \dfrac{-0.1}{(s+1)^2} & \dfrac{1}{(s+1)^2} \end{bmatrix} \quad \varepsilon_L(s) = \begin{bmatrix} s & 0 \\ 0 & s^2 \end{bmatrix} \quad \lim_{s\to\infty} \varepsilon_L(s)G(s) = \lim_{s\to\infty} \Lambda_G(s) = \begin{bmatrix} 2 & 0.2 \\ -0.1 & 1 \end{bmatrix}$$

$$\det\left(\begin{bmatrix} 2 & 0.2 \\ -0.1 & 1 \end{bmatrix}\right) \neq 0 \quad or \quad \infty$$

Wolovich's "Classical" Interactor Matrix Algorithm for 2x2 Transfer Functions:

Step 1.0: This is the step previously used in example 6.1 and is extendible to any size plant. One would start with an initial guess for each row of $G(s)$, labeled as $G_{i,\bullet}(s)$ and increment n_i, until each row of $\lim_{s\to\infty} s^{n_i} G_{i,\bullet}(s) = F_{i,\bullet,0}$ were finite and non-zero.

For each *i=1,2*:

Increment n_i until $\lim_{s\to\infty} s^{n_i} G_{i,\bullet}(s) = F_{i,\bullet,0}$ is finite and non-zero

Assemble each row of $F_{i,\bullet,0}$ together into F_0, $F_0 = \begin{bmatrix} F_{1,\bullet,0} \\ \vdots \\ F_{m,\bullet,0} \end{bmatrix}$

Step 1.1: Assemble each row of $F_{i,\bullet,0}$ together into F_0 and check to see if it has a non-zero determinant.

If F_0 has full rank 2 or has a non-zero determinant, then we can set $F_0 = D_\Lambda$ and

$\varepsilon_L(s)_0 = diag\begin{pmatrix} s^{n_1} & s^{n_2} \end{pmatrix}$.

If $\det(F_0)$ is non-zero:

Set $F_0 = D_\Lambda$ and $\varepsilon_L(s) = \varepsilon_L(s)_0 = diag\begin{pmatrix} s^{n_1} & s^{n_2} \end{pmatrix}$, algorithm is done

Else:

Continue onto Step 2.0

Step 2.0: Row 1 and 2 of F^o are linearly dependent. Find a constant, v_1, such that $F_{2,\bullet,0} = v_1 F_{1,\bullet,0}$.

Set $\varepsilon_L(s)_{2,\bullet,1} = s^x \left(\begin{bmatrix} 0 & s^{n_2} \end{bmatrix} - v_1 \begin{bmatrix} s^{n_1} & 0 \end{bmatrix} \right)$, and increment x until

$\lim_{s\to\infty} \varepsilon_L(s)_{2,\bullet,1} G_{2,\bullet}(s) = F_{2,\bullet,1}$ gives a finite and non-zero solution. Set $F_{1,\bullet,1} = F_{1,\bullet,0}$, and assemble F_1. If $\det(F_1)$ is non-zero, then the algorithm is finished.

Set $v_1 = F_{2,\bullet,0} / F_{1,\bullet,0}$ and $\varepsilon_L(s)_{2,\bullet,1} = s^x \left(\begin{bmatrix} 0 & s^{n_2} \end{bmatrix} - v_1 \begin{bmatrix} s^{n_1} & 0 \end{bmatrix} \right)$

Increment x until $\lim_{s\to\infty} \varepsilon_L(s)_{2,\bullet,1} G_{2,\bullet}(s) = F_{2,\bullet,1}$ is finite and non-zero.

If $\det(F_1)$ is non-zero:

Set $F_1 = D_\Lambda$ and $\varepsilon_L(s) = \begin{bmatrix} s^{n_1} & 0 \\ \varepsilon_L(s)_{2,1,1} & \varepsilon_L(s)_{2,2,1} \end{bmatrix}$, algorithm is done

Else:

Continue onto Step 2.1

Step 2.1: Initialize $k = 2$. Row 1 and 2 of F_{k-1} are linearly dependent. Find a constant, v_k, such that $F_{2,\bullet,k-1} = v_k F_{1,\bullet,k-1}$. Set $\varepsilon_L(s)_{2,\bullet,k} = s^x\left(\varepsilon_L(s)_{2,\bullet,k-1} - v^k\begin{bmatrix} s^{n_1} & 0\end{bmatrix}\right)$, and increment x until $\lim_{s\to\infty}\varepsilon_L(s)_{2,\bullet,k}G_{2,\bullet}(s) = F_{2,\bullet,k}$ gives a finite solution. Set $F_{1,\bullet,k} = F_{1,\bullet,k-1}$, and assemble F_k. If $\det(F_k)$ is non-zero, then the algorithm is finished. Otherwise, increment k and repeat step 2.1.

Initialize $k = 2$.

Set $v_k = F_{2,\bullet,k-1} / F_{1,\bullet,k-1}$ and $\varepsilon_L(s)_{2,\bullet,k} = s^x\left(\varepsilon_L(s)_{2,\bullet,k-1} - v^k\begin{bmatrix} s^{n_1} & 0\end{bmatrix}\right)$

Increment x until $\lim_{s\to\infty}\varepsilon_L(s)_{2,\bullet,k}G_{2,\bullet}(s) = F_{2,\bullet,k}$ is finite and non-zero.

If $\det(F_k)$ is zero:

Increment k and iterate.

Else:

$F_k = D_\Lambda$ and $\varepsilon_L(s) = \begin{bmatrix} s^{n_1} & 0 \\ \varepsilon_L(s)_{2,1,k} & \varepsilon_L(s)_{2,2,k}\end{bmatrix}$, algorithm is done

Example 6.2:

Step 1.0:

$$G(s) = \begin{bmatrix} s & -1 \\ 2s & 5\end{bmatrix}\frac{1}{(s+3)(s+5)}$$

$$\varepsilon_L(s)_0 = diag(s \quad s) \quad F_0 = \begin{bmatrix} 1 & 0 \\ 2 & 0\end{bmatrix}$$

$$det(F_0) = 0$$

Step 2.0:

$$v_1 = [2 \quad 0]/[1 \quad 0] = 2$$

$$\varepsilon_L(s)_{2,\bullet,1} = s^x([0 \quad s] - 2[s \quad 0]) \quad x = 1$$

$$F_1 = \begin{bmatrix} 1 & 0 \\ 0 & 7\end{bmatrix} \quad \det(F_1) \neq 0$$

$$\varepsilon_L(s) = \begin{bmatrix} s & 0 \\ -2s^2 & s^2\end{bmatrix}$$

Example 6.3:

Step 1.0:

$$G(s)=\begin{bmatrix} \dfrac{1}{s+1} & \dfrac{1}{s+2} \\ \dfrac{1}{(s+1)(s+3)} & \dfrac{1}{(s+1)(s+4)} \end{bmatrix}$$

$$\varepsilon_L(s)_0 = diag\begin{pmatrix} s & s^2 \end{pmatrix} \quad F_0 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

$$\det(F_0)=0$$

Step 2.0:

$$v_1 = [1 \quad 0]/[1 \quad 0] = 1$$

$$\varepsilon_L(s)_{2,\bullet,1} = s^x\left([0 \quad s^2]-1[s \quad 0]\right) \quad x=1$$

$$\varepsilon_L(s)_1 = \begin{bmatrix} s & 0 \\ -s^2 & s^3 \end{bmatrix}$$

$$F_1 = \begin{bmatrix} 1 & 1 \\ -3 & -3 \end{bmatrix} \quad \det(F_1)=0$$

Step 2.1:

$$k=2$$

$$v_2 = [1 \quad 1]/[-3 \quad -3] = -3$$

$$\varepsilon_L(s)_{2,\bullet,2} = s^x\left([-s^2 \quad s^3]+3[s \quad 0]\right) \quad x=1$$

$$\varepsilon_L(s)_2 = \begin{bmatrix} s & 0 \\ -s^3+3s^2 & s^4 \end{bmatrix}$$

$$F_2 = \begin{bmatrix} 1 & 1 \\ 9 & 11 \end{bmatrix} \quad \boldsymbol{det}(F_2)\neq 0$$

$$\varepsilon_L(s) = \begin{bmatrix} s & 0 \\ -s^3+3s^2 & s^4 \end{bmatrix}$$

"Classical" Interactor Matrix Algorithm for 3x3 Transfer Functions:

Wolovich's algorithm [6.4] has only been explicitly demonstrated for 2x2, 2x4 and 4x2 transfer functions, here we extend the "Classical" style to 3x3 transfer functions.

Step 1.0: One would start with an initial guess for each row of $G(s)$, labeled as $G_{i,\bullet}(s)$ and increment n_i, until each row of $\lim_{s\to\infty} s^{n_i} G_{i,\bullet}(s) = F_{i,\bullet,0}$ were finite and non-zero.

For each *i=1,2,3*:

Increment n_i until $\lim_{s\to\infty} s^{n_i} G_{i,\bullet}(s) = F_{i,\bullet,0}$

Step 1.1: Assemble each row of $F_{i,\bullet,0}$ together into F_0 and check to see if it has a non-zero determinant.

If F_0 has full rank *3* or has a non-zero determinant, then we can set $F_0 = D_\Lambda$ and

$$\varepsilon_L(s) = \varepsilon_L(s)_0 = diag\begin{pmatrix} s^{n_1} & s^{n_2} & s^{n_3} \end{pmatrix}.$$

If $\det(F_0)$ is non-zero:

Set $F_0 = D_\Lambda$ and $\varepsilon_L(s) = \varepsilon_L(s)_0 = diag\begin{pmatrix} s^{n_1} & s^{n_2} & s^{n_3} \end{pmatrix}$, algorithm is done

Else:

Initialize $k = 0$

Continue onto Step 1.2

Step 1.2: The rank of F_k could be 2. In this case, row 3 of F_k could be expressed as a linear combination of rows 1 and 2. The rank of F_k could also be 1. In this case, both rows 2 and 3 of F_k are linearly dependent on row 1.

If $rank(F_k) = 2$:

Continue to Step 2.0

Else if $rank(F_k) = 1$:

Continue to Step 2.1

Else

$\det(F_k) \neq 0$, $\varepsilon_L(s) = \varepsilon_L(s)_k$

Step 2.0: Increment k. Row 3 of F_{k-1} could be expressed as a linear combination of rows 1 and 2. Find a constants, $v_{1,k}$ and $v_{2,k}$ such that $F_{3,\bullet,k-1} = v_{1,k}F_{1,\bullet,k-1} + v_{2,k}F_{2,\bullet,k-1}$. Set $\varepsilon_L(s)_{3,\bullet,k} = s^x\left(\varepsilon_L(s)_{3,\bullet,k-1} - v_{1,k}\begin{bmatrix}s^{n_1} & 0\end{bmatrix} - v_{2,k}\varepsilon_L(s)_{2,\bullet,k-1}\right)$, and increment x until $\lim_{s\to\infty}\varepsilon_L(s)_{3,\bullet,k}G_{3,\bullet}(s) = F_{3,\bullet,k}$ gives a finite solution. Set $F_{1,\bullet,k} = F_{1,\bullet,k-1}$, and $F_{2,\bullet,k} = F_{2,\bullet,k-1}$ to assemble F_k. If $\det(F_k)$ is non-zero, then the algorithm is finished. Otherwise, one needs to go back to step 1.2.

Increment k by 1.

Find $v_{1,k}$ and $v_{2,k}$ such that $F_{3,\bullet,k-1} = v_{1,k}F_{1,\bullet,k-1} + v_{2,k}F_{2,\bullet,k-1}$

$$\varepsilon_L(s)_{3,\bullet,k} = s^x\left(\varepsilon_L(s)_{3,\bullet,k-1} - v_{1,k}\begin{bmatrix}s^{n_1} & 0\end{bmatrix} - v_{2,k}\varepsilon_L(s)_{2,\bullet,k-1}\right)$$

Increment x until $\lim_{s\to\infty}\varepsilon_L(s)_{3,\bullet,k}G_{3,\bullet}(s) = F_{3,\bullet,k}$ is finite and non-zero.

If $\det(F_k)$ is zero:

Go to Step 1.2

Else:

$F_k = D_\Lambda$ and $\varepsilon_L(s) = \begin{bmatrix} s^{n_1} & 0 & 0 \\ \varepsilon_L(s)_{2,1,k-1} & \varepsilon_L(s)_{2,2,k-1} & \varepsilon_L(s)_{2,3,k-1} \\ \varepsilon_L(s)_{3,1,k} & \varepsilon_L(s)_{3,2,k} & \varepsilon_L(s)_{3,3,k} \end{bmatrix}$, algorithm is done

Step 2.1: Increment k. Row 3 of F_{k-1} can be expressed as a constant, $v_{1,k}$, multiplied by row 1. Also, row 2 of F_{k-1} can be expressed as a constant, $v_{12,k}$, multiplied by row1. $F_{3,\bullet,k-1} = v_{1,k}F_{3,\bullet,k-1}$ and $F_{2,\bullet,k-1} = v_{12,k}F_{1,\bullet,k-1}$. Set $\varepsilon_L(s)_{3,\bullet,k} = s^x\left(\varepsilon_L(s)_{3,\bullet,k-1} - v_{1,k}\begin{bmatrix}s^{n_1} & 0\end{bmatrix}\right)$ and $\varepsilon_L(s)_{2,\bullet,k} = s^y\left(\varepsilon_L(s)_{2,\bullet,k-1} - v_{12,k}\begin{bmatrix}s^{n_1} & 0\end{bmatrix}\right)$. Then increment x and y until both $\lim_{s\to\infty}\varepsilon_L(s)_{3,\bullet,k}G_{3,\bullet}(s) = F_{3,\bullet,k}$ and $\lim_{s\to\infty}\varepsilon_L(s)_{2,\bullet,k}G_{2,\bullet}(s) = F_{2,\bullet,k}$ give a finite solution.

Set $F_{1,\bullet,k} = F_{1,\bullet,k-1}$, and assemble F_k. If $\det(F_k)$ is non-zero, then the algorithm is finished. Otherwise, one needs to go back to step 1.2.

Increment *k* by 1.

Find $v_{1,k}$ and $v_{12,k}$ such that $F_{3,\bullet,k-1} = v_{1,k} F_{1,\bullet,k-1}$ and $F_{2,\bullet,k-1} = v_{12,k} F_{1,\bullet,k-1}$

Set $\varepsilon_L(s)_{3,\bullet,k} = s^x \left(\varepsilon_L(s)_{3,\bullet,k-1} - v_{1,k} \begin{bmatrix} s^{n_1} & 0 \end{bmatrix}\right)$ and

$\varepsilon_L(s)_{2,\bullet,k} = s^y \left(\varepsilon_L(s)_{2,\bullet,k-1} - v_{12,k} \begin{bmatrix} s^{n_1} & 0 \end{bmatrix}\right)$

Then increment *x* and *y* until both $\lim_{s\to\infty} \varepsilon_L(s)_{3,\bullet,k} G_{3,\bullet}(s) = F_{3,\bullet,k}$ and

$\lim_{s\to\infty} \varepsilon_L(s)_{2,\bullet,k} G_{2,\bullet}(s) = F_{2,\bullet,k}$ give a finite solution.

If $\det(F_k)$ is zero:

Go to Step 1.2

Else:

$$F_k = D_\Lambda \text{ and } \varepsilon_L(s) = \begin{bmatrix} s^{n_1} & 0 & 0 \\ \varepsilon_L(s)_{2,1,k} & \varepsilon_L(s)_{2,2,k} & \varepsilon_L(s)_{2,3,k} \\ \varepsilon_L(s)_{3,1,k} & \varepsilon_L(s)_{3,2,k} & \varepsilon_L(s)_{3,3,k} \end{bmatrix}, \text{ algorithm is done}$$

Example 6.4:

Step 1.0:

$$G(s) = \begin{bmatrix} s+3 & -1 & -1 \\ 0 & 2(s+4) & 0 \\ 2(s+4) & 6(s+3) & 5 \end{bmatrix} \frac{1}{(s+2)(s+1)}$$

$$\varepsilon_L(s)_0 = diag(s \quad s \quad s) \quad F_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 2 & 6 & 0 \end{bmatrix}$$

$$\det(F_0) = 0$$

Step 1.2:

$Rank(F_0) = 2$, go to step 2.0

Step 2.0:

$k=1$

$F_{3,\bullet,0}=v_{1,1}F_{1,\bullet,0}+v_{2,1}F_{2,\bullet,0}=2[1\quad 0\quad 0]+3[0\quad 2\quad 0]$

$\varepsilon_L(s)_{3,\bullet,1}=s^x([0\quad 0\quad s]-2[s\quad 0\quad 0]-3[0\quad s\quad 0])\quad x=1$

$$\varepsilon_L(s)_1=\begin{bmatrix} s & 0 & 0 \\ 0 & s & 0 \\ -2s^2 & -3s^2 & s^2 \end{bmatrix}\quad F_1=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 2 & -4 & 7 \end{bmatrix}$$

$\varepsilon_L(s)=\varepsilon_L(s)_1\quad D_\Lambda=F_1$

Example 6.5:

Step 1.0:

$$G(s)=\begin{bmatrix} s^3 & -s & 2s \\ 2s^3 & 1 & 1 \\ 3s^3 & -1 & 1 \end{bmatrix}\frac{1}{(s+1)(s+2)(s+3)(s+4)}$$

$$\varepsilon_L(s)_0=diag(s\quad s\quad s)\quad F_0=\begin{bmatrix} 1 & 0 & 0 \\ 2 & 0 & 0 \\ 3 & 0 & 0 \end{bmatrix}$$

$det(F_0)=0$

Step 1.2:

$Rank(F_0)=1$, go to step 2.1

Step 2.1:

$k=1$

$F_{3,\bullet,0}=v_{1,1}F_{1,\bullet,0}=3[1\quad 0\quad 0]$

$F_{2,\bullet,0}=v_{12,1}F_{1,\bullet,0}=2[1\quad 0\quad 0]$

$\varepsilon_L(s)_{3,\bullet,1}=s^x([0\quad 0\quad s]-3[s\quad 0\quad 0])\quad x=2$

$\varepsilon_L(s)_{2,\bullet,1}=s^y([0\quad s\quad 0]-2[s\quad 0\quad 0])\quad y=2$

$$\varepsilon_L(s)_1=\begin{bmatrix} s & 0 & 0 \\ -2s^3 & s^3 & 0 \\ -3s^3 & 0 & s^3 \end{bmatrix}\quad F_1=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & -4 \\ 0 & 3 & -6 \end{bmatrix}$$

Go back to Step 1.2

Step 1.2:

$Rank(F_1)=2$, go to step 2.1

Step 2.1:

$$k=2$$

$$F_{3,\bullet,1}=v_{1,2}F_{1,\bullet,1}+v_{2,2}F_{2,\bullet,1}=0[1 \quad 0 \quad 0]+1.5[0 \quad 2 \quad -4]$$

$$\varepsilon_L(s)_{3,\bullet,2}=s^x([-3s^3 \quad 0 \quad s^3]-1.5[-2s^3 \quad s^3 \quad 0]) \quad x=1$$

$$\varepsilon_L(s)_2=\begin{bmatrix} s & 0 & 0 \\ -2s^3 & s^3 & 0 \\ 0 & -1.5s^4 & s^4 \end{bmatrix} \quad F_2=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & -4 \\ 0 & -2.5 & -0.5 \end{bmatrix}$$

$$\varepsilon_L(s)=\varepsilon_L(s)_2 \quad D_\Lambda=F_2$$

Example 6.6:

Step 2.1:

$$G(s)=\begin{bmatrix} 0.1z(z-0.85) & -0.2z(z-0.95) & 0.1z \\ 0.1z(z-0.95) & 0.05z(z-0.85) & -0.1z \\ 0 & 0 & 0.1z \end{bmatrix}\frac{1}{(z-0.8)(z-0.9)(z-1)}$$

$$\varepsilon_L(z)_0=diag(z \quad z \quad z^2) \quad F_0=\begin{bmatrix} 0.1 & -0.2 & 0 \\ 0.1 & 0.05 & 0 \\ 0 & 0 & 0.1 \end{bmatrix}$$

$$det(F_0)\neq 0$$

6.1.2 "Modern" Unitary Left Interactor Matrices for Square Transfer functions

The algorithm to obtain the "classical" left interactor matrix is complex and the only function it serves is to make a biproper transfer function. Not surprisingly, an easy two-step solution exists [6.5]. We know that the interactor has the following property for a square system in eq. 6.4.

$$\lim_{s\to\infty}\varepsilon_L(s)G(s)=D_\Lambda \quad D_\Lambda<\infty \quad det(D_\Lambda)\neq 0 \tag{6.4}$$

Suppose that we represented the interactor as the sum of *m x m* matrices corresponding to the number of differentiations, *s*, s^2, etc. as in eq. 6.5.

$$\varepsilon_L(s)=N_o+N_1s+N_2s^2+\ldots+N_ws^w \tag{6.5}$$

Expanding out the limit leads to the following expression in eq. 6.6.

$$\lim_{s\to\infty} \varepsilon_L(s)G(s) = (N_o D) + (N_1 CB + N_1 D(\infty)) + \left(N_2 CAB + N_2 CB(\infty) + N_2 D(\infty)^2\right) + \left(N_w CA^{w-1}B + N_w CA^{w-2}B(\infty) + N_w CA^{w-3}B(\infty)^2 + \ldots + N_w CB(\infty)^{w-1} + N_w D(\infty)^w\right) \tag{6.6}$$

Now, we regroup the terms in eq. 6.6 according to the finite terms first and the infinite terms later in eq. 6.7.

$$\lim_{s\to\infty} \varepsilon_L(s)G(s) = \left(N_o D + N_1 CB + N_2 CAB + \ldots + N_w CA^{w-1}B\right) + \left(N_1 D(\infty) + N_2 CB(\infty) + \ldots + N_w CA^{w-2}B(\infty)\right) + \left(N_2 D(\infty)^2 + \ldots + N_w CA^{w-3}B(\infty)^2\right) + \ldots + \left(N_{w-1} D(\infty)^{w-1} + N_w CB(\infty)^{w-1}\right) + N_w D(\infty)^w \tag{6.7}$$

Because the interactor must yield a finite result, all of the infinite terms must sum to zero. Also, the finite result must be equal to D_Λ. This forms a set of linear algebraic equations as in eq. 6.8.

$$\begin{aligned} &N_w D = 0 \\ &N_{w-1} D + N_w CB = 0 \\ &N_{w-2} D + N_{w-1} CB + N_w CAB = 0 \\ &\vdots \\ &N_o D + N_1 CB + N_2 CAB + \ldots + N_w CA^{w-1}B = D_\Lambda \end{aligned} \tag{6.8}$$

Now for ease of calculation, we can arbitrarily set $D_\Lambda = I$, and we define the markov parameters as M_k as the following matrix below in eq. 6.9.

$$M_k = \begin{bmatrix} D & 0_m & 0_m & \ldots & 0_m \\ CB & D & 0_m & \ldots & 0_m \\ CAB & CB & D & \ldots & 0_m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{k-1}B & CA^{k-2}B & CA^{k-3}B & \ldots & D \end{bmatrix}$$
$$W_k = \begin{bmatrix} I_m & \ldots & 0_m \end{bmatrix} \tag{6.9}$$

Then we increment the integer *k* until eq. 6.10 is satisfied. The smallest *k* which satisfies eq. 6.10, will be defined as *w*. This ensures that a solution will exist.

$$w \equiv \min_{k>0} \left(rank \begin{bmatrix} M_k \\ W_k \end{bmatrix} = rank[M_k] \right) \tag{6.10}$$

Then define *N* as the *m x (mw+m)* solution to the eq. 6.11.

$$NM_w = W_w \tag{6.11}$$

Using pseudo-inverse of M_w, labeled as M_w^+, one can arrive at a solution for N as in eq. 6.12.

$$N = W_w M_w^+ \tag{6.12}$$

Then partition N into $w+1$ submatrices of dimension m x m as in eq. 6.13.

$$N = \begin{bmatrix} N_0 & N_1 & \dots & N_w \end{bmatrix} \quad N_i \in \Re^{m \times m} \tag{6.13}$$

A suitable left interactor matrix can then be formed from the partitioned N_i as follows in eq. 6.14.

$$\varepsilon_L(s) = N_0 + N_1 s + N_2 s^2 + \dots + N_w s^w \tag{6.14}$$

If this were to occur in the digital domain, one would simply use z in place of s as in eq. 6.15.

$$\varepsilon_L(z) = N_o + N_1 z + N_2 z^2 + \dots + N_w z^w \tag{6.15}$$

Example 6.7:

$$G(s) = \begin{bmatrix} \frac{2(s+2)}{(s+1)^2} & \frac{0.25(s-0.25)}{(s+1)^2} \\ \frac{-0.1}{(s+1)^2} & \frac{1}{(s+1)^2} \end{bmatrix} \quad A = \begin{bmatrix} -1 & 4 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 4 \\ 0 & 0 & 0 & -1 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 2 & 0 \\ 0 & 0 \\ 0 & 0.5 \end{bmatrix}$$

$$G(s) = C(sI - A)^{-1} B \qquad C = \begin{bmatrix} 0.25 & 1 & -0.1250 & 0.4 \\ -0.0125 & 0 & 0.5 & 0 \end{bmatrix}$$

$$w = 2$$

$$M_2 = \begin{bmatrix} 0_m & 0_m & 0_m \\ CB & 0_m & 0_m \\ CAB & CB & 0_m \end{bmatrix} \quad CB = \begin{bmatrix} 2 & 0.2 \\ 0 & 0 \end{bmatrix} \quad CAB = \begin{bmatrix} 0 & -0.45 \\ 0.1 & 1.0 \end{bmatrix}$$

$$W_2 = \begin{bmatrix} I_m & 0_m & 0_m \end{bmatrix}$$

$$N = W_2 M_2^+ \quad N_0 = 0_m \quad N_1 = \begin{bmatrix} 0.495 & 0 \\ 0.0495 & 0 \end{bmatrix} \quad N_2 \approx \begin{bmatrix} 0 & 0.09901 \\ 0 & 0.9901 \end{bmatrix}$$

$$\varepsilon_L(s) = \begin{bmatrix} 0.495s & 0.09901s^2 \\ 0.0495s & 0.9901s^2 \end{bmatrix} \quad D_\Lambda = I$$

Recall that from example 6.1 that a suitable interactor for this system is just

$\varepsilon_L(s) = diag(s \quad s^2)$. This method can produce interactors which are cosmetically unattractive.

Interactors may also be numerically unattractive as well. In some cases, round-off error can produce improper realizations of $\Lambda_G(s)$.

Also observe that the high frequency gain, $D_\Lambda = I$, because we have specified that when we made $W_k = [I_m \quad \ldots \quad 0_m]$. Hence, this is called an identity interactor.

6.1.3 Generating Inverse Dynamics

By definition, a biproper transfer function can be inverted and has a square and non-zero D matrix. Given a square and biproper transfer function in state space form as $\Lambda_G(s) = C_\Lambda(sI - A_\Lambda)^{-1}B_\Lambda + D_\Lambda$ one can derive the state space for the inverse dynamics as follows.

$$\dot{x} = A_\Lambda x + B_\Lambda u \quad y = C_\Lambda x + D_\Lambda u \tag{6.16}$$

Now the inverse dynamics will have *u* as an output, so we switch eq. 6.16 around in eq. 6.17.

$$u = (D_\Lambda)^{-1}(y - C_\Lambda x) \tag{6.17}$$

We insert the input relationship, *u*, of eq. 6.17 into eq. 6.16 and arrive at the state space of the inverse dynamics in eq. 6.18.

$$\begin{aligned} \dot{x} &= A_\Lambda x - B_\Lambda (D_\Lambda)^{-1} C_\Lambda x + B_\Lambda (D_\Lambda)^{-1} y \\ u &= -(D_\Lambda)^{-1} C_\Lambda x + (D_\Lambda)^{-1} y \end{aligned} \tag{6.18}$$

For completeness, we re-write eq. 6.18 in the laplace domain in eq. 6.19.

$$\Lambda_G^{-1}(s) = -D_\Lambda^{-1}C_\Lambda\left(sI - A_\Lambda + B_\Lambda D_\Lambda^{-1}C_\Lambda\right)^{-1}B_\Lambda D_\Lambda^{-1} y + D_\Lambda^{-1}u \tag{6.19}$$

For quick inversion of a 2x2, one can use the simple inversion formula below in eq 6.20.

$$\Lambda_G^{-1}(s) = \frac{1}{\Lambda_{G11}(s)\Lambda_{G22}(s) - \Lambda_{G12}(s)\Lambda_{G21}(s)} \begin{bmatrix} \Lambda_{G22}(s) & -\Lambda_{G12}(s) \\ -\Lambda_{G21}(s) & \Lambda_{G11}(s) \end{bmatrix} \tag{6.20}$$

Procedurally, inversion is exactly the same in the digital domain as it is in the continuous domain, only one would be using *z* in place of *s*.

Very often in engineering applications, interactor matrices will be diagonal and uniform, and they will be relative degree one or two (velocity-like or position-like) outputs. Quick formulae for these inverse dynamics are listed below in eq. 6.21 and 6.22.

$$G(s)=(A,B,C,0)\quad sG(s)=(A,B,CA,CB)$$
$$(sG(s))^{-1}=\Lambda_G^{-1}(s)=\left(A-B(CB)^{-1}CA,B(CB)^{-1},-(CB)^{-1}CA,(CB)^{-1}\right) \tag{6.21}$$

$$G(s)=(A,B,C,0)\quad CB=0\quad s^2G(s)=\left(A,B,CA^2,CAB\right)$$
$$\left(s^2G(s)\right)^{-1}=\Lambda_G^{-1}(s)=\left(A-B(CAB)^{-1}CA,B(CAB)^{-1},-(CAB)^{-1}CA,(CAB)^{-1}\right) \tag{6.22}$$

6.2 Interactor Matrices for Fat Systems

6.2.1 Left Interactor Matrices for Fat Systems

For certain tasks, namely squaring down which is discussed in Chapter 2, one may want to turn the plant, $G(s)$, into a proper system, $\Lambda_G(s)$. $G(s)$ has *m* inputs and *p* outputs, $m > p$.

One would not pursue inversion or compensator design until after the plant is squared down.

To get a right interactor matrix, $\varepsilon_R(s)$, for a fat system with more inputs, *m*, than outputs, *p*, one would use the definition in eq. 6.23 below to get it.

$$\lim_{s\to\infty}\varepsilon_L(s)G(s)=\lim_{s\to\infty}\Lambda_G(s)=D_\Lambda\quad rank(D_\Lambda)=p\quad and\quad D_\Lambda\neq\infty \tag{6.23}$$

Example 6.8:

$$G(s)=\begin{bmatrix}\frac{2(s+1)}{(s+2)^2} & \frac{0.2(s+5)}{(s+2)^2} & \frac{5s}{(s+2)^2}\\ \frac{-0.1}{(s+2)^2} & \frac{1}{(s+2)^2} & \frac{2}{(s+2)^2}\end{bmatrix}\quad \varepsilon_L(s)=\begin{bmatrix}s & 0\\ 0 & s^2\end{bmatrix}$$

$$\lim_{s\to\infty}\varepsilon_L(s)G(s)=\lim_{s\to\infty}\Lambda_G(s)=\begin{bmatrix}2 & 0.2 & 5\\ -0.1 & 1 & 1\end{bmatrix}=D_\Lambda$$

$$rank\left(\begin{bmatrix}2 & 0.2 & 5\\ -0.1 & 1 & 1\end{bmatrix}\right)=2\quad and\quad D_\Lambda\neq\infty$$

6.2.2 "Modern" Left Interactor Matrices for Fat Systems

As before, we define the markov parameter matrix, M_k, which is different from section 6.1.2 only in that it will be a fat matrix with more columns than rows.

$$M_k = \begin{bmatrix} D & 0_{p\times m} & 0_{p\times m} & \cdots & 0_{p\times m} \\ CB & D & 0_{p\times m} & \cdots & 0_{p\times m} \\ CAB & CB & D & \cdots & 0_{p\times m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{k-1}B & CA^{k-2}B & CA^{k-3}B & \cdots & D \end{bmatrix}$$
$$IO_k = \begin{bmatrix} I_{p\times p} & 0_{p\times(m-p)} & \cdots & 0_{p\times m} \end{bmatrix} \tag{6.24}$$

Then increment the integer k until eq. 6.25 is satisfied. The smallest k which satisfies eq. 6.25, will be defined as w.

$$w \equiv \min_{k>0}\left(rank\begin{bmatrix} M_k \\ IO_k \end{bmatrix} = rank[M_k] \right) \tag{6.25}$$

Finding left interactors for fat systems (i.e. more inputs than outputs $p < m$) is slightly more complicated than in the square case [6.6]. This is because we can no longer freely assign $D_\Lambda = I$ and use the identity interactor as we did in section 6.1.2. Although it seems odd, we need to figure out what D_Λ or W_w is before we solve for the interactor. From eq. 6.11, we know the following in eq. 6.26.

$$NM_w = W_w = \begin{bmatrix} D_\Lambda & 0_{p\times m} & \cdots & 0_{p\times m} \end{bmatrix} \tag{6.26}$$

With N undefined and D_Λ undefined, the problem is not solvable. However, if we use eq. 6.12, one can express N in terms of W_w or D_Λ, as in eq. 6.27.

$$N = W_w M_w^+ = \begin{bmatrix} D_\Lambda & 0_{p\times m} & \cdots & 0_{p\times m} \end{bmatrix} M_w^+ \tag{6.27}$$

Then, we substitute eq. 6.27 into eq. 6.26 to arrive at eq. 6.28.

$$W_w M_w^+ M_w - W_w = 0$$
$$D_\Lambda\left(\begin{bmatrix} I_m & 0_m & \cdots & 0_m \end{bmatrix} M_w^+ M_w - \begin{bmatrix} I_m & 0_m & \cdots & 0_m \end{bmatrix} \right) = 0 \tag{6.28}$$

From eq. 6.28, we need to solve for the left null space of the $\left(\begin{bmatrix} I_m & 0_m & \cdots & 0_m \end{bmatrix} M_w^+ M_w - \begin{bmatrix} I_m & 0_m & \cdots & 0_m \end{bmatrix} \right)$ matrix in eq. 6.21. Since the null() subroutine in MATLAB only solves the right null space, we simply use the transpose operation to make things easily solvable as in eq. 6.29.

$$\left([I_m \quad 0_m \quad \ldots \quad 0_m]M_w^+ M_w - [I_m \quad 0_m \quad \ldots \quad 0_m]\right)^T D_\Lambda^T = 0 \tag{6.29}$$

With D_Λ and W_w defined, we can now solve for N as the m x $(mw+m)$ solution to the eq. 6.27, where $N = W_w M_w^+$. Then partition N into $w+1$ submatrices of dimension m x m as in eq. 6.30.

$$N = [N_0 \quad N_1 \quad \ldots \quad N_w] \quad N_i \in \Re^{m \times m} \tag{6.30}$$

A suitable left interactor matrix can then be formed from the partitioned N_i as follows in eq. 6.31.

$$\varepsilon_L(s) = N_0 + N_1 s + N_2 s^2 + \ldots + N_w s^w \tag{6.31}$$

Example 6.9:

MATLAB code to form Interactor Matrix for Fat System

```
%Fat Interactor Matrix Solution
m=3; %three inputs
p=2; %two outputs

sys=[zpk([-1],[-2 -2],2) zpk([-5],[-2 -2],.2)...
    zpk([0],[-2 -2],5);
    zpk([],[-2 -2],-0.1) zpk([],[-2 -2],1)...
    zpk([],[-2 -2],2)];
sys=minreal(sys); %make system minimal
[A,B,C,D]=ssdata(sys); %get state space realization
Mw=[D zeros(2,3) zeros(2,3);C*B D zeros(2,3);
    C*A*B C*B D]; %Markov Parameters
Mwplus=pinv(Mw); %pseudo-inverse
dummy=[eye(m) zeros(m,2*m)]*Mwplus*Mw-...
    [eye(m) zeros(m,2*m)];
Dgamt=null(dummy'); %solve for left-nullspace
Dgam=Dgamt'; %left interactor
N=[Dgam zeros(2,6)]*Mwplus;

s=tf('s');
N(1:2,5)=0*N(1:2,5); %correct for machine error
left_int=N(1:2,1:2)+N(1:2,3:4)*s+N(1:2,5:6)*s^2;
gamma_sys = minreal(left_int*sys,1e-4); %clean up
```

$$G(s)=\begin{bmatrix} \dfrac{2(s+1)}{(s+2)^2} & \dfrac{0.2(s+5)}{(s+2)^2} & \dfrac{5s}{(s+2)^2} \\ \dfrac{-0.1}{(s+2)^2} & \dfrac{1}{(s+2)^2} & \dfrac{2}{(s+2)^2} \end{bmatrix} \quad A=\begin{bmatrix} -2 & 2 & 0 & 0 \\ 0 & -2 & 0 & 0 \\ 0 & 0 & -2 & 2 \\ 0 & 0 & 0 & -2 \end{bmatrix}$$

$$B=\begin{bmatrix} 1 & 0.1 & 2.5 \\ -0.5 & 0.15 & -2.5 \\ 0 & 0 & 0 \\ -0.05 & 0.5 & 1 \end{bmatrix} \quad C=\begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$M_2=\begin{bmatrix} 0_{p\times m} & 0_{p\times m} & 0_{p\times m} \\ CB & 0_{p\times m} & 0_{p\times m} \\ CAB & CB & 0_{p\times m} \end{bmatrix} \quad CB=\begin{bmatrix} 2 & 0.2 & 5 \\ 0 & 0 & 0 \end{bmatrix} \quad CAB=\begin{bmatrix} -6 & 0.2 & -20 \\ -0.1 & 1 & 2 \end{bmatrix}$$

$$D_\Lambda^T=null\left(\left[\begin{bmatrix} I_3 & 0_3 & 0_3 \end{bmatrix}\left(M_2^+\right)M_2-\begin{bmatrix} I_3 & 0_3 & 0_3 \end{bmatrix}\right]^T\right)$$

$$N=\begin{bmatrix} D_\Lambda & 0_{2\times 3} & 0_{2\times 3} \end{bmatrix}M_2^+$$

$$N=\begin{bmatrix} 0 & 0 & 0.3201 & 0 & 0 & -0.7568 \\ 0 & 0 & -0.0875 & 0 & 0 & -0.2563 \end{bmatrix}$$

$$\varepsilon_L(s)=\begin{bmatrix} 0.3201s & -0.7568s^2 \\ -0.08751s & -0.2564s^2 \end{bmatrix}$$

6.3 Diagonal Decoupling of Stable, Minimum Phase Systems

It is desired that a diagonal loop shape, $L(s)$, is achieved by a design of a compensator, $C(s)$.

$$G(s)C(s)=L(s) \tag{6.32}$$

In the previous section, the left interactor matrix, $\varepsilon_L(s)$, was introduced. The purpose of the interactor matrix is to make the plant, $G(s)$, into a biproper transfer function, $\Lambda_G(s)$. One simply has to use the following relation, $\Lambda_G(s)=\varepsilon_L(s)G(s)$, to obtain $\Lambda_G(s)$.

After a biproper $\Lambda_G(s)$ is obtained by a feasible choice of $\varepsilon_L(s)$, one simply has to invert and multiply by the modified loopshape, $\varepsilon_L(s)L(s)$. This design procedure is summarized below.

First, we multiply eq. 6.32 through by the interactor matrix.

$$\varepsilon_L(s)G(s)C(s)=\varepsilon_L(s)L(s) \tag{6.33}$$

We define $\Lambda_G(s)$ as a biproper transfer function (square and invertible, i.e. an invertible D matrix).

$$\Lambda_G(s)=\varepsilon_L(s)G(s) \tag{6.34}$$

Then, to obtain the controller, one must simply invert and use the relation below in eq. 6.35.

$$C(s)=\Lambda_G^{-1}(s)\varepsilon_L(s)L(s) \tag{6.35}$$

This is a very simple and quick way to get a compensator that meets the frequency domain specifications defined by $L(s)$.

Example 6.10

The suspension system in Fig. 6.3 has the following state space parameterization below and the right coprime form listed below.

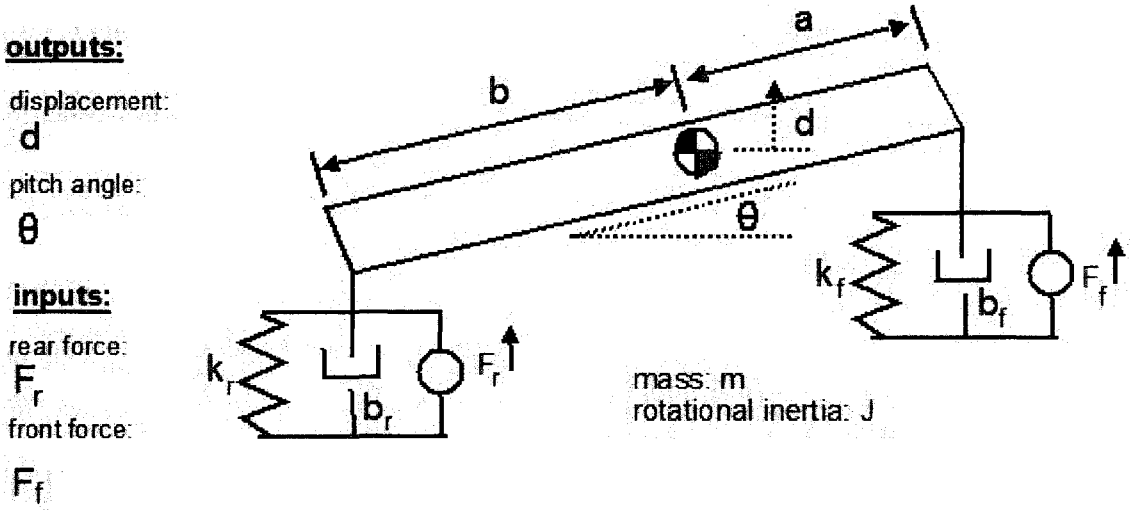


Fig. 6.3. Suspension system with front and rear stiffness, k_f and k_r, and dashpots with damping constants, b_f and b_r, Outputs are displacement d of the center of gravity and pitch angle, θ. Inputs are front force and rear force, F_f and F_r. The rigid body has mass m and rotational inertia J.

$$a_{31}=\frac{-(k_f+k_r)}{m} \quad a_{32}=\frac{(-ak_f+bk_r)}{m} \quad a_{33}=\frac{-(b_f+b_r)}{m} \quad a_{34}=\frac{(-ab_f+bb_r)}{m}$$

$$a_{41}=\frac{1}{J}\left(\frac{-k_f}{a}+\frac{k_r}{b}\right) \quad a_{42}=\frac{-(k_r+k_f)}{J} \quad a_{43}=\frac{1}{J}\left(\frac{-b_f}{a}+\frac{b_r}{b}\right) \quad a_{44}=\frac{-(k_r+k_f)}{J}$$

$$b_{31}=1/m \quad b_{32}=1/m \quad b_{41}=a/J \quad b_{42}=-b/J$$

$$A=\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \quad B=\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ b_{31} & b_{32} \\ b_{41} & b_{42} \end{bmatrix}$$

$$C=\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$x=\begin{bmatrix} d & \theta & \dot{d} & \dot{\theta} \end{bmatrix}^T \quad u=\begin{bmatrix} F_f & F_r \end{bmatrix}^T$$

$$G(s)=\begin{bmatrix} b_{31}(s^2-a_{44}s-a_{42})+b_{41}(a_{34}s+a_{32}) & b_{32}(s^2-a_{44}s-a_{42})+b_{42}(a_{34}s+a_{32}) \\ b_{41}(s^2-a_{33}s-a_{31})+b_{31}(a_{43}s+a_{41}) & b_{42}(s^2-a_{33}s-a_{31})+b_{32}(a_{43}s+a_{41}) \end{bmatrix}\frac{1}{\Delta(s)}$$

$$\Delta(s)=s^4-(a_{44}+a_{33})s^3-(a_{34}a_{43}-a_{33}a_{44}+a_{42}-a_{31})s^2-(a_{32}a_{43}-a_{42}a_{33}+a_{41}a_{34}\cdots$$
$$-a_{44}a_{31})s+a_{31}a_{42}-a_{41}a_{32}$$

There are no transmission zeros of the suspension system with θ and d as outputs (but there would be 2 transmission zeros at s = 0 if $\dot{\theta}$ and $\dot{d}$ were outputs). One can conclude that this is a stable

and minimum phase plant and it should be straightforward to invert. We will use the following parameters listed below.

$$m = 4kg \quad J = 4kg \cdot m^2 \quad a = 1m \quad b = 1m$$
$$k_f = 100N \cdot m^{-1} \quad k_r = 100N \cdot m^{-1}$$
$$b_f = 25N \cdot s \cdot m^{-1} \quad b_r = 25N \cdot s \cdot m^{-1}$$

We will also use a loop shape which crosses over arbitrarily at 20 rad./sec. shown in Fig. 6.4.

$$L(s) = \begin{bmatrix} \frac{1200}{s(s+60)} & 0 \\ 0 & \frac{1200}{s(s+60)} \end{bmatrix}$$

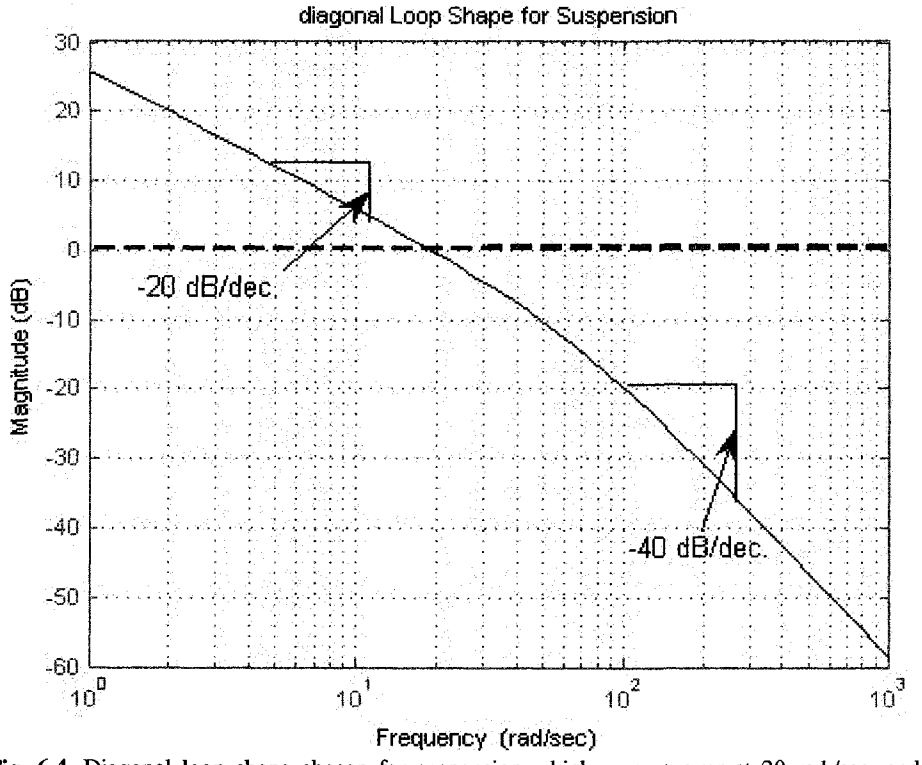


Fig. 6.4. Diagonal loop shape chosen for suspension which crosses over at 20 rad./sec. and has a bandwidth at approximately the same 20 rad./sec.

The left interactor matrix for this system is listed below.

$$\varepsilon_L(s) = \begin{bmatrix} s^2 & 0 \\ 0 & s^2 \end{bmatrix}$$

This gives the following expression for $\Lambda_G(s)$.

$$\Lambda_G(s)=\begin{bmatrix}0.25 & 0.25\\ 0.25 & -0.25\end{bmatrix}\frac{s^2}{s^2+12.5s+50}$$

And it is easy to invert this expression as follows below.

$$\Lambda_G^{-1}(s)=\begin{bmatrix}2 & 2\\ 2 & -2\end{bmatrix}\frac{s^2+12.5s+50}{s^2}$$

The compensator is easily computed as follows.

$$\begin{aligned}C(s)&=\Lambda_G^{-1}(s)\varepsilon_L(s)L(s)\\ &=1200\begin{bmatrix}2 & 2\\ 2 & -2\end{bmatrix}\frac{s^2+12.5s+50}{s(s+60)}\end{aligned}$$

Example 6.10 – MATLAB code for Suspension Compensator Design via Inverse Dynamics

```
J=4;m=4;a=1;b=1;kf=100;bf=25;
kr=100;br=25;

a31=-(kf+kr)/m;a32=(-a*kf+b*kr)/m;
a33=-(bf+br)/m;a34=(-a*bf+b*br)/m;
a41=-kf/(J*a)+kr/(J*b);
a42=-(kf+kr)/J;
a43=-bf/(J*a)+br/(J*b);
a44=-(bf+br)/J;
b31=1/m;b32=1/m;
b41=a/J;b42=-b/J;

A=[zeros(2) eye(2);a31 a32 a33 a34;
    a41 a42 a43 a44];
B=[0 0;0 0; b31 b32; b41 b42];
C=[eye(2) zeros(2)];

sys=ss(A,B,C,zeros(2));

%form loop shape
%cross-over at 20 rad./sec.
L=zpk([],[0 -60],60*20)*eye(2);

%form left interactor
s=tf('s');
left_int=s^2*eye(2);

%get gamma_g and invert
gamma_g=minreal(left_int*sys);
inv_gamma_g=inv(gamma_g);
Comp=inv_gamma_g*left_int*L;
Comp=minreal(Comp) %clean up
```

One should note that this compensator is very much like a PID compensator (with a filter, of course). Also, one will find in this case that $\Lambda_G^{-1}(s)$ could also be computed by the formula below.

$$\Lambda_G^{-1}(s)=\left(s^2G(s)\right)^{-1}=\begin{bmatrix}A^{inv}=A-B(CAB)^{-1}CA & B^{inv}=B(CAB)^{-1}\\ C^{inv}=-(CAB)^{-1}CA & D^{inv}=(CAB)^{-1}\end{bmatrix}$$

Example 6.11

It is important to keep in mind that these decoupling techniques are basically doing pole-zero cancellations. Therefore, it can sometimes help to first use a regulator in order to move poles away from the $j\omega$ axis before decoupling.

Figure 6.5 shows a block diagram of an NT-33 aircraft at Mach 0.2, Sea-Level (SL), [2.21] with

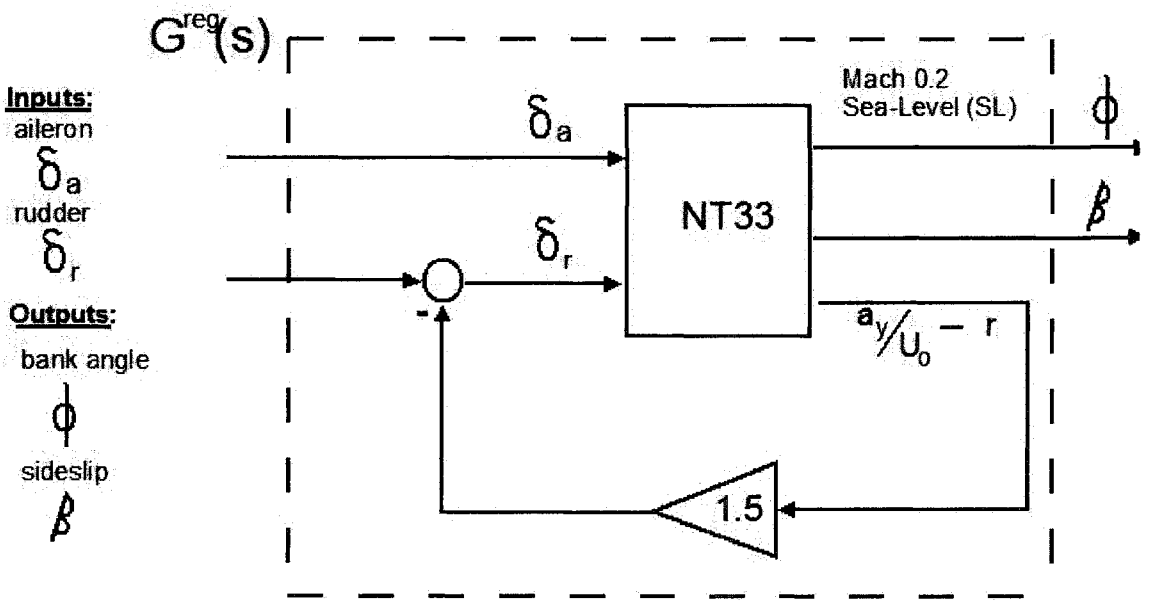


Fig. 6.5. Block diagram NT-33 with regulator. The forward speed is U_o, lateral acceleration is a_y and r is yaw rate. The dutch roll mode for this aircraft is very close to the $j\omega$ axis, so the simple regulator sufficiently damps the dutch roll mode. This design will invert $G^{reg}(s)$, which is $G(s)$ with a regulator.

Greg(s) Mach 0.2, SL

$$A = \begin{bmatrix} -0.1197 & 0 & -0.9576 & 0.1352 \\ -5.4922 & -2.0300 & 0.6230 & 0.002536 \\ 0.4444 & -0.1160 & -1.9882 & 0.2516 \\ 0 & 1.0000 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0.02825 \\ 6.0100 & -0.01197 \\ 0.0286 & -1.187 \\ 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

State Space. Input vector is $u = [\delta_a \quad \delta_r]^T$ units are [rad. rad.]. The output vector is $y = [\phi \quad \beta]^T$, units are [rad. rad.] respectively.

a regulator that feeds back $a_y/U_o - r$ into the rudder to damp out a nearly unstable dutch roll mode.

A state space realization of $G^{reg}(s)$ is listed above. The loop shape for this $G^{reg}(s)$ is listed below and we will cross over aggressively at 7 rad./sec.

$$L(s) = \begin{bmatrix} \dfrac{210}{s(s+30)} & 0 \\ 0 & \dfrac{7}{s} \end{bmatrix}$$

A satisfactory left interactor matrix for this system is listed below.

$$\varepsilon_L(s)=\begin{bmatrix} s^2 & 0 \\ 0 & s \end{bmatrix}$$

Now instead of inverting $G(s)$, we will invert $G^{reg}(s)$ so that we don't put zeros so close to the $j\omega$ axis.

Following the same steps as before in example 6.10, one arrives at the following compensator.

$$C^{reg}(s)=\Lambda_{Greg}^{-1}(s)\varepsilon_L L(s)\approx\begin{bmatrix} \dfrac{34.9418(s+2.075)(s-.04356)}{s(s+30)} & \dfrac{0.49353(s+8.064)(s+68.81)}{s(s+42.24)} \\ \dfrac{33.8770(s+0.1196)(s-52.14)}{s(s+42.24)(s+30)} & \dfrac{247.7881(s+1.707)(s+0.4035)}{s(s+42.24)(s+30)} \end{bmatrix}$$

6.4 Diagonal Decoupling of Unstable, Minimum Phase Systems with $G^{reg}(s)$-$C^{reg}(s)$

As in example 6.11, a two step design was shown where a regulator moved lightly damped poles away from the $j\omega$ axis, then a decoupling compensator was designed. This procedure can also be extended to unstable poles as well. We summarize the control design procedure as follows below.

Step 1 – Design a stablilizing regulator, $H(s)$, for the plant, $G(s)$

Step 2 – Invert $G^{reg}(s)=(I+G(s)H(s))^{-1}G(s)$ to obtain the desired loop shape, $L(s)$, with $C^{reg}(s)$.

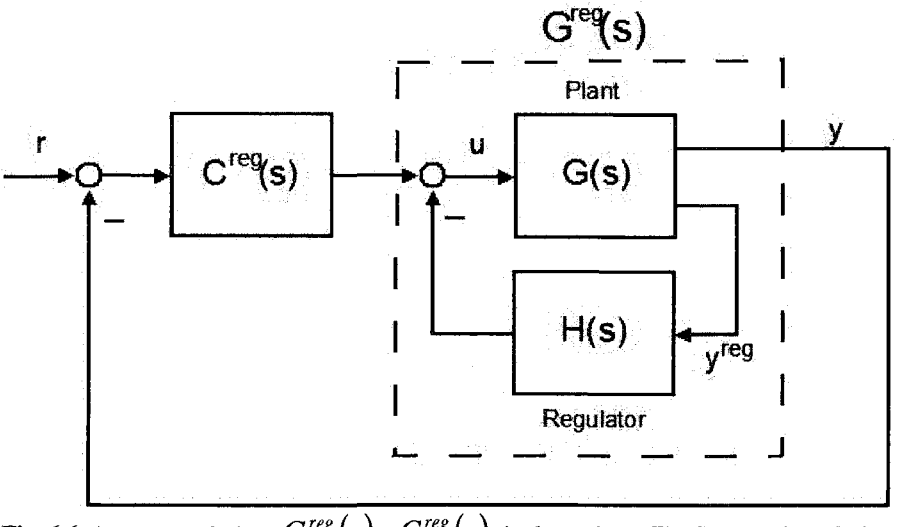


Fig. 6.6. A two-step design, $G^{reg}(s)-C^{reg}(s)$, is shown here. The first step is to design a regulator, *H(s)*, for the plant. The next step is to use a decoupling compensator $C^{reg}(s)$ to decouple the outputs which are to be controlled.

Figure 6.6 shows the $G^{reg}(s)-C^{reg}(s)$ feedback structure. One can use different outputs for the regulator design step, than in the decoupling compensator step. One could also use the same outputs for the regulator step, it does not really matter. The purpose of the regulator is to place poles in desirable locations. One can use classical control concepts for pole placement or one can use modern approaches. Refer to section 2.24 for state feedback and observer regulator strategies or sections 2.13 to 2.16 and chapters 4 and 5 for classical control approaches.

After the regulator design is done. The math behind this $C^{reg}(s)$ is the same as in the previous section because $G^{reg}(s)$ is now stable and minimum-phase. We decide upon a desired loop shape, $L(s)$.

$$G^{reg}(s)C^{reg}(s)=L(s) \tag{6.37}$$

Then we follow the same mathematical procedures as before in section 6.3, summarized in eq. 6.38.

$$\begin{aligned}&\varepsilon_L(s)G^{reg}(s)C^{reg}(s)=\varepsilon_L(s)L(s), \quad \Lambda^{-1}_{Greg}(s)=\left(\varepsilon_L(s)G^{reg}(s)\right)^{-1}\\ &C^{reg}(s)=\Lambda^{-1}_{Greg}(s)\varepsilon_L(s)L(s)\end{aligned} \tag{6.38}$$

Example 6.12

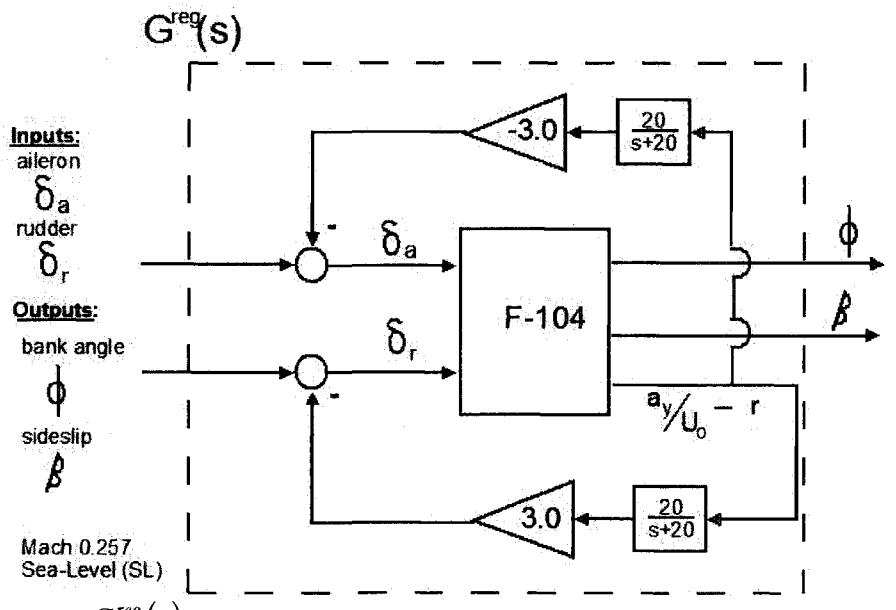


Fig. 6.7. $G^{reg}(s)$ for the F-104. We use the regulator designed in section 5.5.1.

The F-104 in its approach flight condition of Mach 0.257 at Sea-Level is open-loop unstable. Its dutch roll mode has negative damping. The unstable dutch roll mode has the roots of

$s^2 - 0.34s + 3.98$.

Luckily, with the outputs of ϕ and β, the plant has its transmission zeros in the LHP. Open loop data for the F-104 at this flight condition can be found in section 5.5.

We will use the regulator designed in section 5.5, which fed back $\dot{\beta}$, or $a_y/U_o - r$,

Greg(s) for F-104 Mach 0.257, SL

```
       -0.1780         0   -1.0000    0.1122   -0.2377
      -20.9000   -1.3800    1.1600         0   -4.4250
A=     2.6800   -0.0993   -0.1570         0    8.9175
          0     1.0000         0         0         0
       -1.4240         0   -8.0000    0.8976  -21.9020

          0    0.0317
     4.7600    5.3500
B=   0.2660   -0.9230
          0         0
          0    0.2536

C=    0     0     0     1     0
      1     0     0     0     0
```

State Space. Input vector is $u = [\delta_a \quad \delta_r]^T$ units are [rad. rad.]. The output vector is $y = [\phi \quad \beta]^T$, units are [rad. rad.] respectively. Here, $y^{reg} = \left[a_y/U_o - r \right]$.

into the rudder and aileron with a constant Rudder to Aileron Interconnect (RAI) of -1. The washed out regulator had a gain margin of 20dB and a phase margin of approximately 60 degrees.

We will cross over at 7 rad./sec. and 3 rad./sec. for ϕ and β, respectively.

$$L(s) = \begin{bmatrix} \dfrac{210}{s(s+30)} & 0 \\ 0 & \dfrac{3}{s} \end{bmatrix}$$

In this case, the regulator does not effect the interactor matrix.

$$\varepsilon_L(s) = \begin{bmatrix} s^2 & 0 \\ 0 & s \end{bmatrix}$$

With the interactor matrix defined, we can get the inverted dynamics of Greg shown below.

$$\Lambda_{Greg}^{-1}(s) \approx \begin{bmatrix} \dfrac{0.21(s+27.92)(s+2.74)}{s^2(s+38.77)} & \dfrac{-35.46(s+18.05)(s+3.92)(s-0.26)}{s(s+38.77)(s+20)} \\ \dfrac{1.77(s-2.58)(s+0.17)}{s^2(s+38.77)} & \dfrac{31.55(s+17.72)(s+3.22)(s+1.36)}{s(s+38.77)(s+20)} \end{bmatrix}$$

Our final compensator, $C^{reg}(s)$, results from eq. 6.37 and 6.38.

$$C^{reg}(s) = \Lambda_{Greg}^{-1}(s)\varepsilon_L L(s)$$

$$C^{reg}(s) \approx \begin{bmatrix} \dfrac{44.1176(s+27.92)(s+2.74)(s-0.01067)}{s(s+30)(s+38.77)} & \dfrac{-106.4(s+18.05)(s+3.92)(s-0.264)}{s(s+38.77)(s+20)} \\ \dfrac{370(s-2.58)}{s(s+30)(s+38.77)} & \dfrac{94.6(s+1.36)(s+3.22)(s+17.7)}{s(s+38.77)(s+20)} \end{bmatrix}$$

Example 6.12 – MATLAB code for decoupling F-104 with Greg-Creg

```
L=[zpk([],[0 -30],7*30) 0;0 zpk([],[0],7)];

s=tf('s');
left_int=[s^2 0;0 s];
gamma_greg=left_int*Greg;
inv_gamma_greg=minreal(inv(gamma_greg),1e-1);

Creg=inv_gamma_greg*left_int*L;
Creg=minreal(Creg,1e-1);  %clean up
```

6.5 Diagonal Decoupling of Unstable, Minimum Phase Systems with Q-Synthesis

It is possible to handle instability by first designing a stable regulator, and then inverting, as in section 6.3 and in example 6.12, and the author personally prefers the two step design. Nonetheless, this section will detail a strategy to design decoupling controllers with a single controller via the Q parameterization.

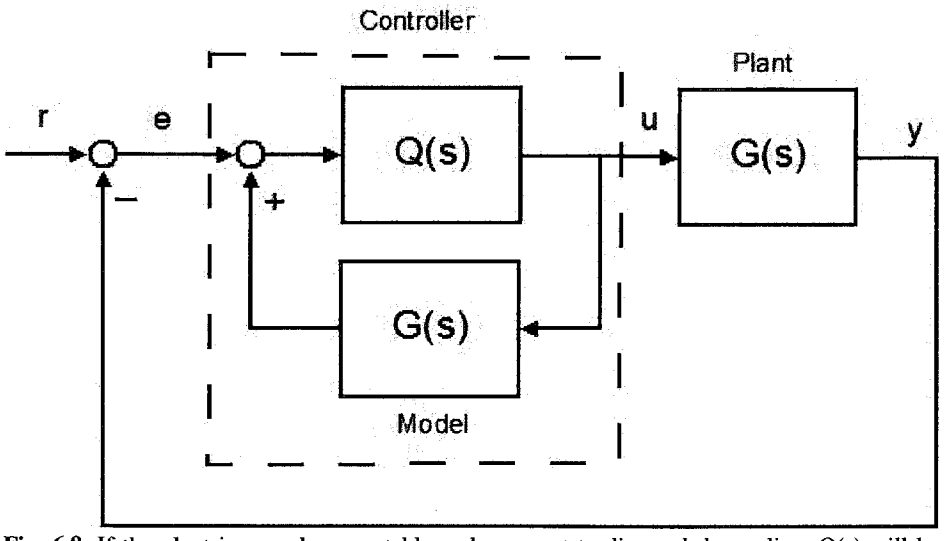


Fig. 6.8. If the plant is open-loop unstable and we want to diagonal decoupling, Q(s) will be non-minimum phase (unstable transmission zeros). In this case, we will have to make unstable-pole zero cancellations within the Q(s) and G(s) positive feedback loop. This can sometimes be a numerical challenge. A two-step design, $G^{reg}(s) - C^{reg}(s)$, is recommended.

Diagonal decoupling of unstable systems is arriving at an achievable loop-shape $L(s)$ or equivalent co-sensitivity, $T(s)$. $T(s)$ will have overshoot for an unstable system and $L(s)$ must contain unstable poles. Although it is not completely necessary to use the Q parameterization, it is a good way to quickly check whether or not $T(s)$ or $L(s)$ has been specified properly.

The Q parameterization from section 2.23 can be listed as follows in eq. 6.39.

$$T(s) = G(s)Q(s) \tag{6.39}$$

Recall also that $T(s) = (I + L(s))^{-1} L(s)$ from section 2.22. With Q defined in eq. 6.39, one can get the compensator, $C(s)$, from eq. 6.40.

$$C(s) = Q(s)(I - G(s)Q(s))^{-1} = (I - Q(s)G(s))^{-1} Q(s) \tag{6.40}$$

To synthesize Q, one can simply use the left interactor discussed in previous sections.

$$Q(s) = (\varepsilon_L(s)G(s))^{-1} \varepsilon_L(s)T(s) = \Lambda_G^{-1}(s)\varepsilon_L(s)T(s) \tag{6.41}$$

Stability of the closed loop can be met if and only if $Q(s)$ is stable. So, the usefulness of the Q parameterization is that it can be used to tell whether or not a specification of $T(s)$ is possible.

Given that $G(s)$ is unstable and minimum phase (all transmission zeros of $G(s)$ are stable), $G(s)$ can be stabilized by some compensator $C(s)$. We expect that $\Lambda_G^{-1}(s)$ to be stable, but with unstable transmission zeros. Thus, $Q(s)$ should also be stable, but should have unstable transmission zeros.

Example 6.13

The F-104 in its approach flight condition of Mach 0.257 at Sea-Level is open-loop unstable. Its dutch roll mode has negative damping. Luckily, with the outputs of ϕ and β, the plant is minimum phase. Precise decoupling is more difficult in this case because the unstable mode means that overshoot must be incorporated into the loop-shape. The unstable dutch roll mode has the roots of $s^2 - 0.34s + 3.98$ and we will use an aggressive loop shape crosses over at 10 rad./sec. Note that the loop shape has to include the unstable dutch roll mode.

$$L(s)=\begin{bmatrix} \dfrac{400(s+2)^2}{s(s^2-0.3402s+3.98)(s+40)} & 0 \\ 0 & \dfrac{10(s+2)^2}{s(s^2-0.3402s+3.98)} \end{bmatrix}$$

This means that we will achieve the co-sensitivity, $T(s)=(I+L(s))^{-1}L(s)$, listed below.

$$T(s)=\begin{bmatrix} \dfrac{300(s+2)^2}{(s+1.204)(s+16.14)(s^2+12.42s+61.77)} & 0 \\ 0 & \dfrac{10(s+2)^2}{(s+1.193)(s^2+8s+61.77)} \end{bmatrix}$$

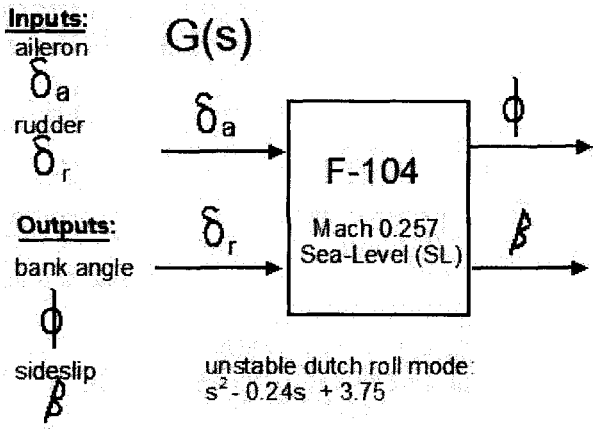


Fig. 6.9. Block diagram F-104. In this flight condition, the aircraft is open-loop unstable, but is also minimum phase (i.e. transmission zeros are stable).

A suitable left interactor, which is the same as in example 4.2.2, is listed below

$$\varepsilon_L(s)=\begin{bmatrix} s^2 & 0 \\ 0 & s \end{bmatrix}$$

As expected $\Lambda_G^{-1}(s)$ is stable, but has unstable transmission zeros corresponding to the unstable dutch roll mode. Also, note that the poles of $\Lambda_G^{-1}(s)$ correspond to the zeros of $G(s)$. The transfer function of $\Lambda_G^{-1}(s)$ is listed below.

$$\Lambda_G^{-1}(s)\approx\begin{bmatrix} \dfrac{0.21(s+27.92)(s+2.74)(s-0.0107)}{s^2(s+38.77)} & \dfrac{-35.46(s+0.972)(s-0.961)}{s(s+38.77)} \\ \dfrac{1.763(s-2.58)(s+0.173)}{s^2(s+38.77)} & \dfrac{31.56(s^2+0.4s+3.89)}{s(s+38.77)} \end{bmatrix}$$

Thus, we can compute $Q(s)$ by eq. 6.41 and the result is listed below. We know that T has been chosen properly and the closed loop will be stable because $Q(s)$ is stable.

$$Q(s)\approx\begin{bmatrix} \dfrac{84(s+2.74)(s-.011)(s^2+4s+4)}{(s+38.77)(s+1.185)(s^2+10.65s+48.54)} & \dfrac{-354.55(s+0.97)(s-.096)(s+2)^2}{(s+38.77)(s+1.176)(s^2+8.5s+34)} \\ \dfrac{705.14(s+2)^2(s-2.58)(s+0.173)}{(s+38.77)(s+27.83)(s+1.185)(s^2+10.65s+48.54)} & \dfrac{315.46(s+2.74)(s+2)^2(s^2+0.4s+3.89)}{(s+38.77)(s+1.176)(s^2+8.5s+34)} \end{bmatrix}$$

Now, we still need a compensator, and one would use eq. 6.40 to get the compensator.

G(s), lateral F-104 at Mach 0.257, SL

```
      -0.1780          0   -1.0000    0.1122
A=   -20.9000    -1.3800    1.1600         0
       2.6800    -0.0993   -0.1570         0
          0       1.0000         0         0

          0       0.0317
B=     4.7600     5.3500
       0.2660    -0.9230
          0          0

C=       0       0       0       1
         1       0       0       0
```

State Space. Input vector is $u = [\delta_a \quad \delta_r]^T$ units are [rad. rad.]. The output vector is $y = [\phi \quad \beta]^T$, units are [rad. rad.] respectively.

Example 6.13 – MATLAB code for decoupling of unstable F-104

```
syslat=ss(A,B,C,zeros(2));
g=eig(A);
bad_poles=[g(1) g(2)];
good_zeros=[-2 -2];

s=tf('s');
left_int=[s^2 0;0 s];

L=[zpk([good_zeros],[0 bad_poles -40],400) 0;
    0 zpk([good_zeros],[0 bad_poles],10)];

T=feedback(L,eye(2));

gamma_g=minreal(left_int*syslat);
inv_gamma_g=inv(gamma_g);

Q=minreal(inv_gamma_g*left_int*T,1e-1);
F=inv(eye(2)-T);

Comp=minreal(Q*F,1e-1); %Clean Up
```

An unfortunate thing in this case is that $C(s)$ is unstable. When it is connected into the closed loop, it will stabilize the system and decouple it. But it might make some uneasy to actually implement a controller with unstable poles. In this case, it is not absolutely unnecessary since one can employ the two-step strategy of section 4.3.

Also, the author has found this compensator difficult to implement for numerical reasons. Model order reductions of the 6th order compensator have not been met with success for the author.

6.6 Diagonal Decoupling of Stable and Non-Minimum Phase Systems with $G^{reg}(s)$-$C^{reg}(s)$

For a stable system with unstable transmission zeros (i.e. non-minimum phase behavior), it can be shown [6.7,2.16] that if diagonal decoupling is pursued, then each of diagonal elements of the loopshape must have all of the transmission zeros in each diagonal element.

$$L(s)=\begin{bmatrix} \dfrac{(s-s_{o1}^{RHP})(s-s_{o2}^{RHP})\ldots(s-s_{ow}^{RHP})}{d_{11}(s)} & 0\ldots & 0 \\ 0\ldots & \ddots & 0 \\ 0\ldots & 0 & \dfrac{(s-s_{o1}^{RHP})(s-s_{o2}^{RHP})\ldots(s-s_{ow}^{RHP})}{d_{mm}(s)} \end{bmatrix}$$

$$L_{pr}(s)=0 \quad r\neq p \tag{6.42}$$

$$L_{pp}(s)=\frac{(s-s_{o1}^{RHP})(s-s_{o2}^{RHP})\ldots(s-s_{ow}^{RHP})}{d_{pp}(s)} \quad \boldsymbol{Re}[s_{ok}^{RHP}]>0 \quad \forall k=1,2\ldots w$$

With this in mind, we will now do a longitudinal design example with the DC-8 aircraft in a landing configuration.

Example 6.14

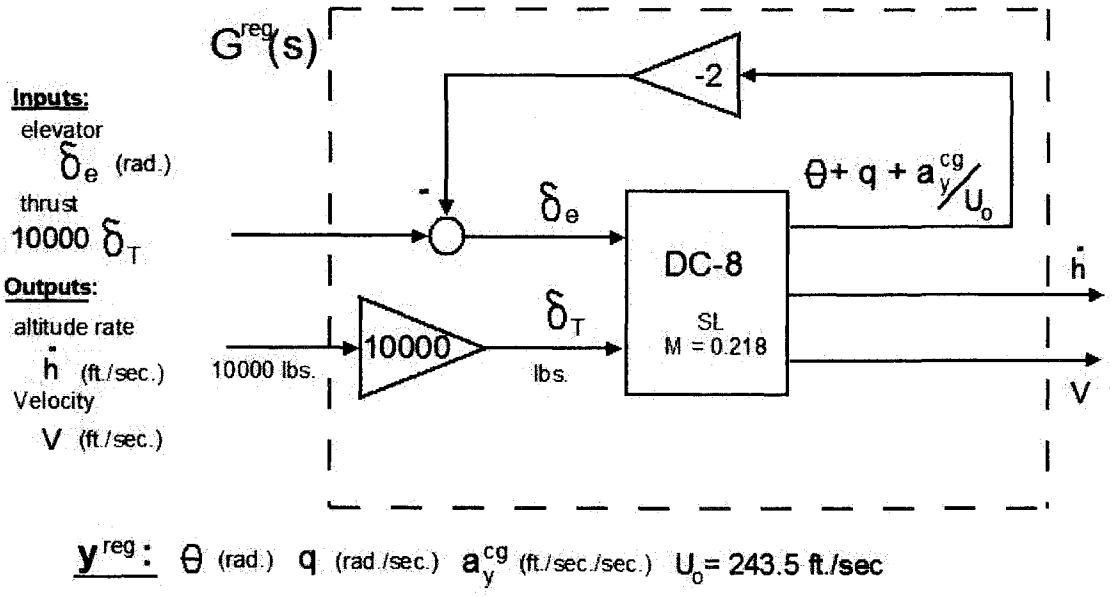


Fig. 6.10 For this design example, we will be designing a decoupling automatic landing system using the $G^{reg}(s)-C^{reg}(s)$ procedure.

G(s), longitudinal DC-8 at Mach 0.218, SL

$$A = \begin{bmatrix} -0.0291 & 0.0629 & 0 & -32.2000 \\ -0.2506 & -0.6277 & 243.5000 & 0 \\ 0.0002 & -0.0080 & -1.0525 & 0 \\ 0 & 0 & 1.0000 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0.1695e\text{-}3 \\ -10.1900 & 0 \\ -1.3391 & 0 \\ 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & -1.0000 & 0 & 243.5000 \\ 1.0000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.0000 \\ 0 & 0 & 1.0000 & 0 \\ -0.2506 & -0.6277 & 0 & 0 \end{bmatrix} \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -10.1900 & 0 \end{bmatrix}$$

State Space. Input vector is $u = \begin{bmatrix} \delta_e & \delta_T \end{bmatrix}^T$ units are [rad. lbs.]. The output vector is $y = \begin{bmatrix} \dot{h} & V & \theta & q & a_z^{cg} \end{bmatrix}^T$, units are [ft./sec. ft./sec. rad. rad./sec. ft./sec./sec.] respectively. $y^{reg} = \begin{bmatrix} \theta & q & a_z^{cg} \end{bmatrix}^T$, units are [rad. rad./sec. ft./sec./sec.] respectively.

Greg(s), longitudinal DC-8 at Mach 0.218, SL

$$A = \begin{bmatrix} -0.0291 & 0.0629 & 0 & -32.2000 \\ -0.2312 & -0.5792 & 224.6940 & -18.8060 \\ 0.0027 & -0.0017 & -3.5238 & -2.4714 \\ 0 & 0 & 1.0000 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 1.6947 \\ -9.4030 & 0 \\ -1.2357 & 0 \\ 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & -1.0000 & 0 & 243.5000 \\ 1.0000 & 0 & 0 & 0 \end{bmatrix}$$

State Space. Input vector is $u = \begin{bmatrix} \delta_e & 10000\delta_T \end{bmatrix}^T$ units are [rad. 10000lbs.]. The output vector is $y = \begin{bmatrix} \dot{h} & \Delta u \end{bmatrix}^T$, units are [ft./sec. ft./sec. rad. rad./sec. ft./sec./sec.] respectively.

As before, we will use the two step design with $G^{reg}(s) - C^{reg}(s)$, as seen in previous sections 6.4 and 6.3. We have used a regulator design to place the phugoid and short period modes on the real axis. The state space for $G(s)$ and $G^{reg}(s)$ is listed above.

There is an unstable zero at $s_{01}^{RHP} = 3.764$, which means that this unstable zero must appear in the diagonal entries of the loopshape as shown below.

$$L(s)=\begin{bmatrix} \dfrac{-0.5(s-3.764)}{(s+3.764)} & 0 \\ 0 & \dfrac{-0.5(s-3.764)}{(s+3.764)} \end{bmatrix}$$

We have also chosen a bandwidth of 0.5 rad./sec. which is approximately 7 times slower than the unstable transmission zero at $s_{01}^{RHP}=3.746$.

We find that an acceptable interactor is simply s, and is listed below.

$$\varepsilon_L(s)=\begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix}$$

We now use the interactor to generate inverse dynamics which are listed below.

$$\Lambda_{Greg}(s)=\varepsilon_L(s)G^{reg}(s)$$

$$\Lambda_{Greg}^{-1}(s)\approx\begin{bmatrix} \dfrac{0.10635(s+2.069)(s+1.615)(s+0.419)}{s(s+4.817)(s-3.764)} & \dfrac{-0.02459(s^2+3.746s+4.765)}{s(s+4.817)(s-3.764)} \\ \dfrac{0.03712(s-35.3)(s+1.08)}{s(s+4.817)(s-3.764)} & \dfrac{0.5901(s-0.001748)(s-3.747)(s+4.83)}{s(s+4.817)(s-3.764)} \end{bmatrix}$$

Example 6.14 – MATLAB code for decoupling of DC-8

```
s=tf('s');

sysg=ss(A,B,C,D);
H=[0 0 -2 -2 -2/Uo;0 0 0 0 0];
scale_inp=[1 0;0 10000];
sysg=feedback(sysg*scale_inp,H);

Greg=sysg(1:2,1:2);

evil_zero=max(zero(Greg));
left_int=[s 0;0 s];
L=[zpk([evil_zero],[-evil_zero 0],[-0.5]) 0;
    0 zpk([evil_zero],[-evil_zero 0],[-0.5])];

left_intL=minreal(left_int*L);
inv_Gamma_greg=minreal(inv(left_int*Greg));
Creg=minreal(inv_Gamma_greg*left_intL,1e-5);
```

The final controller synthesis step involves making unstable pole-zero cancellations between $\Lambda_{Greg}^{-1}(s)$ and $\varepsilon_L(s)L(s)$. If there is an error in this step, the controller, $C^{reg}(s)$ will have unstable poles corresponding to the unstable transmission zeros of $G^{reg}(s)$.

$$C^{reg}(s) = \Lambda_{Greg}^{-1}(s)\varepsilon_L(s)L(s)$$

$$C^{reg}(s) \approx \begin{bmatrix} \dfrac{-0.053174(s+2.069)(s+1.615)(s+0.419)}{s(s+4.817)(s+3.764)} & \dfrac{0.012296(s^2+3.746s+4.765)}{s(s+4.817)(s+3.764)} \\ \dfrac{-0.018558(s-35.3)(s+1.08)}{s(s+4.817)(s-3.764)} & \dfrac{-0.29504(s-0.001748)(s-3.747)(s+4.83)}{s(s+4.817)(s+3.764)} \end{bmatrix}$$

6.7 Robustness Issues with Diagonal Decoupling

It should be fairly intuitive that inverting the following plant in eq. 6.43 for controller synthesis could create problems.

$$G(s) = \begin{bmatrix} 1 & 1 \\ 1 & 1.0001 \end{bmatrix} \frac{1}{(s+1)(s+2)} \tag{6.43}$$

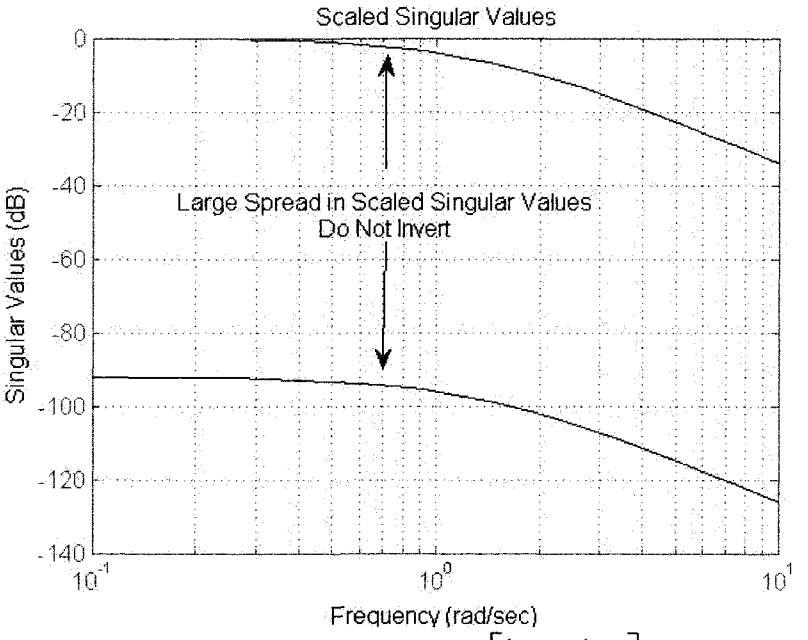


Fig. 6.11. Open Loop singular values of $G(s) = \begin{bmatrix} 1 & 1 \\ 1 & 1.0001 \end{bmatrix} \frac{1}{(s+1)(s+2)}$. We consider these inputs and outputs to be well-scaled.

Note that this plant has stable transmission zeros and is also open loop stable. However, this plant has a high condition number and very high relative gain array entries across all frequencies. Additionally, it is very close to being functionally uncontrollable because $\boldsymbol{det}(G(s)) = \dfrac{0.0001}{(s+1)(s+2)} \approx 0 \quad \forall s$.

When one encounters open loop dynamics like those shown in Fig. 6.11, there are fundamental problems with the control scheme. The plant may be better treated as a SISO control problem.

Restructuring or reconsidering inputs and outputs is most likely a good idea because a plant with a very high condition number is fundamentally difficult to control well by any method.

One thing that we must note here is that condition number alone is not enough to indicate this poor condition of near functional uncontrollability. Consider the following plant in eq. 6.44.

$$G(s)=\begin{bmatrix}1 & 0\\ 0 & 9001\end{bmatrix}\frac{1}{(s+1)(s+2)} \tag{6.44}$$

This plant should not exhibit any problems with dynamic inversion, yet, it too has a very high condition number. Thus, we introduce a diagonal scaling matrix so that these diagonal plants do not get mistaken for those which are almost functionally uncontrollable. It has been suggested that one could use a scaling scheme in order to gauge the robustness of inversion and point out ill conditioned control problems [2.10].

One might use a diagonal scaling at output of the plant with D_{sclo} as a diagonal scaling matrix at the output of the plant.

$$\begin{aligned} &consider: \quad D_{sclo}G(s)\\ &plot \quad \sigma(D_{sclo}G(j\omega)) \quad \omega<(5 \quad to \quad 10)\omega_{BW}\end{aligned}$$

If the spread in singular values is very large and there have been several attempts to scale properly, then it is very likely that the plant is poorly conditioned. Inverse based control is not a good idea in this case.

One might also use a diagonal scaling at the input of the plant, with D_{sclI} as a diagonal scaling matrix at the input of the plant.

$$\begin{aligned} &consider: \quad G(s)D_{sclI}\\ &plot \quad \sigma(G(j\omega)D_{sclI}) \quad \omega<(5 \quad to \quad 10)\omega_{BW}\end{aligned}$$

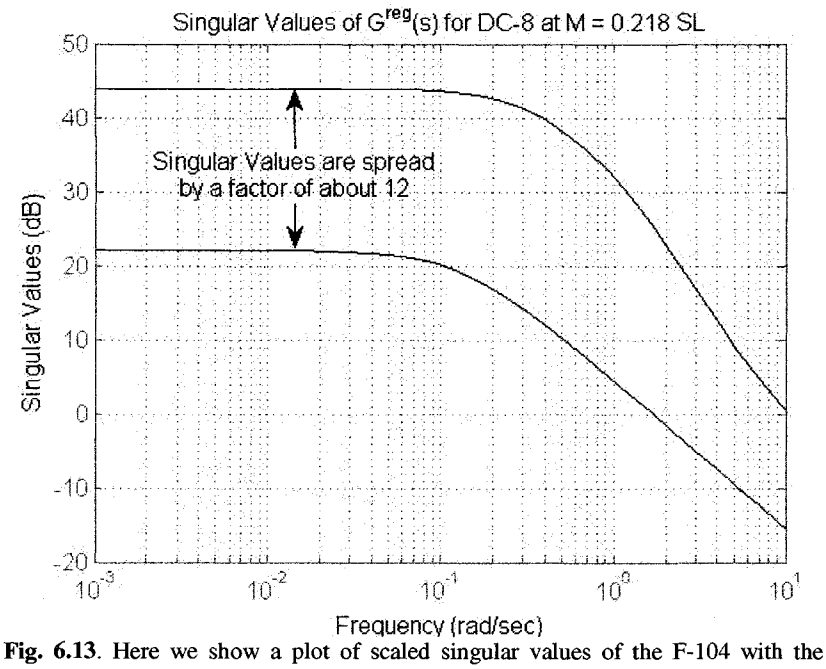


Fig. 6.13. Here we show a plot of scaled singular values of the F-104 with the regulator from section 6.4. Inputs of $G^{reg}(s)$ are δ_e and $10000\delta_T$ in units of [rad. 10000lbs.] respectively.

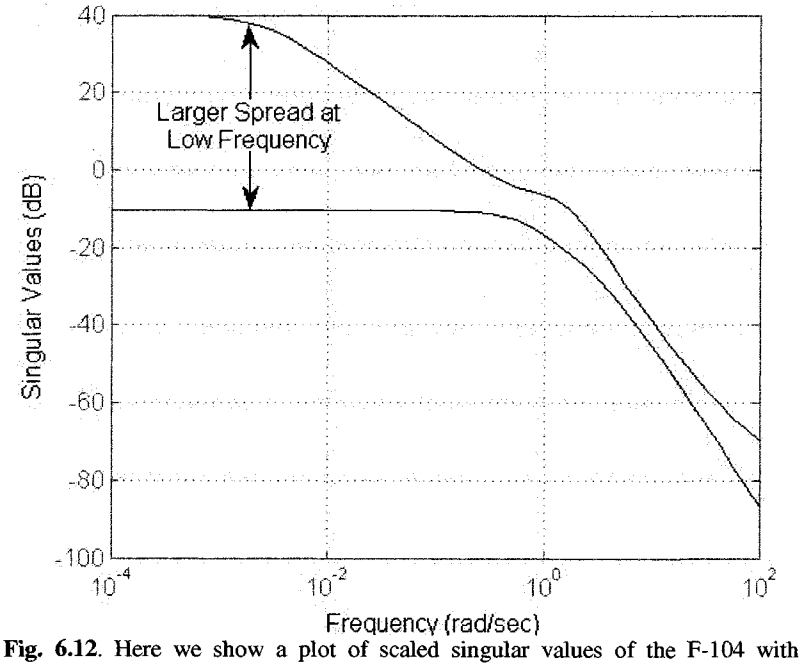


Fig. 6.12. Here we show a plot of scaled singular values of the F-104 with regulator from section 5.51. The scaling matrix is $D_{sclo} = \begin{bmatrix} 0.1 & 0 \\ 0 & 1 \end{bmatrix}$. Outputs of $G^{reg}(s)$ are ϕ and β respectively. Even with the scaling in place, the singular values are spread by a factor of nearly 100. This would indicate poor performance at low frequency.

Once again, if the spread in singular values is very large and there have been several attempts to scale properly, then it is very likely that the plant is poorly conditioned. Inverse based control is not a good idea in this case.

Figure 6.12 shows scaled singular values for the F-104 from section 6.4, where we used the $G^{reg}(s) - C^{reg}(s)$ two-step design procedure to achieve stability and diagonal decoupling. From Fig. 6.12 we can see that the scaled singular values shows a large spread at low frequency. Therefore, we have concerns that decoupling will perform poorly at low frequency.

In Fig. 6.13 we can see that the singular values

are spread over a factor of about 12. This manifests itself across a broad frequency range. There is little information available on what kind of spread is reasonable, but a spread of 10 is probably acceptable, whereas 100 may cause some concern.

An important result relates the relative gain array (RGA) to off nominal in the presence of multiplicative input uncertainty [6.7, 2.10].

We first relate the complimentary sensitivity at output, $T(s)$, the complimentary sensitivity at input, $T_I(s)$, and the inverse dynamics, $G^{-1}(s)$, with the following expression.

$$\begin{aligned} G(s)^{-1}T(s) &= C(s)(I+G(s)C(s))^{-1} = \left(C(s)^{-1}+G(s)\right)^{-1} \\ &= (I+C(s)G(s))^{-1}C(s) = T_I(s)G(s)^{-1} \end{aligned} \tag{6.45}$$

Next, we discuss the perturbed sensitivity, $S^p(s)$, in the presence of multiplicative input uncertainty $w_I\Delta_I(s)$. We also assume that $C(s)=G^{-1}(s)L(s)$.

$$\begin{aligned} S^p(s) &= \left(I+G^p(s)C(s)\right)^{-1} = \left(I+G(s)(I+w_I\Delta_I(s))G^{-1}(s)L(s)\right)^{-1} \\ &= \left(I+L(s)+G(s)w_I\Delta_I(s)G^{-1}(s)L(s)\right)^{-1} \\ &= \left(S(s)^{-1}+G(s)w_I\Delta_I(s)G^{-1}(s)L(s)\right)^{-1} \\ &= S(s)\left(I+G(s)w_I\Delta_I(s)G^{-1}(s)L(s)S(s)\right)^{-1} \\ &= S(s)\left(I+G(s)w_I\Delta_I(s)G^{-1}(s)T(s)\right)^{-1} \end{aligned} \tag{6.46}$$

We now insert the expression of eq. 6.45 into eq. 6.46 and arrive at the following simplified expression below in eq. 6.47.

$$\begin{aligned} S^p(s) &= S(s)\left(I+G(s)w_I\Delta_I(s)G(s)^{-1}T(s)\right)^{-1} \\ &= S(s)\left(I+G(s)w_I\Delta_I(s)T_I(s)G(s)^{-1}\right)^{-1} \\ &= S(s)\left(G(s)(I+w_I\Delta_I(s)T_I(s))G(s)^{-1}\right)^{-1} \\ &= S(s)G(s)(I+w_I\Delta_I(s)T_I(s))^{-1}G(s)^{-1} \end{aligned} \tag{6.47}$$

Now, we assume that diagonal decoupling was achieved and the loop shape was uniform across the diagonal. Note that if $T(s)$ is diagonal then $T_I(s)$ is not guaranteed to be diagonal. In general, $T_I(s)\neq T(s)$ for MIMO systems. The assumption of uniform diagonal decoupling is expressed below.

$$\begin{aligned}
&\text{if} \quad L(s)=(L(s))_{11}I, \quad (L(s))_{11} \quad \text{is} \quad \text{SISO} \\
&\Rightarrow \quad L_I(s)=G^{-1}(s)L(s)G(s)=(L(s))_{11}G^{-1}(s)G(s) \quad \text{and} \quad T_I(s) \quad \text{are} \quad \text{diaganol} \\
&\Rightarrow \quad T_I(s)=(T_I(s))_{11}I, \quad (T_I(s))_{11} \quad \text{is} \quad \text{SISO} \\
&\Rightarrow \quad S(s)=(S(s))_{11}I, \quad (S(s))_{11} \quad \text{is} \quad \text{SISO}
\end{aligned} \tag{6.48}$$

With this rather strict assumption, we can assume that the matrix, $(I+w_I\Delta_I(s)T_I(s))^{-1}$, is diagonal.

$$(I+w_I\Delta_I(s)T_I(s))^{-1}=\begin{bmatrix} 1-\dfrac{(w_I)_{11}(\Delta_I(s))_{11}(T_I(s))_{11}}{1+(w_I)_{11}(\Delta_I(s))_{11}(T_I(s))_{11}} & 0\ldots & 0 \\ 0\ldots & \ddots & 0 \\ 0\ldots & 0 & 1-\dfrac{(w_I)_{mm}(\Delta_I(s))_{mm}(T_I(s))_{11}}{1+(w_I)_{mm}(\Delta_I(s))_{mm}(T_I(s))_{11}} \end{bmatrix} \tag{6.49}$$

We can express the diagonal entries, $(S^p(s))_{kk}$, in terms of the relative gain array, $RGA(G(s))$ below.

$$\begin{aligned}
&S^p(s)=S(s)G(s)(I+w_I\Delta_I(s)T_I(s))^{-1}G^{-1}(s) \\
&(S^p(s))_{kk}=(S(s))_{11}\sum_{v=1}^{m}[RGA(G(s))]_{kv}r(s)_v \\
&r(s)_v=1-\frac{(w_I)_{vv}(\Delta_I(s))_{vv}(T_I(s))_{11}}{1+(w_I)_{vv}(\Delta_I(s))_{vv}(T_I(s))_{11}}
\end{aligned} \tag{6.50}$$

Purely for the sake of reinforcing the result, we imagine the case where there is no input uncertainty. We notice that everything behaves nominally due the special property of the relative gain array (RGA).

$$\begin{aligned}
&\sum_{v=1}^{m}[RGA(G(s))]_{kv}=1 \quad \text{when} \quad \text{rank[G(s)]}=m \\
&\text{Suppose} \quad (w_I)_{vv}=0, r(s)_v=1 \quad \forall v=1\ldots m \\
&(S^p(s))_{kk}=(S(s))_{11}\sum_{v=1}^{m}[RGA(G(s))]_{kv}=(S(s))_{11}
\end{aligned} \tag{6.51}$$

Now we re-introduce uncertainty at input and we also imagine a worst case scenario where $[RGA(G(j\omega))]_{kv}(w_I)_{vv}(\Delta_I(j\omega))_{vv}(T_I(j\omega))_{11}$ are out phase for each v such that large amplifications occur.

$$\text{Suppose} \quad (w_I)_{vv} \neq 0 \quad \forall v = 1 \ldots m$$

$$(S^p(s))_{kk} = (S(s))_{11} \sum_{v=1}^{m} [RGA(G(s))]_{kv} r(s)_v \qquad \sum_{v=1}^{m} [RGA(G(s))]_{kv} = 1$$

$$(S^p(s))_{kk} = (S(s))_{11} \left[1 - \sum_{v=1}^{m} [RGA(G(s))]_{kv} \frac{(w_I)_{vv} (\Delta_I(s))_{vv} (T_I(s))_{11}}{1 + (w_I)_{vv} (\Delta_I(s))_{vv} (T_I(s))_{11}} \right] \tag{6.52}$$

$$\text{Worst Case:} \quad \begin{array}{l} [RGA(G(j\omega))]_{kv} (w_I)_{vv} (\Delta_I(j\omega))_{vv} (T_I(j\omega))_{11} > 0 \\ \qquad\qquad or \\ [RGA(G(j\omega))]_{kv} (w_I)_{vv} (\Delta_I(j\omega))_{vv} (T_I(j\omega))_{11} < 0 \end{array} \quad \begin{array}{l} \forall v = 1 \ldots m \\ \\ \forall k = 1 \ldots m \end{array}$$

To add tangibility to our discussion, we provide a brief numerical result below for a poorly conditioned 2 input, 2 output system. Notice that the perturbed sensitivity is amplified by a factor of 2000 with only 10% input uncertainty. There can be no doubt that this is worrisome.

$$\text{Quick Example} \quad m = 2$$

$$G(s) = \begin{bmatrix} 1 & 1 \\ 1 & 1.0001 \end{bmatrix} \frac{1}{(s+1)(s+2)} \qquad G^{-1}(s) = \begin{bmatrix} 10001 & -10000 \\ -10000 & 10000 \end{bmatrix} (s+1)(s+2)$$

$$RGA(G(j\omega)) = \begin{bmatrix} 10001 & -10000 \\ -10000 & 10001 \end{bmatrix}$$

$$(w_I)_{11} (\Delta_I(j\omega))_{11} (T_I(j\omega))_{11} = -0.1 \quad (w_I)_{22} (\Delta_I(j\omega))_{22} (T_I(j\omega))_{11} = 0.1 \tag{6.53}$$

$$\sum_{v=1}^{m} [RGA(G(s))]_{1v} \frac{(w_I)_{vv} (\Delta_I(s))_{vv} (T_I(s))_{11}}{1 + (w_I)_{vv} (\Delta_I(s))_{vv} (T_I(s))_{11}} = -2020.3$$

$$\sum_{v=1}^{m} [RGA(G(s))]_{2v} \frac{(w_I)_{vv} (\Delta_I(s))_{vv} (T_I(s))_{11}}{1 + (w_I)_{vv} (\Delta_I(s))_{vv} (T_I(s))_{11}} = 2020.3$$

$$(S^p(s))_{11} = 2021.3 (S(s))_{11} \quad (S^p(s))_{22} = -2019.3 (S(s))_{11}$$

We can also use the following singular value inequality to obtain a lower bound for the maximum singular value of the perturbed sensitivity below. $\|RGA(G(j\omega))\|_{k\infty}$ represents the maximum row sum of the relative gain array. In the previous numerical example, $\|RGA(G(j\omega))\|_{k\infty} = 20001$.

$$\bar{\sigma}(S^p(j\omega)) \geq \max_k \left| (S^p(j\omega))_{kk} \right|$$

$$\bar{\sigma}(S^p(j\omega)) \geq (S(j\omega))_{11} \left(1 + \frac{|w_I \Delta_I(j\omega) (T_I(j\omega))_{11}|}{|1 + w_I \Delta_I(j\omega) (T_I(j\omega))_{11}|} \|RGA(G(j\omega))\|_{k\infty} \right) \tag{6.54}$$

Table. 6.1. A list of plants to decouple and plants not to decouple

	Diagonal Decoupling with $G^{reg}(s)-C^{reg}(s)$	Feedforward Action in Decentralized Control
Lateral Aircraft System Inputs: aileron, rudder Outputs: roll angle, yaw rate	Very poor robustness at low frequency, very high relative gain array (RGA) at low frequency due to aircraft's tendency to make steady turn. Do not decouple.	Has the same potential for poor performance and potentially instability if used incorrectly. Being careful not to invert low frequency dynamics and instead inverting at higher frequency can be very powerful.
Lateral Aircraft System Inputs: aileron, rudder Outputs: roll angle, sideslip angle	Mildly poor robustness at low frequency due to RGA at low frequency being usually around 3 to 8. Though sometimes the RGA may be as low as 1. Choose a lower bandwidth control for sideslip angle. Expect sensor or actuator failure to cause low frequency instability. Consider triangular decoupling procedure in next section.	The ability to choose at which frequency to invert is very powerful, though one must be careful not to make unstable pole zero cancellations even with stable zero dynamics in this case. Depending upon design, sensor or actuator failure may or may not cause low frequency instability.
Longitudinal Aircraft System Inputs: elevator, thrust Outputs: altitude rate, forward velocity	Fairly uniform robustness across all frequencies. However this technique rotates open loop unstable zero directions, which may be a fairly strange thing to do. Otherwise, this control technique may have potential.	The unstable zero dynamics of this plant make precise decoupling with feedforward action in decentralized challenging.
Two-Tank Apparatus Inputs: flows into each tank Outputs: Volume of each tank	Decoupling this plant is fundamentally impractical. The only true way to decouple in this case is to remove the pipe that connects the two tanks.	In this case, it makes little difference which method is used.
Car with Front and Rear Wheel Steering Inputs: front and rear wheel steering angle Outputs: yaw rate and lateral velocity (sideslip)	Very poor robustness at low frequency, very high relative gain array (RGA) at low frequency because both the front and rear wheels will enter the vehicle into a steady turn.	Feedforward action in decentralized control suggested a very intuitive input distribution scheme. Very successful in this case.
Suspension Inputs: front and rear forces Outputs: pitch angle and vertical displacement	Exceptionally conditioned plant, ideal for diagonal decoupling	Exceptionally conditioned plant, no problems with using either method

One should note that the regulator design step in the $G^{reg}(s)-C^{reg}(s)$ design procedure will affect the RGA and one should not expect $RGA(G(j\omega)) \neq RGA\left((I+G(j\omega)H(j\omega))^{-1}G(j\omega)\right)$ in general. However, in the event that the plant is poorly conditioned and $RGA(G(j\omega)) >> 1000$, then one should expect that $RGA(G(j\omega)) \approx RGA\left((I+G(j\omega)H(j\omega))^{-1}G(j\omega)\right)$ because a nearly rank deficient matrix $G(j\omega)$ multiplied by any matrix will be nearly rank deficient.

6.8 Triangular Decoupling of Minimum Phase TITO Plants with $G^{reg}(s)$-$C^{reg}(s)$

We assume that $G^{reg}(s)=(I+G(s)H(s))^{-1}G(s)$ is already designed and a *diagonal* left interactor, $\varepsilon_L(s)$ already exists and is known, details on the $G^{reg}(s)-C^{reg}(s)$ design for diagonal decoupling can be found in sections 6.3 and 6.4. In this section we will simply make one minor change. The math remains the same.

Our desired Loop shape $L(s)$ is triangular and is listed below, where $L(s)_{11}, L(s)_{22}$ and $L_{12}(s)$ are SISO transfer functions.

$$L(s)=\begin{bmatrix} L_{11}(s) & L_{12}(s) \\ 0 & L_{22}(s) \end{bmatrix} \tag{6.55}$$

Our inverse dynamics are listed below with a *diagonal* left interactor.

$$\begin{aligned} &\varepsilon_L(s)G^{reg}(s)=\Lambda_{Greg}(s) \\ &\Lambda_{Greg}(s)^{-1}=\frac{1}{\Lambda_{Greg}(s)_{11}\Lambda_{Greg}(s)_{22}-\Lambda_{Greg}(s)_{21}\Lambda_{Greg}(s)_{12}}\begin{bmatrix} \Lambda_{Greg}(s)_{22} & -\Lambda_{Greg}(s)_{12} \\ -\Lambda_{Greg}(s)_{21} & \Lambda_{Greg}(s)_{11} \end{bmatrix} \end{aligned} \tag{6.56}$$

If one knows what $L_{12}(s)$ should be already, then one can simply use the math below to get a controller.

$$\begin{aligned} &\varepsilon_L(s)G^{reg}(s)C^{reg}(s)=\varepsilon_L(s)L(s), \quad \Lambda_{Greg}(s)^{-1}=\left(\varepsilon_L(s)G^{reg}(s)\right)^{-1} \\ &C^{reg}(s)=\Lambda_{Greg}(s)^{-1}\varepsilon_L(s)L(s) \end{aligned} \tag{6.57}$$

Now, suppose we do not have any idea what $L_{12}(s)$ is supposed to be, but we have an idea what the controller should be like. We write the following math out below.

$$\begin{bmatrix} C^{reg}(s)_{11} & 0 \\ C^{reg}(s)_{21} & C^{reg}(s)_{22} \end{bmatrix} = \Lambda_{Greg}(s)^{-1}\varepsilon_L(s)L(s) \tag{6.58}$$

We insert the following substitution of $\varepsilon_L(s)L(s) = \Lambda_L(s)$.

$$\begin{bmatrix} C^{reg}(s)_{11} & 0 \\ C^{reg}(s)_{21} & C^{reg}(s)_{21} \end{bmatrix} = \begin{bmatrix} \left(\Lambda_{Greg}(s)^{-1}\right)_{11} & \left(\Lambda_{Greg}(s)^{-1}\right)_{12} \\ \left(\Lambda_{Greg}(s)^{-1}\right)_{21} & \left(\Lambda_{Greg}(s)^{-1}\right)_{22} \end{bmatrix} \begin{bmatrix} \Lambda_L(s)_{11} & \Lambda_L(s)_{12} \\ 0 & \Lambda_L(s)_{22} \end{bmatrix} \tag{6.59}$$

By this constraint, we can see that the following must be true of $L_{12}(s)$ below.

$$\Lambda_L(s)_{12} = -\frac{\left(\Lambda_{Greg}(s)^{-1}\right)_{12}}{\left(\Lambda_{Greg}(s)^{-1}\right)_{11}}\Lambda_L(s)_{22} = \frac{\Lambda_{Greg}(s)_{12}}{\Lambda_{Greg}(s)_{22}}\Lambda_L(s)_{22} \tag{6.60}$$

Sometimes, $L_{12}(s)$ may be a bit big. Therefore, one may reduce the off-diagonal loop shape by a factor, *f*, which is less than 1.

$$\Lambda_L(s)_{12} = f\frac{\Lambda_{Greg}(s)_{12}}{\Lambda_{Greg}(s)_{22}}\Lambda_L(s)_{22} \quad 0 \le f \le 1 \tag{6.61}$$

If one were to multiply by f being not equal to 1, then $C^{reg}(s)_{12} \neq 0$.

With $L_{12}(s)$ now chosen, we can easily calculate $C^{reg}(s)$ below.

$$C^{reg}(s) = \Lambda_{Greg}(s)^{-1}\varepsilon_L(s)L(s) \tag{6.62}$$

Example 6.15

As before, we will use the magical $G^{reg}(s) - C^{reg}(s)$ design procedure. Only we will be doing triangular decoupling. The X-15 has a very slow and unstable spiral root. Some small pitch angle feedback was used to correct this. We have also added some feedback of $\dot{\beta} = a_{cg}^{y}/U_o - r$ into the rudder.

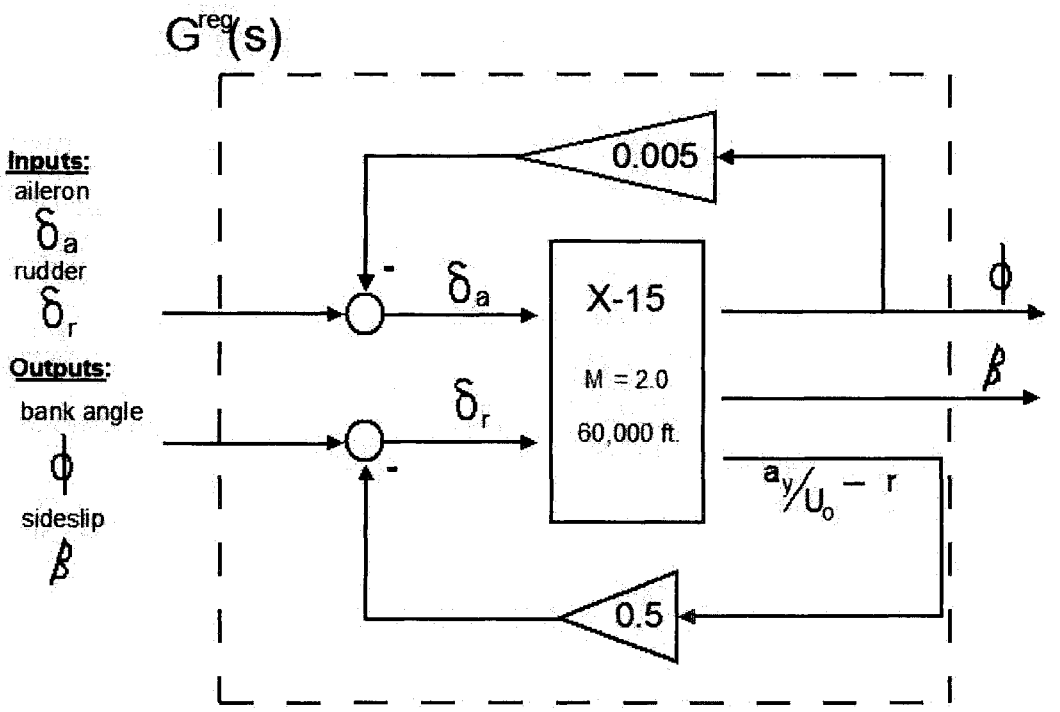


Fig. 6.14. For this design example, we will be designing a decoupling automatic landing system using the $G^{reg}(s)-C^{reg}(s)$ procedure.

We show a singular value plot of $G^{reg}(s)$ for the X-15 to select a good crossover frequency.

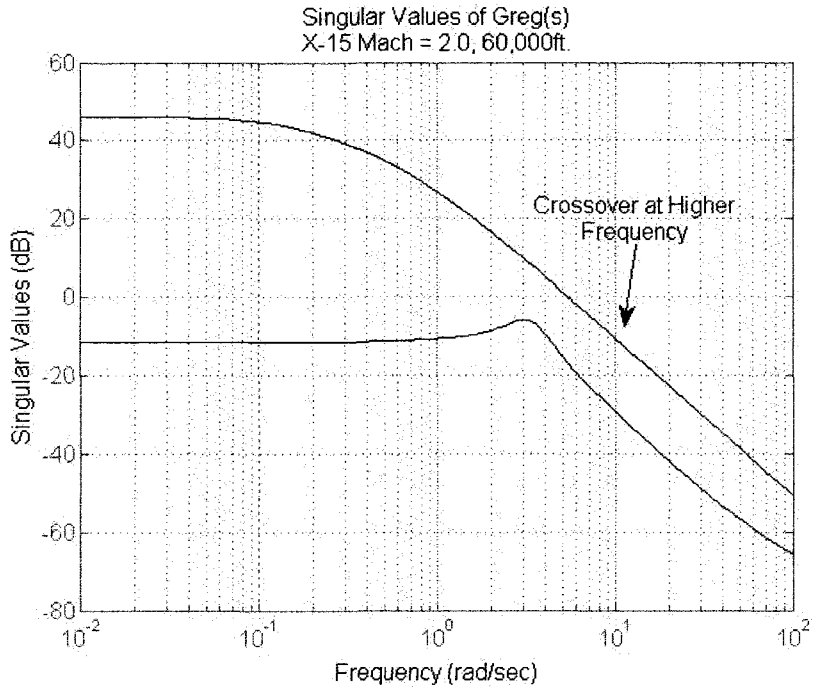


Fig. 6.15. From the open loop singular value plot, we can see that we would want to use a pretty high crossover frequency to control the X-15. We want to crossover where the bode plot has a nice $1/s$ -like behavior.

It may look like the X-15 is poorly conditioned at low frequency, but the RGA, covered in section 4.2, is actually quite reasonable. Inverse control should not be problematic at low frequency.

$$RGA\left(G^{reg}(0)\right) = G(0) \otimes G(0)^{-T} = \begin{bmatrix} 1.0274 & -0.0274 \\ -0.0274 & 1.0274 \end{bmatrix}$$

$\otimes$ is element by element multiplication

We decide on the following loop shape for the first output, $y_1 = \phi$, which has a very aggressive crossover frequency of 10 rad./sec.

$$L(s)_{11} = \frac{500}{s(s+50)}$$

The second output, $y_2 = \beta$, is less important, and we will be much more gentle to cut down on the magnitude of $L(s)_{12}$.

$$L(s)_{22} = \frac{0.25}{s}$$

Our interactor is simply given below.

$$\varepsilon_L(s)=\begin{bmatrix} s^2 & 0 \\ 0 & s \end{bmatrix}$$

Now we need to decide on our off-diagonal loop-shape $L_{12}(s)$.

$$\Lambda_{Greg}(s)_{12} \approx \frac{4.1222s^2(s^2+0.2287s+9.508)}{(s+0.09177)(s+0.9298)(s^2+1.729s+10.94)}$$

$$\Lambda_{Greg}(s)_{22} \approx \frac{0.041712s(s+67.77)(s+0.9578)(s+0.09767)}{(s+0.9298)(s+0.09177)(s^2+1.729s+10.94)}$$

$$\Lambda_L(s)_{12} \approx f\frac{98.8263s(s^2+0.2287s+9.508)}{(s+67.77)(s+0.9578)(s+0.09767)}0.25$$

We choose an f value of 0.1. Below is a bode plot of $L_{12}(s)$ and $L_{11}(s)$. Notice that we had to make $L_{22}(s)$ small in order to keep $L_{12}(s)$ small.

$L_{12}(s)$ will sometimes be very strong for lateral control with outputs of ϕ and β. Sometimes it is necessary to make performance sacrifices in β to cut down on this intense cross coupling.

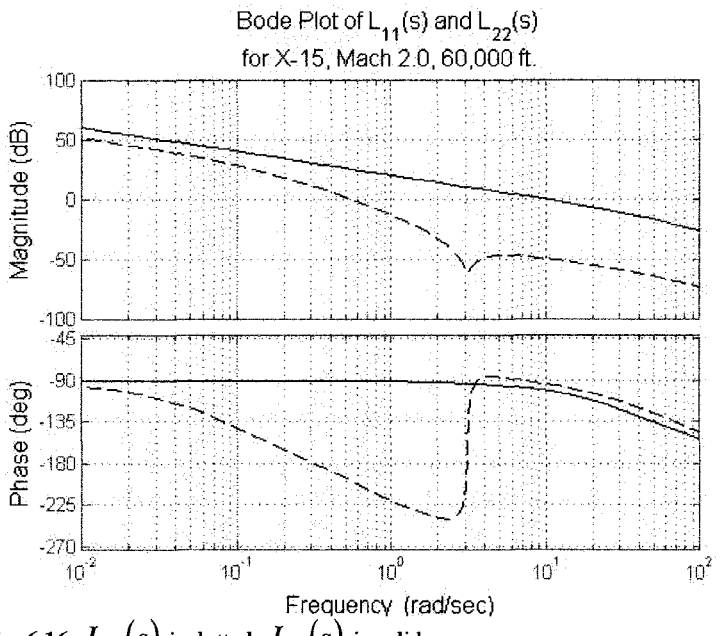


Fig. 6.16. $L_{12}(s)$ is dotted. $L_{11}(s)$ is solid.

It is not always true that $L_{12}(s)$ will be stable using this control method. In this case, one may just have to guess an acceptable value of $L_{12}(s)$.

One may want to just use a PID or PD wing leveler and $\dot{\beta}$ feedback to the rudder.

Example 6.15 - MATLAB code for Triangular Decoupling of X-15

```
clc
clear all

%X-15 lateral stability derivatives
%at Mach 2.0, 60k ft.

Uo=1936;
bank=0/180*pi;
cbank=cos(bank);
Yv=-.127;
Lbeta=-2.36;
Nbeta=11.1;
Lp=-1.02;
Np=-0.00735;
Lr=0.103;
Nr=-0.196;

Ysda=-0.00498;
Lda=28.7;
Nda=0.993;
Ysdr=.0426;
Ldr=4.21;
Ndr=-2.88;

Alat=[Yv 0 -1 32.2/Uo*cbank;Lbeta Lp Lr 0;
    Nbeta Np Nr 0;0 1 0 0];
Blat=[Ysda Ysdr;Lda Ldr;Nda Ndr;0 0];
Clat=[0 0 0 1;1 0 0 0;Yv 0 -1 32.2*cbank/Uo];
Dlat=[zeros(2,2);Ysda    Ysdr];

syslat=ss(Alat,Blat,Clat,Dlat);
Greg=feedback(syslat,[0.003 0 0;0 0 0.5]);

s=tf('s');
left_int=[s^2 0;0 s];
Gamma_Greg=minreal(left_int*Greg(1:2,1:2),1e-2);

L11=zpk([],[0 -50],500);
Gamma_L11=zpk([0],[-50],500);
L22=zpk([],[-1],1);
Gamma_L22=s*L22;

Gamma_L12=0.2*Gamma_L11;

Creg=inv(Gamma_Greg)*[Gamma_L11 Gamma_L12;...
    0 Gamma_L22];

Creg=minreal(Creg,1e-1);
```

6.9 Decoupling at Higher Frequency

We know that it is a bad idea to invert across all frequencies when the low frequency behavior is exceptionally singular as indicated by the RGA. Sometimes, one may want to take a look at the D_Λ matrix as a hint for a pre-compensator and then use decentralized control.

G(s), lateral T-38 at Mach 0.6, SL

```
        -0.3110         0    1.0000    0.0481          0    0.0675
A =     -29.690   -0.6277    0.7850         0    B =  19.270    8.334
         17.650    0.0965   -0.5970         0          0.870   -6.20
              0     1.000         0         0              0       0

C =       0         0         0    1.0000
          0         0    1.0000         0
```

State Space. Input vector is $u = [\delta_a \quad \delta_r]^T$ units are [rad. rad.]. The output vector is $y = [\phi \quad r]^T$, units are [rad./sec. rad./sec.].

For this example, we will use data from the T-38 at Mach = 0.6 at Sea Level (SL).

We see the singular value plot shown below.

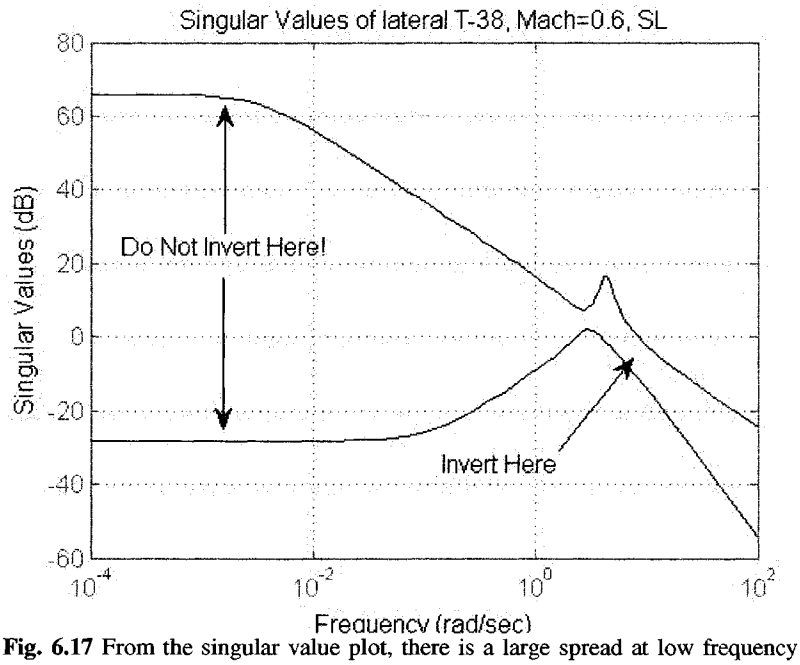


Fig. 6.17 From the singular value plot, there is a large spread at low frequency which means that there may potentially be the possibility of ill conditioned behavior at low frequency.

We have a suspicion that plant may be poorly conditioned from the singular value plot. Our suspicion is confirmed by checking the RGA at 0 frequency.

$$RGA(G(0)) = \begin{bmatrix} 240.2767 & -239.2767 \\ -239.2767 & 240.2767 \end{bmatrix}$$

Given that the relative gain array is so high at low frequency, we know that we should not invert there. Perhaps we should look at inverting at high frequency. We use a left interactor listed below.

$$\varepsilon_L(s) = \begin{bmatrix} s^2 & 0 \\ 0 & s \end{bmatrix}$$

Then we take the high frequency gain of $\Lambda_G(s) = \varepsilon_L(s)G(s)$.

$$\lim_{s\to\infty} \Lambda_G(s) = D_\Lambda = \begin{bmatrix} 19.27 & 8.334 \\ 0.876 & -6.200 \end{bmatrix}$$

We will then invert this matrix and use this as our pre-compensator, $P^{re} = D_\Lambda^{-1}T$, and transform it with a diagonal T matrix so that $P^{re} = \begin{bmatrix} 1 & RAI \\ ARI & 1 \end{bmatrix}$. These are the familiar Aileron-Rudder Interconnect (ARI) and Rudder to Aileron Interconnect (RAI) seen previously in Chapter 5.

From this data, we choose the following Pre-compensator.

$$T = \begin{bmatrix} 20.4475 & 0 \\ 0 & -6.5789 \end{bmatrix},\ P^{re} = \begin{bmatrix} 1 & RAI \\ ARI & 1 \end{bmatrix},\ ARI = 0.1413, RAI = -0.4325$$

With these gains designed, we use the following decentralized controller, with the pre-compensator. The same controller could have been designed using the techniques described in Chapter 5.

$$C(s) = \begin{bmatrix} 1 & -0.4325 \\ 0.1413 & 1 \end{bmatrix} \begin{bmatrix} \dfrac{4(s+3)(s+0.0035)}{s(s+20)} & 0 \\ 0 & \dfrac{-0.2s}{(s+0.25)} \end{bmatrix}$$

6.10 Limitations of Decoupling

The plant in section 2.30 with an unstable pole at s = 1 and an unstable transmission zero at s = 1 can be decoupled using the $G^{reg}(s) - C^{reg}(s)$ design procedure. One may find this perplexing.

Further details on which MIMO plants can be decoupled and which MIMO plants cannot be decoupled can be found in Ref. [6.8].

6.11 Compressing a $G^{reg}(s)$-$C^{reg}(s)$ Control Design into a Single Controller

The multiple loop closures of the $G^{reg}(s)-C^{reg}(s)$ design procedure is a reasonable control design procedure. However, one may wish to use a single controller instead of two. This section shows how one can achieve this compression *for the special case where* $y^{reg} = y$.

The control input can be expressed as follows in eq. 6.63.

$$u(s)=-H(s)y(s)+C^{reg}(s)e(s)=\left(H(s)+C^{reg}(s)\right)e(s)-Hr(s) \tag{6.63}$$

This would consist of a feed-forward element as well as a feedback element. The goal is to condense this into a single feedback controller. Therefore, the inverse of the sensitivity is used,

$$r(s)=\left[S^{reg}(s)\right]^{-1}=\left[I+(I+G(s)H(s))^{-1}G(s)C^{reg}(s)\right]e(s).$$

$$u(s)=\left[H(s)+C^{reg}(s)-H(s)-H(s)(I+G(s)H(s))^{-1}G(s)C^{reg}(s)\right]e(s) \tag{6.64}$$

With the matrix inversion lemma, $(I+HG)^{-1}=I-H(I+GH)^{-1}G$, eq. 6.64 can be simplified to give eq. 6.65.

$$u=\left[(I+H(s)G(s))^{-1}C^{reg}(s)\right]e \tag{6.65}$$

The total compensator, $C^{tot}(s)$, is listed in eq. 6.66.

$$C^{tot}(s)=\left[(I+H(s)G(s))^{-1}C^{reg}(s)\right] \tag{6.66}$$

Alternatively, one could use the definition of $C^{reg}(s)=G(s)^{-1}(I+G(s)H(s))L(s)$ and substitute this into 6.64 to arrive at eq. 6.67.

$$\begin{aligned} u&=\left[C^{reg}(s)-H(s)L(s)\right]e \\ C^{tot}(s)&=\left[C^{reg}-H(s)L(s)\right] \end{aligned} \tag{6.67}$$

6.12 Decoupling and Decentralized Outer Loop Control Study for Damaged P17

The effect of wing damage on inner loop control has been studied previously in sections 4.5 through 4.7 and section 5.7. This section will investigate the effect that damage has on outer-loop control structures which track outputs of heading angle, $\psi_v \approx \psi+\beta$, flight path angle, $\gamma \approx \theta-\alpha$, and

forward velcotiy, u. Inputs and outputs for outer loop control are shown below in Fig. 6.18. In outer loop control, inputs can be commands to inner loop control system. An example of which would be an altitude control system which would modify commands to pitch upwards to climb and pitch downwards to descend.

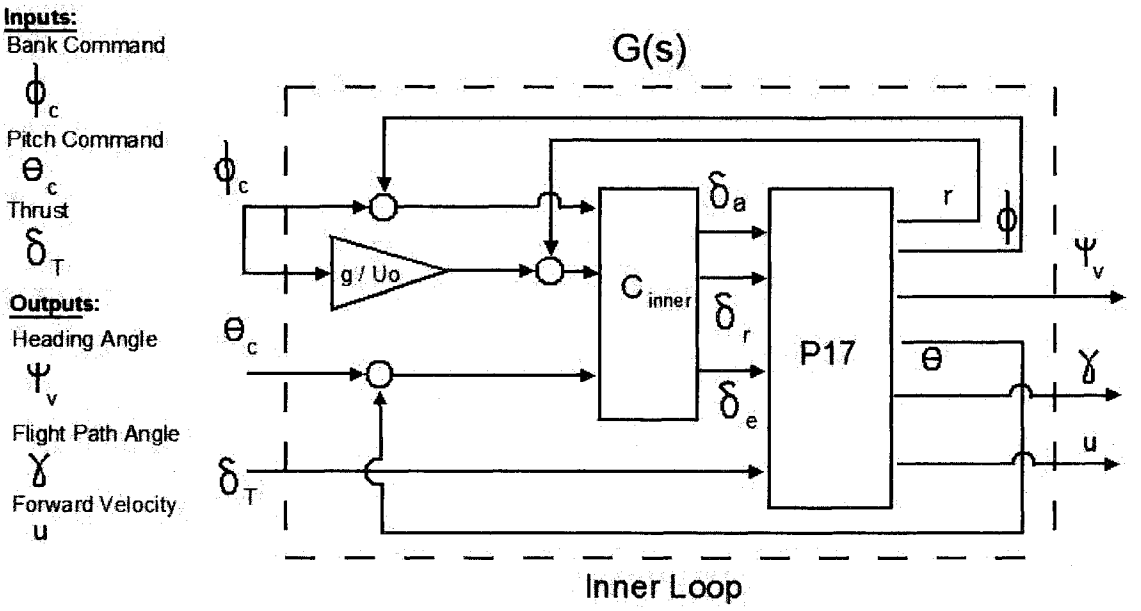


Fig. 6.18 Inputs and outputs for an outerloop control scheme where turn coordination is used in the inner loop.

The scheme shown in Fig. 6.18 is slightly different than the scheme used for the navion design used previously in example 4.7. There is a turn coordination system here to enforce a coordinated turn [6.9].

6.12.1 Review of Inner Loop Dynamics

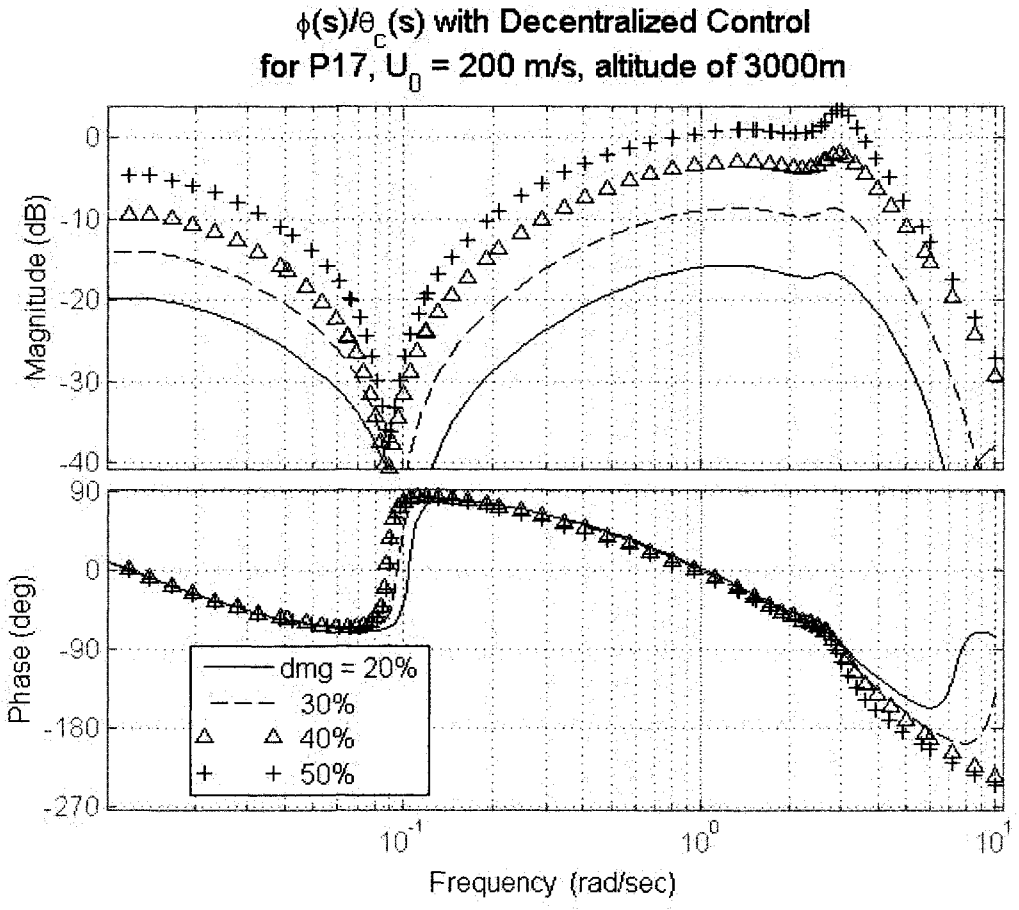


Fig. 6.19 Longitudinal to lateral cross coupling increases with increasing wing damage.

Of particular interest with damage is the tendency to bank when pitch up or pitch down. This behavior is shown in Fig. 6.19. One can see that the bank angle response to pitch command can become quite sizeable as damage increases. All of these responses are dynamically stable, however, it is reasonable to believe that a pilot or autopilot may become confused by this behavior when attempting to fly the aircraft. The response is barely noticeable at or below 20% damage. Damage above 30% begins to become worrisome.

It was shown in sections 4.5 through 4.7 that one can increase the roll gain to decrease the magnitude of $\frac{\phi}{\theta_c}(s)$. This can be achieved because the plant is triangular in nature. It was also shown in section 5.7 that one can further reduce the magnitude of $\frac{\phi}{\theta_c}(s)$ by adding cross-feed with an Elevator to Mega-Aileron Interconnect (EMAI). *However, one must take care not to add significant longitudinal-*

lateral cross-feed to the undamaged aircraft because this would unnecessarily endanger the undamaged aircraft.

The inner loop controller for this section will be assumed to be a decentralized controller taken partially from section 5.7. The bank angle controller was paired with the mega-aileron as in eq. 6.68.

$$C_{\delta ma\phi}(s)=\frac{-10(s+0.015)(s+3.75)}{s(s+15)} \tag{6.68}$$

The yaw rate controller was paired with the rudder as in eq. 6.69.

$$C_{\delta r}(s)=\frac{-0.574(s^2+0.966s+6.48)}{s(s+0.575)} \tag{6.69}$$

The integrator was removed from the pitch angle controller in section 5.7 and the controller in eq. 6.70 is used in this section instead.

$$C_{\delta r}(s)=\frac{-16(s+0.9)(s+0.1)}{(s+0.01)(s+20)} \tag{6.70}$$

6.12.2 Performance of Decentralized Outer Loop Autopilot

Relative Gain Array (RGA) versus frequency plots are shown in Fig. 6.20, 6.21 and 6.22 for the design of the outer loop autopilot with inputs and outputs defined by Fig. 6.18. The data used was the P17 at 200 m/s at an altitude of 3000m with 20% wing damage. The state space data for the P17 in this flight

condition can be found in section 5.7.

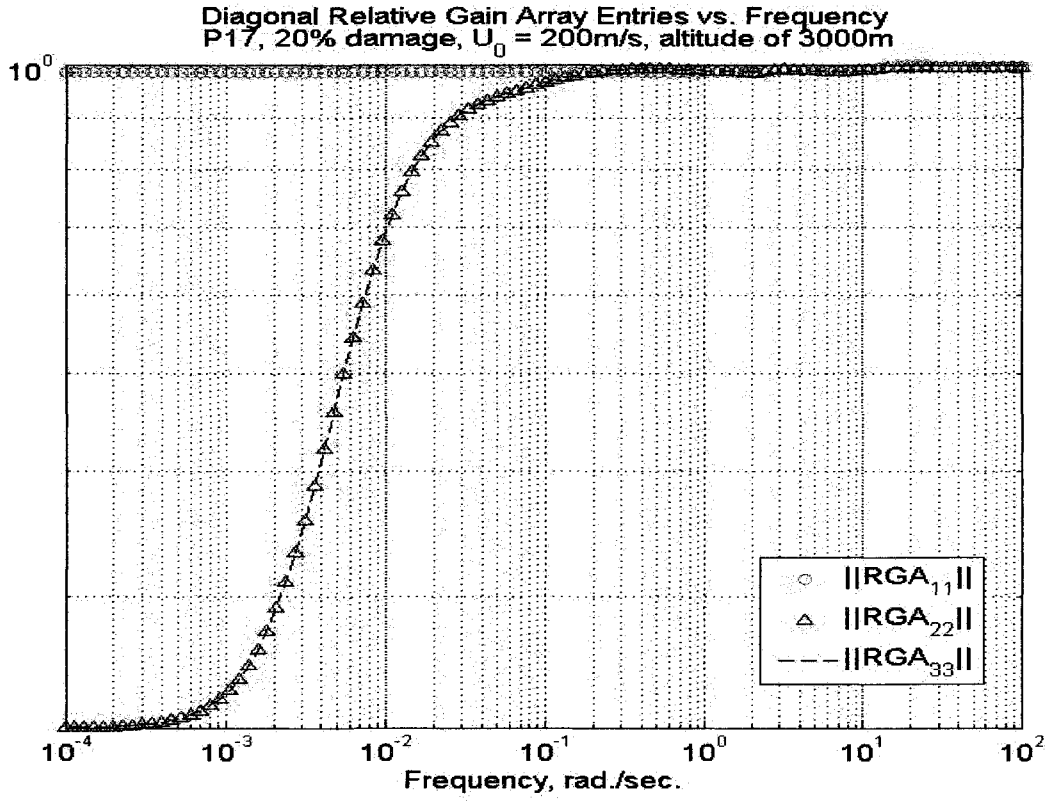


Fig. 6.20 Diagonal entries of the RGA for outer loop outputs

As can be seen from Fig. 6.20, the RGA becomes strongly diagonal in crossover region of 0.1 to 1 rad./sec., which indicates that a diagonal and decentralized controller will be a good choice of control for this frequency regime. Hence, it would be recommended to pair $\psi_v \rightarrow \left(\phi_c + {}^{g}/_{U_o} r_c\right)$, $\gamma \rightarrow \theta_c$

and $u \rightarrow \delta_T$ provided that the controller actually achieved bandwidths in the 0.1 to 1 rad./sec. regime.

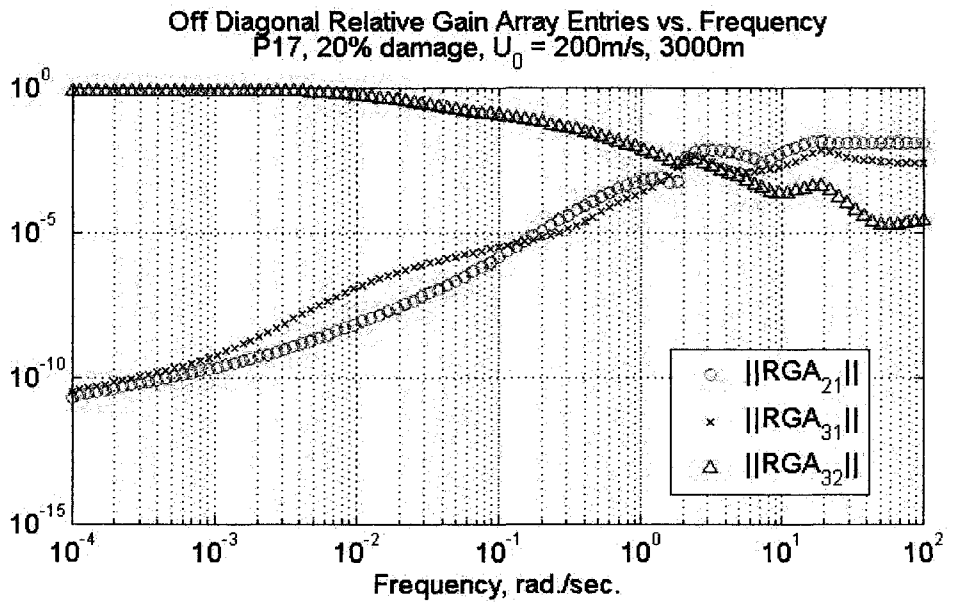


Fig. 6.21 Off diagonal RGA entries for outer loop outputs

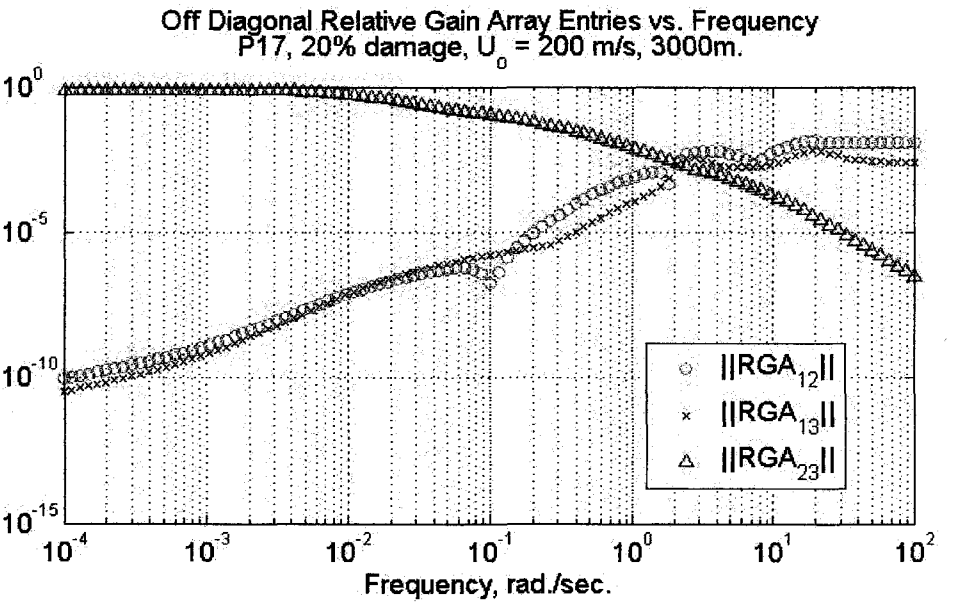


Fig. 6.22 More off diagonal RGA entries for outer loop outputs

Notice that at very low frequencies, the relative gain array suggests a strange pairing $\gamma \rightarrow \delta_T$ and $u \rightarrow \theta_c$. This is not unusual and the same tendency can be observed from example 4.7 which was the navion outer loop controller design. Recall also from example 4.7 that the coupling numerators, discussed in section 4.3, suggested that instability would be likely with a pairing of $\gamma \rightarrow \delta_T$ and $u \rightarrow \theta_c$. This is

because traditional open-loop linear aircraft models suggest that the d.c. gain of the $u/\delta_T(s)$ transfer function is 0. This is obviously inaccurate. Recall from section 3.2 that linear aircraft models are less coherent below 0.01 Hz.

There is no substantial difference between the RGA plots of Fig. 6.20, 6.21 and 6.22 and the RGA plots of the undamaged aircraft. Therefore, a traditional decentralized controller was designed to meet bandwidths of 0.3 rad./sec., 1.0 rad./sec. and 0.1 rad./sec. for control of heading, flight path angle and forward speed respectively. The heading angle controller is simply a gain of 6.

$$C_{\phi_c \psi_v}(s) = 6 \tag{6.71}$$

The flight path angle controller is simply a PI controller both with gains of 1.

$$C_{\theta_c \gamma}(s) = \frac{s+1}{s} \tag{6.72}$$

The airspeed speed controller is also a simple PI controller as shown in eq. 6.73. The units of thrust, δ_T, is in Newtons and the airspeed would be in units of meters per second. Even with this massive usage of thrust, a bandwidth of only 0.1 rad./sec. was achieved.

$$C_{\delta_T u}(s) = 50{,}000\left(\frac{s+0.0025}{s}\right) \tag{6.73}$$

A significant nuisance of longitudinal outer loop control design is caused by gravity. Gravity is responsible for tendencies to decrease airspeed during climbing and to increase airspeed during descent. Successfully counteracting this effect requires a great deal of thrust, regardless of whether one uses decentralized control or decoupling techniques found in section 6.6.

A linear simulation was run with 20% wing damage in Fig. 6.23. It is clear that the level of longitudinal to lateral cross coupling is quite small and not worth any design effort. At 20% wing

damage, the $\psi_v / \gamma_c (s)$ transfer function remains below -40dB.

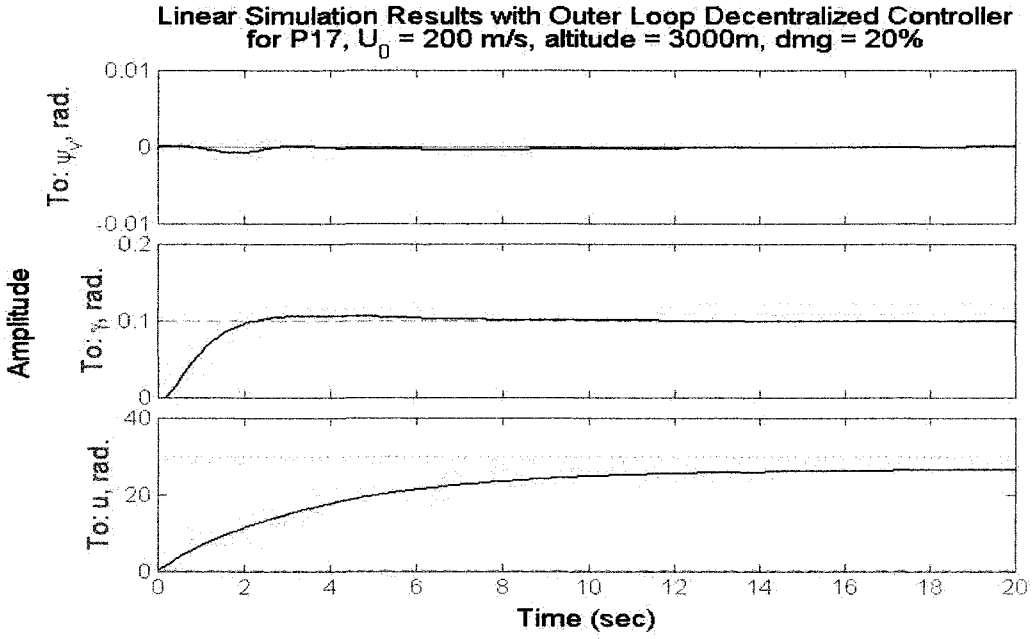


Fig. 6.23 Step response to increase speed and climb without changing heading with 20% wing damage

As can be seen in Fig. 6.24, the handling gets slightly worse with 40% wing damage. The heading angle correction is slowed, which is unfortunate. However, it is most likely at a tolerable level of

error for basic survival of the aircraft.

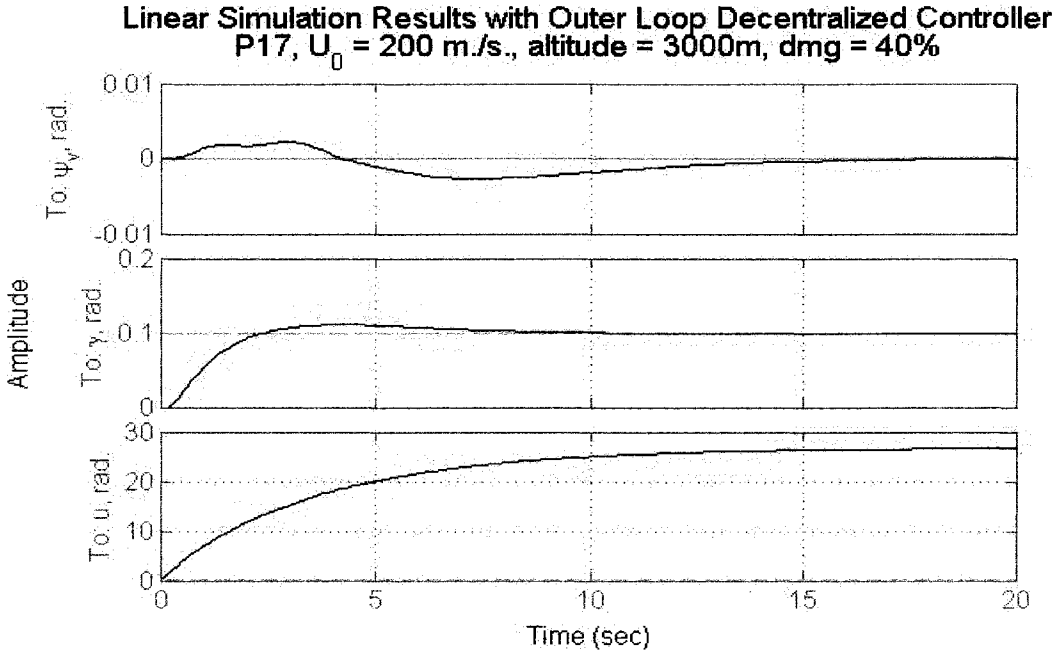


Fig. 6.24 Step response to increase speed and climb without changing heading with 40% damage

6.12.3 Performance of Decoupled Outer Loop Autopilot

Relative Gain Array (RGA) versus frequency plots shown in Fig. 6.20, 6.21 and 6.22 indicate that an inversion of the outer loop will not be ill-conditioned. As discussed in section 6.7, if the RGA entries became large, like above 10 or 100, inversion control would not be advisable. Also, outer loop poles are well damped from the inner loop controller. Inversion based outer loop control could work for a wing damaged aircraft.

Unstable transmission zeros typically accompany outer loop control and special accommodations would need to be made as in section 6.6. An exception would be an aircraft with a canard which would have stable transmission zeros for the outputs of γ and u. However, transmission zeros may be poorly damped as was the case with the X-29 [6.10]. Neither situation is ideal.

Unstable outer loop transmission zeros for the P17 with 20% damage are listed in table 6.2. Notice that there is pair of oscillatory unstable zeros which is somewhat unusual. However, they are at a high frequency for outer loop control and are less of a concern than an

Unstable Transmission zeros of G(s)
8.508, 4.423+18.409j, 4.423-18.409j

Table 6.2. MIMO Transmission zeros of G(s) which are unstable

engine time delay.

An acceptable diagonal loop-shape for outer loop control would therefore be listed in eq. 6.75.

$$L(s)=\begin{bmatrix} \frac{0.3}{s} & 0 & 0 \\ 0 & \frac{1}{s} & 0 \\ 0 & 0 & \frac{0.1}{s} \end{bmatrix} \frac{-(s-8.508)(s^2-8.845s+358.4)}{(s+8.508)(s+18)^2} \tag{6.75}$$

With a highly accurate model, an inverse controller would actually show less cross coupling over the decentralized controller. However, according to the modeled data, a decentralized controller is adequate even with damage. Also, the decoupling controller would add longitudinal lateral cross-feed. *If this cross-feed were misapplied to the undamaged aircraft, then would unnecessarily endanger the undamaged aircraft. Without good fault detection and isolation or system identification, it would be best not to pursue a decoupling controller.*

Chapter 7 –Dynamics and Control of a Maneuvering Flexible Vehicle

7.1 Introduction to Dynamics of a Maneuvering Flexible Vehicle

This section introduces flexible dynamics associated to a maneuvering vehicle using a supersonic beam, which is easier to analyze than a subsonic beam model.

Typically, aeroelastic effects are analyzed as though the aircraft is restrained. For example, a wing is assumed to be a cantilever attached to a motionless rigid base. However, when an elastic aircraft is actually flying, this rigid base to which the wing is attached is not motionless. Typical justification for such an assumption of this rigid base is that there is a large separation in frequency between rigid body motions and flexible dynamics.

Table. 7.1. Comparison of "unified theory" of aeroelastic vehicles and the traditional "mean axis"

	Unified Theory	Mean Axis
Aerodynamics	Same as Mean Axis	Same as Unified Theory
Flexible dynamics	Same as Mean Axis	Same as Unified Theory
Momentum and Mass Effects	Rigid body mode and flexible modes may influence one another	Linear and angular momentum of flexible modes are conserved, no interaction between flexible and rigid modes
Reference point	Arbitrary	Center of mass of undeflected structure
Strengths	Can be extended to any aircraft at any flight condition	Easier to use, very good for low dynamic pressures and an appropriately stiff aircraft
Weaknesses	Difficult to implement, many times will calculate insignificant terms	May ignore important effects for highly flexible aircraft at very high dynamic pressures

Models which consider an elastic vehicle which is also maneuvering (not attached to a motionless base) have been developed [3.25,7.1,7.2]. Frequently, a "mean axis" assumption is made [3.25]. In a perfect vacuum, using the "mean axis" assumption, rigid body dynamics would be completely decoupled from flexible dynamics [3.25]. Thus, according to the "mean axis" assumption, the only source of coupling between rigid body dynamics and flexible dynamics are aerodynamic forces [3.25]. For example, a wing which moves up and down will produce variations in lift which act on the rigid body motion. However, the momentum of the wings moving up and down will not cause any variation in rigid body motion under the "mean axis" assumption.

Recently, a "unified theory" for aeroelastic vehicles has been developed [7.1,7.2]. This "unified theory" is rather elaborate as it captures all momentum and mass related effects that elastic deformation have on rigid motions and all the effects that rigid motions have on elastic deformations. Table 7.1

displays the fundamental differences between this "unified theory" and the "mean axis" style. In this section, a numerical example will explore both the "unified theory" style and the "mean axis" style.

7.2 Longitudinal Equations of Motion for a Beam Model Using Unified Theory

A supersonic flexible beam is chosen as an example because of the simplicity associated with analysis. Only the longitudinal equations of motion will be studied and the lift distribution is assumed to be the same as a flat plate.

For supersonic flow over a flat plate and most thin airfoils, lift is uniformly distributed and the aerodynamic center is at the midpoint of the chord length (also the length of the beam). This is convenient because the c.g. of the beam is at the midpoint of the chord as well. The c.g. of the beam will be exactly at the neutral point. Therefore, this "vehicle" is perfectly balanced.

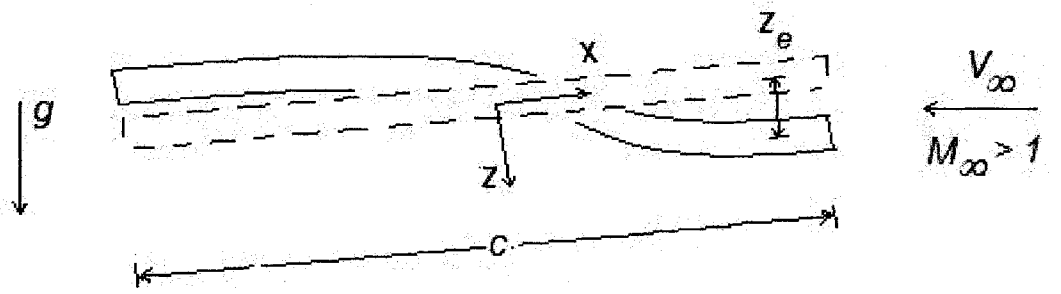


Fig. 7.1. Supersonic flexible beam has length c. The origin is located at the midpoint of the beam, where the c.g. of the rigid beam would be. The x and z axes are in a *body axis system, not a wind/stability axis system.* The elastic deformation only occurs in the z-axis and is labeled **Ze.** A supersonic flight condition is chosen because the aerodynamic center of a flat plate is at the midpoint, which is exactly the location of the c.g. of the rigid beam.

Fig. 7.1 shows the geometry of the supersonic beam from the side view. The supersonic beam is only allowed to deflect in the z-axis of the *body axis system.* This elastic deflection will be labeled as **Ze**.

Fig. 7.2 shows the cross section geometry of the supersonic beam.

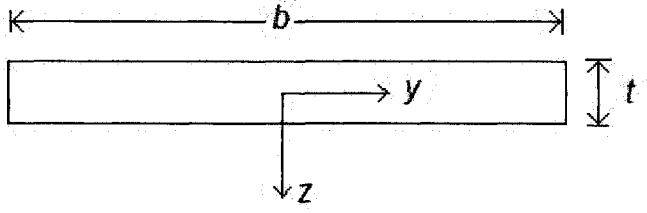


Fig. 7.2.Cross section geometry of supersonic beam.

A Lagrangian approach is used to derive equations of motion under the unified theory [7.1,7.2] and will be used again in this case.

To begin, one needs kinematic expressions for the velocity of an arbitrary point on the beam. Since only the longitudinal equations are of interest, lateral variables will be constrained to zero, $v(t) = r(t) = p(t) = 0$.

$$\begin{bmatrix} u \\ w \end{bmatrix} = \begin{bmatrix} u_r + (z+z_e)q_r \\ w_r - xq_r + \dot{z}_e \end{bmatrix} \tag{7.1}$$

Now, one considers the kinetic energy of the supersonic beam in eq. 7.2.

$$\begin{aligned} T = \tfrac{1}{2}\int \rho_b \begin{bmatrix} u \\ w \end{bmatrix}^T \begin{bmatrix} u \\ w \end{bmatrix} dV &= \tfrac{1}{2}\int \rho_b dV(u_r^2 + w_r^2) + \int \rho_b (z+z_e) dV(u_r q_r) \\ &+ \int \rho_b (-x) dV(w_r q_r) + \int \rho_b (\dot{z}_e) dV(w_r) + \int \rho_b (-x\dot{z}_e) dV(q_r) \\ &+ \tfrac{1}{2}\int \rho_b (x^2 + z^2 + 2z_e z + z_e^2) dV(q_r^2) + \tfrac{1}{2}\int \rho_b (\dot{z}_e^2) dV \end{aligned} \tag{7.2}$$

The elastic deflection, **Ze**, will be represented as a mode shape times a generalized coordinate via the separation of variables representation.

$$z_e(x,t) = \sum_{i=1}^{\infty} \Phi_i(x)\eta_i(t) \approx \sum_{i=1}^{N} \Phi_i(x)\eta_i(t) \tag{7.3}$$

Inserting the mode shape representation into the kinetic energy, one arrives at eq. 7.4.

$$\begin{aligned} T = \tfrac{1}{2}\int \rho_b dV(u_r^2 + w_r^2) &+ \int \rho_b (z) dV(u_r q_r) + \sum_{i=1}^{N} \int \rho_b \Phi_i(x) dV(\eta_i u_r q_r) \\ + \int \rho_b (-x) dV(w_r q_r) &+ \sum_{i=1}^{N} \int \rho_b \Phi_i(x) dV(\dot{\eta}_i w_r) + \sum_{i=1}^{N} \int \rho_b (-x\Phi_i(x)) dV(\dot{\eta}_i q_r) \\ &+ \tfrac{1}{2}\int \rho_b (x^2 + z^2) dV(q_r^2) + \sum_{i=1}^{N} \int \rho_b (z\Phi_i(x)) dV(\eta_i q_r^2) + \\ + \tfrac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \int \rho_b (\Phi_i(x)\Phi_j(x)) dV(q_r^2 \eta_i \eta_j) &+ \tfrac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \int \rho_b (\Phi_i(x)\Phi_j(x)) dV(\dot{\eta}_i \dot{\eta}_j) \end{aligned} \tag{7.4}$$

The origin is located at the un-deformed c.g., $\int \rho_b (x) dV = \int \rho_b (z) dV = 0$. There is symmetry in the x-z plane, $\int \rho_b z \Phi_i(x) dV = 0$. Also, the modeshapes are assumed to be orthogonal, which is true for modeshapes of free response *in a vacuum*, $\int \rho_b \Phi_i(x)\Phi_j(x) dV = 0, \quad i \neq j$. Including these relations gives an updated expression for the kinetic energy in eq. 7.5.

$$T = \tfrac{1}{2}\int \rho_b dV\left(u_r^2 + w_r^2\right) + \sum_{i=1}^{N}\int \rho_b \Phi_i(x) dV(\eta_i u_r q_r) + \sum_{i=1}^{N}\int \rho_b \Phi_i(x) dV(\dot{\eta}_i w_r)$$
$$+ \sum_{i=1}^{N}\int \rho_b(-x\Phi_i(x)) dV(\dot{\eta}_i q_r) + \tfrac{1}{2}\int \rho_b\left(x^2 + z^2\right) dV\left(q_r^2\right) \qquad (7.5)$$
$$+ \tfrac{1}{2}\sum_{i=1}^{N}\int \rho_b\left(\Phi_i^2(x)\right) dV\left(q_r^2\eta_i^2\right) + \tfrac{1}{2}\sum_{i=1}^{N}\int \rho_b\left(\Phi_i^2(x)\right) dV\left(\dot{\eta}_i^2\right)$$

The potential energy considered here is the strain energy and the gravitational potential energy, which is given in eq. 7.6.

$$V = \tfrac{1}{2}\sum_{i=1}^{N}\int EI\left(\Phi_i''(x)\right)^2 dV\left(\eta_i^2\right) + \int \rho_b g\left(h_r + x\sin(\theta_r) - \cos(\theta_r)\sum_{i=1}^{N}\Phi_i(x)\eta_i\right) dV \qquad (7.6)$$

Since the origin is also the c.g., $\int \rho_b(x\sin(\theta_r)) dV = 0$. The change in gravitational energy due to elastic deformation will be neglected in this case, $\cos(\theta_r)\int \rho_b g\sum_{i=1}^{N}\Phi_i(x)\eta_i \approx 0$ (although one could account for this). These updates to the potential energy are given in eq. 7.7.

$$V = \tfrac{1}{2}\sum_{i=1}^{N}\int EI\left(\Phi_i''(x)\right)^2 dV\left(\eta_i^2\right) + \int \rho_b g(h_r) dV \qquad (7.7)$$

Aerodynamic forces and moments will be considered at a later stage. For now, the lagrangian will be pursued further to get mass and stiffness terms.

$$\partial T/\partial u_r = mu_r + \sum_{i=1}^{N}\int \rho_b \Phi_i(x) dV \eta_i q_r$$
$$\partial T/\partial w_r = mw_r + \sum_{i=1}^{N}\int \rho_b \Phi_i(x) dV \dot{\eta}_i$$
$$\partial T/\partial q_r = I_{yy} q_r + \sum_{i=1}^{N}\int \rho_b \Phi_i^2(x) dV \eta_i^2 q_r + \sum_{i=1}^{N}\int \rho_b \Phi_i(x) dV \eta_i u_r \qquad (7.8)$$
$$- \sum_{i=1}^{N}\int \rho_b x\Phi_i(x) dV \dot{\eta}_i$$
$$\partial T/\partial \dot{\eta}_i = \int \rho_b \Phi_i(x) dV w_r - \int \rho_b x\Phi_i(x) dV q_r + \int \rho_b \Phi_i^2(x) dV \dot{\eta}_i$$
$$\partial T/\partial \eta_i = \int \rho_b \Phi_i(x) dV u_r q_r + \int \rho_b \Phi_i^2(x) dV \eta_i q_r^2$$

Now, to get equations of motion, without aerodynamics, one must use the following relations for a body-fixed coordinate system, shown in eq. 7.9. (note: ***L=T-V,*** $\tilde{\omega}r = \omega \times r$). In eq. 7.9,

$$C_r = \begin{bmatrix} \cos\theta_r & -\sin\theta_r \\ \sin\theta_r & \cos\theta_r \end{bmatrix},\ \partial L/\partial R_r = [0 \quad mg]^T, E_r = 1 \text{ and } \partial L/\partial \theta_r \approx 0 \text{ from eq. 7.7.}$$

$$\begin{aligned} \partial/\partial t\left(\partial L/\partial V_r\right) + \tilde{\omega}_r \partial L/\partial V_r - C_r \partial L/\partial R_r &= \sum F \\ \partial/\partial t\left(\partial L/\partial \omega_r\right) + \tilde{V}_r \partial L/\partial V_r + \tilde{\omega}_r \partial L/\partial \omega_r - \left(E_r^{-T}\right)\partial L/\partial \theta_r &= \sum M \\ \partial/\partial t\left(\partial L/\partial \dot{\eta}_i\right) - \partial L/\partial \eta_i &= \int \Phi_i \left(\sum f\right) dV \end{aligned} \tag{7.9}$$

Only the longitudinal equations of motion are of interest here, so $v(t) = r(t) = p(t) = 0$ and as a consequence, $V_r = [u_r \quad w_r]^T$, $\omega_r = q_r$. Performing the manipulations of eq. 7.9 yields eq. 7.10.

$$\begin{aligned} \sum F_x &= m\dot{u}_r + mw_r q_r + \sum_{i=1}^{N} \int \rho_b \Phi_i(x) dV \left(\eta_i \dot{q}_r + 2\dot{\eta}_i q_r\right) + mg \sin\theta_r \\ \sum F_z &= m\dot{w}_r - mu_r q + \sum_{i=1}^{N} \int \rho_b \Phi_i(x) dV \left(\ddot{\eta}_i - \eta_i q_r^2\right) - mg \cos\theta_r \\ \sum M_y &= I_{yy}\dot{q}_r + \sum_{i=1}^{N} \int \rho_b \Phi_i^2(x) dV \left(\eta_i^2 \dot{q}_r + 2\eta_i \dot{\eta}_i q_r\right) - \sum_{i=1}^{N} \int \rho_b x \Phi_i(x) dV \left(\ddot{\eta}_i\right) \\ &\quad + \sum_{i=1}^{N} \int \rho_b \Phi_i(x) dV \left(\eta_i \dot{u}_r + \eta_i w_r q_r\right) \\ \int \Phi_i(x)\left(\Sigma f_z(x)\right) dx &= \int \rho_b \Phi_i(x)\left[\dot{w}_r - u_r q_r - x\dot{q}_r + \Phi_i(x)\ddot{\eta}_i - \Phi_i(x) q_r^2 \eta_i\right] dV \\ &\quad + \int EI\left(\Phi_i''(x)\right)^2 dx \\ \Delta\dot{\theta}_r &= \Delta q_r \end{aligned} \tag{7.10}$$

The next thing to do is to focus on aerodynamics, which will be added in as external forces and moments. To start, only aerodynamics for a rigid supersonic beam will be considered. For a flat plate in a supersonic flight condition, lift is uniformly distributed over the surface of the plate. Also, if 3D aerodynamic effects are ignored, theoretical lift and drag coefficients are easily available. Fig. 7.3 shows the lift distribution for an idealized supersonic flat plate.

Notice from Fig. 7.3 that lift and drag act in a different axis system than the body fixed x-z axis. This makes a coordinate transformation necessary. Also, one should note that $\alpha = \tan^{-1}\left(w_r/u_r\right)$.

$$\begin{bmatrix} F_x \\ F_z \end{bmatrix} = \begin{bmatrix} -\cos\alpha & \sin\alpha \\ -\sin\alpha & -\cos\alpha \end{bmatrix} \begin{bmatrix} D_{rag} \\ L_{ift} \end{bmatrix} \tag{7.11}$$

For an infinitely thin flat plate neglecting 3D effects, aerodynamic theory predicts the following relationship between angle of attack (AOA, α) and lift.

$$L_{ift} = q_\infty S C_{L_{ift}} \quad D_{rag} = q_\infty S C_{D_{rag}} \quad C_{L_{ift}} = 4\alpha \Big/ \sqrt{M_\infty^2 - 1} \quad C_{D_{rag}} = 4\alpha^2 \Big/ \sqrt{M_\infty^2 - 1}$$
$$S = cb \quad q_\infty = \tfrac{1}{2}\rho_{air} V_\infty^2 \tag{7.12}$$

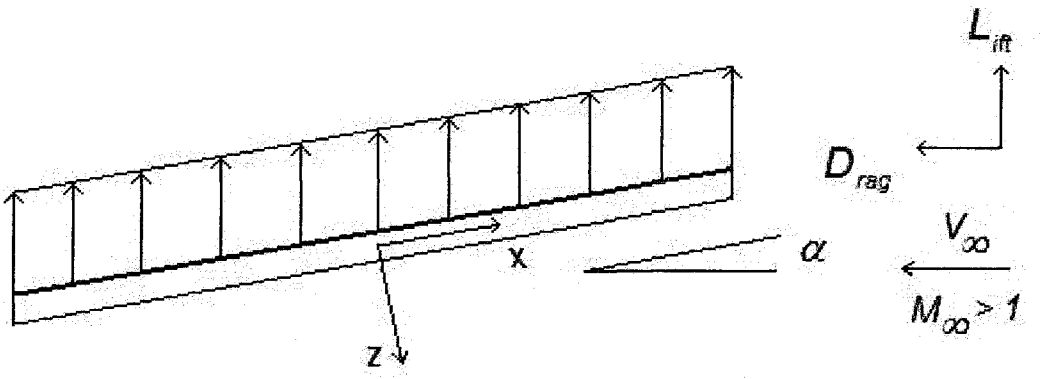


Fig. 7.3. Lift distribution for a supersonic flat plate. Notice that lift acts up, while it is necessary to sum forces in the x and z axis system. Because of this, a coordinate transformation is necessary.

Since this supersonic beam is perfectly balanced, no aerodynamic moment is generated about the origin from the lift distribution. Hence, no moment is generated from AOA, so one can safely assume $C_{m\alpha} = 0$. However, there will be some amount of damping involved and $C_{mq} < 0$. Unfortunately, this C_{mq} is more difficult to quantify and no theoretical prediction is available, so a somewhat reasonable approximation will be made in eq. 7.17.

Also, note that there is no drag produced at $\alpha = 0$, which is not very realistic. Now, it is true that at $\alpha = 0$, there is no *induced* drag present. However, there should be some *parasitic* drag present at $\alpha = 0$. So, once again, a somewhat reasonable number will be added in the numeric example after linearization.

In order to get a clear picture of the dynamics involved in this system, the equations of motion will be linearized. To begin the linearization, the vehicle will be trimmed with a flight path angle of 0. In other words, the lift will be set exactly equal to the weight of the vehicle. This will then define equilibrium AOA, α_o, defined in eq. 7.13.

$$\alpha_o = {mg\sqrt{M_\infty^2 - 1}}\Big/{4q_\infty S} \tag{7.13}$$

Supersonic flight conditions usually produce a very small value of α_o and small angle approximations are very easily justified. This may not be true in the case of a very slow speed landing, where a vehicle may have a trimmed angle of attack of up to α_o=10 deg.

Continuing with the linearization, one takes the derivative (i.e. the first variation) of α around w_{ro} and u_{ro}. This is expressed in eq. 7.14 and 7.15, with a small angle approximation.

$$\partial\alpha = \frac{1}{1+\left({w_{ro}}/{u_{ro}}\right)^2}\frac{\partial w_{ro}}{u_{ro}} - \frac{1}{1+\left({w_{ro}}/{u_{ro}}\right)^2}\frac{w_{ro}\partial u_{ro}}{u_{ro}^2} \approx \frac{\partial w_{ro}}{u_{ro}} - \frac{w_{ro}\partial u_{ro}}{u_{ro}^2} \tag{7.14}$$

$$\partial V = \sqrt{u_r^2 + w_r^2} \approx \frac{u_{ro}}{V_\infty}\partial u_r + \frac{w_{ro}}{V_\infty}\partial w_r \tag{7.15}$$

Linearizing the force relationship with the small angle approximation gives the following relation in eq. 7.16, with $\ell_\infty \equiv \sqrt{M_\infty^2 - 1}$ ..

$$\begin{aligned} \partial L_{ift} &= \frac{4q_\infty S}{\ell_\infty}\partial\alpha + \frac{8q_\infty S\alpha_o}{V_\infty \ell_\infty}\partial V \\ \partial D_{rag} &= \frac{8q_\infty S\alpha_o}{\ell_\infty}\partial\alpha + \frac{8q_\infty S\alpha_o^2}{V_\infty \ell_\infty}\partial V \end{aligned}, \quad \begin{bmatrix}\partial F_x \\ \partial F_z\end{bmatrix} = \begin{bmatrix}-1 & \alpha_o \\ -\alpha_o & -1\end{bmatrix}\begin{bmatrix}\partial D_{rag} \\ \partial L_{ift}\end{bmatrix} \tag{7.16}$$

Collecting the forces and linearizing terms from eq. 7.10 gives the following linearized expression in eq. 7.17. The linearization point for 0 flight path angle is defined as: $u_{ro} = V_\infty \cos\alpha_o$, $w_{ro} = V_\infty \sin\alpha_o, \theta_{ro} = \alpha_o$ and $q_{ro} = \eta_{io} = \dot{\eta}_{io} = 0$. Note that in eq. 7.18, the ∂ symbol has been replaced by the Δ symbol. So, for example, $\Delta w_r = w_r - w_{ro}$. Note that volumetric integrals can be reduced to one dimensional integrals as: $\int_{-c/2}^{c/2}\rho_b\Phi_i(x)dV = \int_{-c/2}^{c/2}(\rho_b tb)\Phi_i(x)dx$.

Because the c.g. of the rigid beam/airfoil is directly over the neutral point, $C_{m\alpha} \approx 0$. An approximation of C_{mq} will be derived assuming a local perturbation in AOA can be given by $\alpha_{extra}(x) \approx {}^{q_r x}/_{V_\infty}$. Integrating along the beam gives the following approximation for C_{mq} in eq. 7.17.

$$C_{mq} \approx \frac{-1}{3\ell_\infty}, \quad \frac{\partial M}{\partial q} = \frac{q_\infty Sc}{V_\infty} C_{mq} \tag{7.17}$$

$$\begin{aligned}
m\Delta\dot{u}_r &= -mw_{ro}\Delta q_r + q_\infty S\left(\frac{4\alpha_o^2}{\ell_\infty}\right)\left(\frac{\Delta w_r}{u_{ro}} - \frac{w_{ro}\Delta u_r}{u_{ro}^2}\right) - mg\cos\theta_{ro}\Delta\theta_r \\
m\Delta\dot{w}_r + \sum_{i=1}^{N}\int_{-c/2}^{c/2}\rho_b\Phi_i(x)dV(\Delta\ddot{\eta}_i) &= mu_{ro}\Delta q_r - q_\infty S\left(\frac{4}{\ell_\infty} + \frac{8\alpha_o^2}{\ell_\infty}\right)\left(\frac{\Delta w_r}{u_{ro}} - \frac{w_{ro}\Delta u_r}{u_{ro}^2}\right) \\
&\quad - \frac{q_\infty S}{V_\infty}\left(\frac{8\alpha_o^3}{\ell_\infty} + \frac{8\alpha_o^2}{\ell_\infty}\right)\left(\frac{w_{ro}}{V_\infty}\Delta w_r + \frac{u_{ro}}{V_\infty}\Delta u_r\right) - mg\sin\theta_{ro}\Delta\theta_r \\
I_{yy}\Delta\dot{q}_r - \sum_{i=1}^{N}\int_{-c/2}^{c/2}\rho_b x\Phi_i(x)dV(\Delta\ddot{\eta}_i) &= \frac{q_\infty Sc}{V_\infty} C_{mq}\Delta q_r \\
\int_{-c/2}^{c/2}\rho_b\Phi_i^2(x)dV(\Delta\ddot{\eta}_i) + \int_{-c/2}^{c/2}\rho_b\Phi_i(x)dV(\Delta\dot{w}_r) - \int_{-c/2}^{c/2}\rho_b x\Phi_i(x)dV(\Delta\dot{q}_r) &= \int_{-c/2}^{c/2}\rho_b\Phi_i(x)dV(u_{ro}\Delta q_r) \\
- \int_{-c/2}^{c/2}\rho_b\Phi_i(x)dV(g\sin\theta_{ro}\Delta\theta_r) - q_\infty b\left(\frac{4}{\ell_\infty} + \frac{8\alpha_o^2}{\ell_\infty}\right)\int_{-c/2}^{c/2}\Phi_i(x)dx\left(\frac{\Delta w_r}{u_{ro}} - \frac{w_{ro}\Delta u_r}{u_{ro}^2}\right) \\
- \frac{q_\infty S}{V_\infty}\left(\frac{8\alpha_o^3}{\ell_\infty} + \frac{8\alpha_o^2}{\ell_\infty}\right)\int_{-c/2}^{c/2}\Phi_i(x)dx\left(\frac{w_{ro}}{V_\infty}\Delta w_r + \frac{u_{ro}}{V_\infty}\Delta u_r\right) \\
- \int_{-c/2}^{c/2} EI\left(\Phi_i''(x)\right)^2 dx(\Delta\eta_i)
\end{aligned} \tag{7.18}$$

7.3 A Very Thin Supersonic Flexible Beam with Steady Aerodynamics

Modeshapes for a free-free beam *in a vacuum* can be obtained from ref. [2.7] and are listed in eq. 7.19. Here, cosh and sinh represent hyberbolic sines and cosines.

$$\Phi_k(x) = \vartheta_k\left[\cos\left(\kappa_k\left(x + {}^{c}/_{2}\right)\right) + \cosh\left(\kappa_k\left(x + {}^{c}/_{2}\right)\right)\right] + \left[\sin\left(\kappa_k\left(x + {}^{c}/_{2}\right)\right) + \sinh\left(\kappa_k\left(x + {}^{c}/_{2}\right)\right)\right] \tag{7.19}$$

The κ_k values come from the solution to the transcendental equation described in eq. 7.20.

$$\cos(\kappa_k c)\cosh(\kappa_k c) = 1 \tag{7.20}$$

The ϑ_k value can be calculated once the κ_k value is known via eq. 7.21.

$$\vartheta_k = \left(\cos(\kappa_k c) - \cosh(\kappa_k c)\right) \Big/ \left(\sin(\kappa_k c) + \sinh(\kappa_k c)\right) \tag{7.21}$$

Beam geometry is chosen as a 5cm thick solid piece of aluminum with the physical parameters shown below. This corresponds to a thickness over chord ratio of 1%, which is unreasonably thin.

$$\rho_b = 2770 kg/m^3 \quad E = 70GPa \quad c = 5m$$
$$b = 1m \quad t = 0.05m \quad I = \tfrac{1}{12} bt^3$$
$$M_\infty = 2 \quad V_\infty = 680.6 \quad \rho_{air} = 1.225 kg/m^3$$

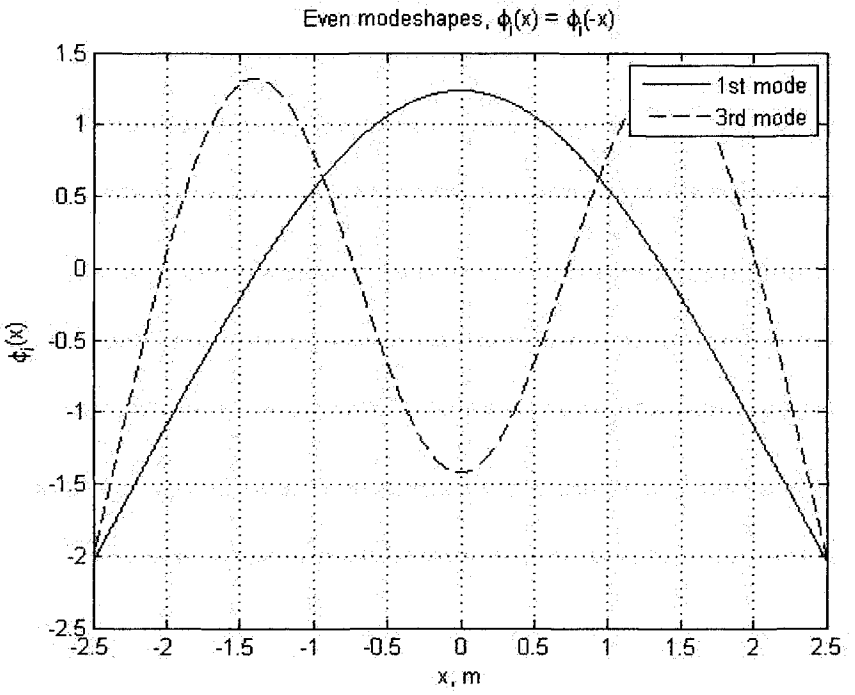


Fig. 7.4. 1st and 3rd modeshape for a free-free beam. Note that due to the fact that these modeshapes are even functions, some terms will naturally become 0.

Computing modeshapes from eq. 7.19-21 gives the first four modeshapes shown in Fig. 7.4 and Fig. 7.5. One should notice from eq. 7.18, that if the quantities $\int_{-c/2}^{c/2} \rho_b \Phi_i(x) dV = \int_{-c/2}^{c/2} (\rho_b tb) \Phi_i(x) dx$ and $\int_{-c/2}^{c/2} \rho_b x \Phi_i(x) dV = \int_{-c/2}^{c/2} (\rho_b tb) x \Phi_i(x) dx$ are non-zero, then the mass matrix of eq. 7.18 will not be perfectly diagonal and there will be cross-coupling between elastic and rigid motions.

However, it turns out that due to symmetry, $\int_{-c/2}^{c/2} x\Phi_i(x)dx = 0$ and $\int_{-c/2}^{c/2} \Phi_i(x)dx = 0$. From Fig. 7.4, one can see that the modeshapes are even functions, and thus $\int_{-c/2}^{c/2} x\Phi_i(x)dx$ must be 0. Also, due to equal positive and negative areas, $\int_{-c/2}^{c/2} \Phi_i(x)dx$ is 0 as well.

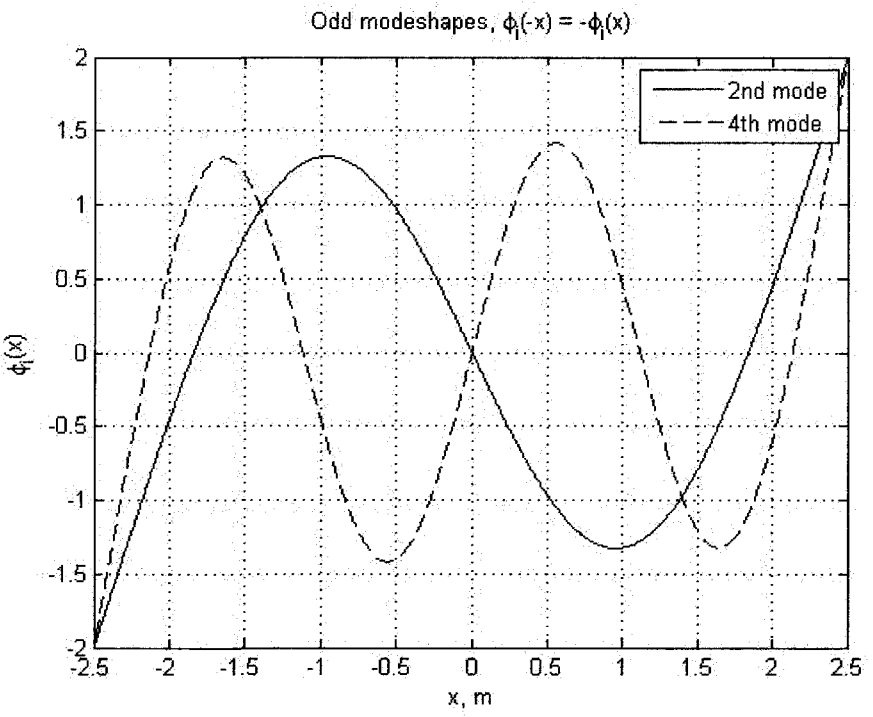


Fig. 7.5. 1st and 3rd modeshape for a free-free beam. Note that due to the fact that these modeshapes are even functions, some terms will naturally become 0.

The same holds true for odd modeshapes, seen in Fig. 7.5. Because they are odd functions, $\int_{-c/2}^{c/2} \Phi_i(x)dx = 0$. Also, because of equal positive and negative areas, $\int_{-c/2}^{c/2} x\Phi_i(x)dx$ is 0 as well.

When $\int_{-c/2}^{c/2} \rho_b x\Phi_i(x)dV = 0$ and $\int_{-c/2}^{c/2} \rho_b \Phi_i(x)dV = 0$, then the mass matrix of eq. 7.18 will be diagonal and the "mean axis" assumption can be employed with confidence. By inspection of eq. 7.18, one can see that flexible dynamics and rigid dynamics can be completely decoupled.

This is a popular assumption because a flexible object *in a vacuum* will exhibit this phenomena and one should expect the natural modes of vibration to give such symmetry as long as the origin is the c.g. of the flexible object. Models which use a "mean axis" assumption have been used for quite some time and can be found in Refs. [3.25].

Amending eq. 7.18 to include the symmetry of the modes gives completely decoupled equations of motion in eq. 7.22. Note that these are linearized equations of motion in eq. 7.22. Mild coupling would be present in the non-linear equations of eq. 7.10 because of the non-zero term $\int_{-c/2}^{c/2} \rho_b \Phi_i^2(x) dV \neq 0$.

$$
\begin{aligned}
m\Delta\dot{u}_r &= -mw_{ro}\Delta q_r + q_\infty S\left(\frac{4\alpha_o^2}{\ell_\infty}\right)\left(\frac{\Delta w_r}{u_{ro}} - \frac{w_{ro}\Delta u_r}{u_{ro}^2}\right) - mg\cos\theta_{ro}\Delta\theta_r \\
m\Delta\dot{w}_r &= mu_{ro}\Delta q_r - q_\infty S\left(\frac{4}{\ell_\infty} + \frac{8\alpha_o^2}{\ell_\infty}\right)\left(\frac{\Delta w_r}{u_{ro}} - \frac{w_{ro}\Delta u_r}{u_{ro}^2}\right) \\
&- \frac{q_\infty S}{V_\infty}\left(\frac{8\alpha_o^3}{\ell_\infty} + \frac{8\alpha_o^2}{\ell_\infty}\right)\left(\frac{w_{ro}}{V_\infty}\Delta w_r + \frac{u_{ro}}{V_\infty}\Delta u_r\right) - mg\sin\theta_{ro}\Delta\theta_r \qquad (7.22) \\
I_{yy}\Delta\dot{q}_r &= \frac{q_\infty Sc}{V_\infty} C_{mq}\Delta q_r \\
\int_{-c/2}^{c/2} \rho_b \Phi_i^2(x) dV(\Delta\ddot{\eta}_i) &= -\int_{-c/2}^{c/2} EI(\Phi_i''(x))^2 dx(\Delta\eta_i)
\end{aligned}
$$

7.4 Example of a Very Thin Supersonic Flexible Beam with Piston Theory

One thing about aeroelasticity which can become very difficult is the unsteady aerodynamic portion of the analysis. Linear modal analysis of elastic structures is quite solvable. Static aerodynamic analysis of rigid structures can also become readily solvable for certain situations like incompressible flow. Honestly combining the two gives a transient aerodynamics problem and elastic structures problem. To the dismay of many, this creates some truly difficult Partial Differential Equations (PDE), even with simplifying assumptions like incompressible flow and linear elasticity.

Most often, the two problems are solved separately, even though the true PDE will not be satisfied exactly. For a similar case of the supersonic beam (with different boundary conditions), Ref. [7.3] obtains modeshapes from an elastic PDE in a vacuum and later inserts aerodynamic forces. This

could potentially lead to inaccuracy. This sub-section will solve the PDE more accurately and explore the level of inaccuracy produced by getting modeshapes in a vacuum. It turns out that the inaccuracy is small.

One thing noticeably absent from the previous section is a description how the elastic deflection, **Ze**, affects lift. There is a simple model available which uses a linearized of "piston theory" [7.3]. Equation 7.22 is a 1st order approximation to piston theory model and is unfortunately not a good physical model of an unstable oscillatory flutter. However, the 1st order approximation of eq. 7.22 can explain stable aeroelastic oscillations. The true piston theory is non-linear and hence it is more difficult to demonstrate that it explains unstable oscillatory flutter. Therefore, this only be a modeshape calculation.

$$\Delta pressure \approx {2q_\infty}/{\ell_\infty}\left({\partial z_e}/{\partial x}+\frac{M_\infty^2-2}{M_\infty^2-1}\frac{1}{V_\infty}{\partial z_e}/{\partial t}\right) \tag{7.22}$$

Including eq. 7.22 into the beam model with no external forces gives the PDE shown in eq. 7.23.

$$\rho_b A\left(\dot{w}_r-u_r q_r-x\dot{q}_r+\ddot{z}_e(x,t)-q_r^2 z_e(x,t)\right)+\frac{2q_\infty b}{\ell_\infty}\left(\frac{\partial z_e(x,t)}{\partial x}+\frac{M_\infty^2-2}{M_\infty^2-1}\frac{1}{V_\infty}\frac{\partial z_e(x,t)}{\partial t}\right) \tag{7.23}$$
$$+\frac{\partial}{\partial x^2}\left(EI\frac{\partial^2 z_e(x,t)}{\partial x^2}\right)=0$$

Assuming that the supersonic beam is trimmed in steady level flight, $\dot{w}_r=\dot{q}_r=q_r=0$, and the PDE reduces to that shown in eq. 7.24.

$$\rho_b A\left(\ddot{z}_e(x,t)\right)+\frac{2q_\infty b}{\ell_\infty}\left(\frac{\partial z_e(x,t)}{\partial x}+\frac{M_\infty^2-2}{M_\infty^2-1}\frac{1}{V_\infty}\frac{\partial z_e(x,t)}{\partial t}\right)+\frac{\partial^2}{\partial x^2}\left(EI\frac{\partial^2 z_e(x,t)}{\partial x^2}\right)=0 \tag{7.24}$$

The elastic deflection, **Ze**, will be represented by the separation of variables in eq. 7.25.

$$z_e(x,t)=\sum_{k=1}^{\infty}\Phi_k(x)\eta_k(t)\approx\sum_{k=1}^{N}\Phi_k(x)\eta_k(t) \tag{7.25}$$

Substituting eq. 7.25 into 7.24 and doing some manipulation yields the following in eq. 7.26.

$$\frac{1}{\eta_k}\left(\rho_b A\ddot{\eta}_k+\frac{M_\infty^2-2}{M_\infty^2-1}\frac{2q_\infty b}{\ell_\infty V_\infty}\dot{\eta}_k\right)=\frac{1}{\Phi_k}\left(-EI\frac{\partial\Phi_k}{\partial x^4}-\frac{2q_\infty b}{\ell_\infty}\frac{\partial\Phi_k}{\partial x}\right) \tag{7.26}$$

We set the right hand side of eq. 7.26 is equal to some unknown constant, N_k. This is done for the left hand side as well. This gives two separate Ordinary Differential Equation, (ODE) listed in eq. 7.27 and eq. 7.28.

$$\rho_b A\ddot{\eta}_k + \frac{M_\infty^2 - 2}{M_\infty^2 - 1}\frac{2q_\infty b}{\ell_\infty V_\infty}\dot{\eta}_k - N_k\eta_k = 0 \tag{7.27}$$

$$-EI\frac{\partial\Phi_k}{\partial x^4} - \frac{2q_\infty b}{\ell_\infty}\frac{\partial\Phi_k}{\partial x} - N_k\Phi_k = 0 \tag{7.28}$$

If N_k is positive in eq. 7.27, then the solution for η must have one real unstable root and one real stable root. Thus, 1st order approximations of piston theory cannot explain unstable oscillatory flutter because any instability must appear on the real axis.

Analysis will be limited to N_k being a negative number and only stable aeroelastic oscillations will be pursued. The value of this unknown constant, N_k, is determined by the specific boundary conditions of the PDE. For the moment, we will just pretend as though N_k is a defined negative number. This means that eq. 7.28 will give a complex conjugate pair and two real roots. Also, eq. 7.27 will give a stable complex conjugate pair with positive damping.

Focusing more on the roots of eq. 7.28, one can get an analytical solution to this fourth order expression with the following roots in eq. 7.29.

$$\begin{aligned}
&h_1 = \frac{2q_\infty b}{\ell_\infty EI} \quad h_2 = \frac{N_k}{EI} \quad h_3 = 27h_1^2 + \sqrt{\left(27h_1^2\right)^2 - 4\left(12h_2^2\right)^2} \\
&h_4 = \frac{2^{1/3}12h_2}{3h_3^{1/3}} + \left(\frac{h_3}{54}\right)^{1/3} \\
&r_{1,2} = -\frac{1}{2}\sqrt{h_4} \pm \frac{1}{2}\sqrt{\frac{2h_1}{\sqrt{h_4}} - h_4} \quad r_{3,4} = -\frac{1}{2}\sqrt{h_4} \pm \frac{1}{2}\sqrt{\frac{-2h_1}{\sqrt{h_4}} - h_4} \\
&r_{1,2} = [a_1, a_2] \quad r_{3,4} = [a_3 + b_3 j, a_3 - b_3 j]
\end{aligned} \tag{7.29}$$

Given the roots in eq. 7.29, we obtain an expression for $\Phi_k(x)$ as follows in eq. 7.30 with N_k.

$$\Phi_k(x) = c_1 e^{a_1(x+c/2)} + c_2 e^{a_2(x+c/2)} + c_3 e^{a_3(x+c/2)} \cos(b_3(x+c/2)) + c_4 e^{a_3 x} \sin(b_3(x+c/2)) \tag{7.30}$$

Now, more expressions are needed to solve for these constants of c_1, c_2, c_3, c_4, and N_k. This is where boundary conditions become useful. The boundary conditions for a free-free beam are listed here, $\Phi_k''(-c/2) = \Phi_k'''(-c/2) = \Phi_k''(c/2) = \Phi_k'''(c/2) = 0$.

Assembling the boundary conditions in a matrix form gives the following result in eq. 7.31. The *BSM* matrix is named so because *BSM* is a big singular matrix with the correct value of N_k.

$$\begin{aligned}
&BSM\phi^T = 0 \quad \phi = [c_1 \quad c_2 \quad c_3 \quad c_4] \\
&BSM_{11} = a_1^2 \quad BSM_{12} = a_2^2 \quad BSM_{13} = a_3^2 - b_3^2 \quad BSM_{14} = 2a_3 b_3 \\
&BSM_{21} = a_1^3 \quad BSM_{22} = a_2^3 \quad BSM_{23} = a_3^3 - 3a_3 b_3^2 \quad BSM_{24} = 3a_3^2 b_3 - b_3^2 \\
&BSM_{31} = a_1^2 e^{a_1 c} \quad BSM_{32} = a_2^2 e^{a_2 c} \\
&BSM_{33} = e^{a_3 c}\left(a_3^2 \cos(b_3 c) - 2a_3 b_3 \sin(b_3 c) - b_3^2 \cos(b_3 c)\right) \\
&BSM_{34} = e^{a_3 c}\left(a_3^2 \sin(b_3 c) + 2a_3 b_3 \cos(b_3 c) - b_3^2 \sin(b_3 c)\right) \\
&BSM_{41} = a_1^3 e^{a_1 c} \quad BSM_{42} = a_2^3 e^{a_2 c} \\
&BSM_{43} = e^{a_3 c}\left(a_3^3 \cos(b_3 c) - 3a_3^2 b_3 \sin(b_3 c) - 3a_3 b_3^2 \cos(b_3 c) - b_3^3 \sin(b_3 c)\right) \\
&BSM_{44} = e^{a_3 c}\left(a_3^3 \sin(b_3 c) + 3a_3^2 b_3 \cos(b_3 c) - 3a_3 b_3^2 \sin(b_3 c) - b_3^3 \cos(b_3 c)\right)
\end{aligned} \tag{7.31}$$

Now one must find the correct values of N_k for which *BSM* is actually a singular matrix and $\det(BSM) = 0$. This is summarized below.

$$\mathbb{N} = \{N_k : \det(BSM) = 0\} \quad k = 1 \ldots \infty \tag{7.32}$$

Possible solution methods to find N_k could include a minimization of $\det(BSM)$ or possibly minimizing smallest eigenvalue of *BSM*. Either solution method can work. The author used MATLAB's fminbnd() optimization subroutine. The ϕ vector is then the eigenvector associated with the eigenvalue of *BSM* which is 0 which is also the null space of *BSM*.

$$\phi = null(BSM) \tag{7.33}$$

The physical parameters for this example are listed below.

$\rho_b = 2770 kg/m^3 \quad E = 70GPa \quad c = 5m \quad b = 1m \quad t = 0.05m \quad I = \frac{1}{12} bt^3$

$M_\infty = 2 \quad V_\infty = 680.6 \quad \rho_{air} = 1.225 kg/m^3$

Table. 7.2. Modeshape parameters with piston theory

Mode	$(-N_k/EI)^{1/4} c$	¢	a_1	a_2	a_3	b_3
1	4.7490	-0.4576 -0.0314 -0.3848 0.8010	-1.0663	0.8174	0.1244	0.9582
2	7.8526	-0.5496 3.9905e-4 -0.5389 0.6418	-1.6154	1.5243	0.0455	1.5712
3	10.9954	-0.5653 -1.2757e-5 -0.5646 0.6013	-2.2222	2.1757	0.0232	2.1992
4	14.1371	-0.5717 4.9154e-7 -0.5714 0.5836	-2.8414	2.8133	0.0141	2.8275

The parameters of the first four modeshapes are listed in table 7.2. Notice that with the first mode, $a_1 \neq a_2$. However, with higher order modes, $a_1 \approx a_2$.

Figures 7.6 and 7.7 show the new modeshapes. Notice that the 1st mode has a noticeable asymmetry associated with it. The 3rd mode is barely affected by the inclusion of the piston theory

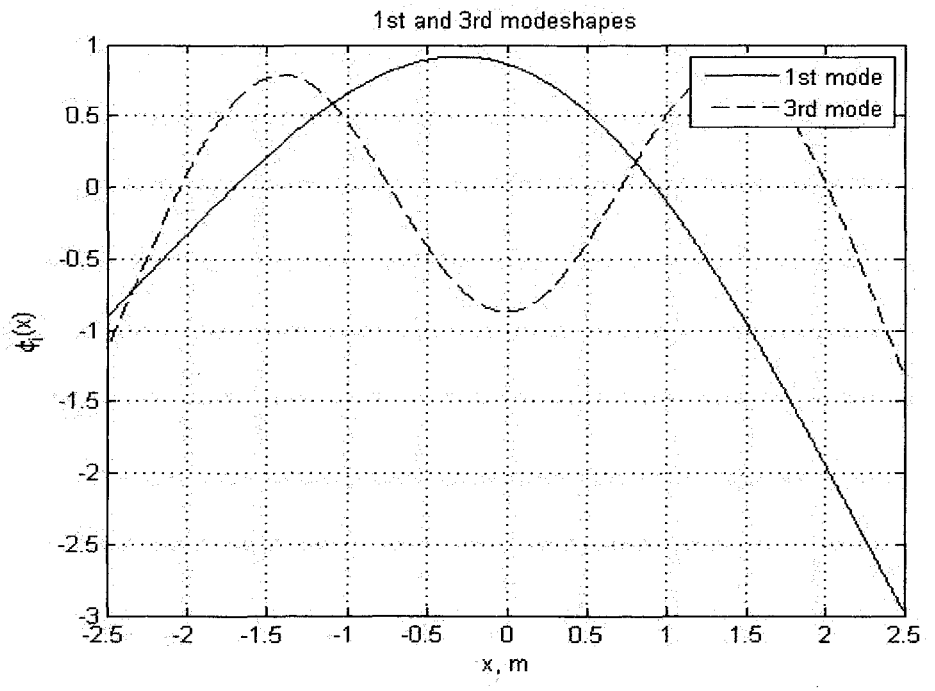


Fig. 7.6. 1st and 3rd modeshape for a free-free beam with the piston theory included. Notice that there is a significant asymmetry associated with the first modeshape. The 3rd modeshape is barely affected by the inclusion of the piston theory.

The 2nd and 4th modeshapes are not that significantly affected by the piston theory.

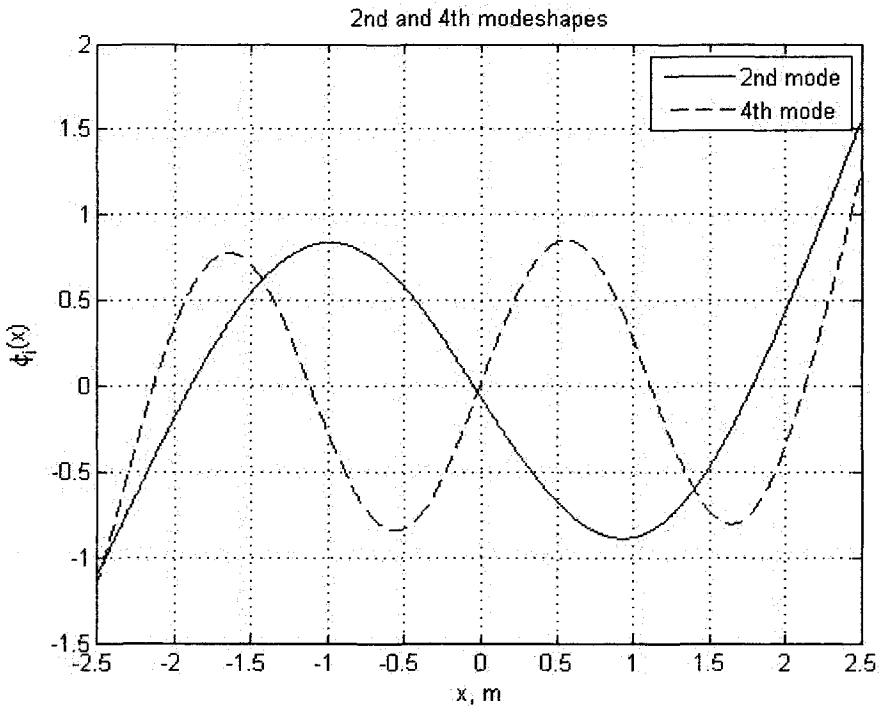


Fig. 7.7. 2nd and 4th modeshapes for a free-free beam with the piston theory included. There is a small amount of asymmetry associated with both modes, but not nearly as much as seen with the 1st mode in Fig. 7.6.

Adding the piston theory forces from eq. 7.22 and inserting them into the total equations of motion gives eq. 7.34. The assumption of orthogonal modeshapes is not present in eq. 7.34, since it is no longer be the case that the modeshapes are orthogonal.

$$
\begin{aligned}
m\Delta\dot{u}_r &= -mu_{ro}\Delta q_r + q_\infty S\left(\frac{4\alpha_o^2}{\ell_\infty}\right)\left(\frac{\Delta w_r}{u_{ro}} - \frac{w_{ro}\Delta u_r}{u_{ro}^2}\right) - mg\cos\theta_{ro}\Delta\theta_r \\
&+ \alpha_o\frac{M_\infty^2-2}{M_\infty^2-1}\frac{2q_\infty b}{\ell_\infty V_\infty}\sum_{j=1}^{N}\int_{-c/2}^{c/2}\Phi_j(x)dx(\Delta\dot{\eta}_i) + \alpha_o\frac{2q_\infty b}{\ell_\infty}\sum_{j=1}^{N}\int_{-c/2}^{c/2}\Phi_j'(x)dx(\Delta\eta_i) \\
m\Delta\dot{w}_r &+ \sum_{j=1}^{N}\int_{-c/2}^{c/2}\rho_b\Phi_j(x)dV(\Delta\ddot{\eta}_i) = mu_{ro}\Delta q_r - q_\infty S\left(\frac{4}{\ell_\infty}+\frac{8\alpha_o^2}{\ell_\infty}\right)\left(\frac{\Delta w_r}{u_{ro}} - \frac{w_{ro}\Delta u_r}{u_{ro}^2}\right) \\
&- \frac{q_\infty S}{V_\infty}\left(\frac{8\alpha_o^3}{\ell_\infty}+\frac{8\alpha_o^2}{\ell_\infty}\right)\left(\frac{w_{ro}}{V_\infty}\Delta w_r + \frac{u_{ro}}{V_\infty}\Delta u_r\right) - mg\sin\theta_{ro}\Delta\theta_r \\
&- \frac{M_\infty^2-2}{M_\infty^2-1}\frac{2q_\infty b}{\ell_\infty V_\infty}\sum_{j=1}^{N}\int_{-c/2}^{c/2}\Phi_j(x)dx(\Delta\dot{\eta}_i) - \frac{2q_\infty b}{\ell_\infty}\sum_{j=1}^{N}\int_{-c/2}^{c/2}\Phi_j'(x)dx(\Delta\eta_i) \\
I_{yy}\Delta\dot{q}_r &- \sum_{j=1}^{N}\int_{-c/2}^{c/2}\rho_b x\Phi_i(x)dV(\Delta\ddot{\eta}_i) = \frac{2q_\infty b}{\ell_\infty V_\infty}\frac{M_\infty^2-2}{M_\infty^2-1}\sum_{j=1}^{N}\int_{-c/2}^{c/2}x\Phi_i(x)dx(\Delta\dot{\eta}_i) \\
&+ \frac{2q_\infty b}{\ell_\infty}\sum_{j=1}^{N}\int_{-c/2}^{c/2}x\Phi_i'(x)dx(\Delta\eta_i) + \frac{q_\infty Sc}{V_\infty}C_{mq}\Delta q_r \\
&\sum_{j=1}^{N}\int_{-c/2}^{c/2}\rho_b\Phi_i(x)\Phi_j(x)dV(\Delta\ddot{\eta}_i) + \int_{-c/2}^{c/2}\rho_b\Phi_i(x)dV(\Delta\dot{w}_r) - \int_{-c/2}^{c/2}\rho_b x\Phi_i(x)dV(\Delta\dot{q}_r) = \\
&\int_{-c/2}^{c/2}\rho_b\Phi_i(x)dV(u_{ro}\Delta q_r) - \int_{-c/2}^{c/2}\rho_b\Phi_i(x)dV(g\sin\theta_{ro}\Delta\theta_r) \\
&- q_\infty b\left(\frac{4}{\ell_\infty}+\frac{8\alpha_o^2}{\ell_\infty}\right)\int_{-c/2}^{c/2}\Phi_i(x)dx\left(\frac{\Delta w_r}{u_{ro}} - \frac{w_{ro}\Delta u_r}{u_{ro}^2}\right) \\
&- \frac{q_\infty b}{V_\infty}\left(\frac{8\alpha_o^3}{\ell_\infty}+\frac{8\alpha_o^2}{\ell_\infty}\right)\int_{-c/2}^{c/2}\Phi_i(x)dx\left(\frac{w_{ro}}{V_\infty}\Delta w_r + \frac{u_{ro}}{V_\infty}\Delta u_r\right) - \sum_{j=1}^{N}\int_{-c/2}^{c/2}EI\left(\Phi_i''(x)\Phi_j''(x)\right)dx(\Delta\eta_i) \\
&- \frac{2q_\infty b}{\ell_\infty}\sum_{j=1}^{N}\int_{-c/2}^{c/2}\Phi_i(x)\Phi_j'(x)dx(\Delta\eta_j) - \frac{2q_\infty b}{\ell_\infty V_\infty}\frac{M_\infty^2-2}{M_\infty^2-1}\sum_{j=1}^{N}\int_{-c/2}^{c/2}\Phi_i(x)\Phi_j(x)dx(\Delta\dot{\eta}_i) \\
&\Delta\dot{\eta}_k = \Delta\dot{\eta}_k \\
&\Delta\dot{\theta}_r = \Delta q_r
\end{aligned}
\tag{7.34}
$$

Under a mean axis assumption, $\int_{-c/2}^{c/2}\rho_b\Phi(x)dV \approx \int_{-c/2}^{c/2}\rho_b x\Phi(x)dV \approx 0$, and all modeshapes are orthogonal. The equations using mean axis assumptions are listed below in eq. 7.35.

$$
\begin{aligned}
m\Delta\dot{u}_r &= -mu_{ro}\Delta q_r + q_\infty S\left(\frac{4\alpha_o^2}{\ell_\infty}\right)\left(\frac{\Delta w_r}{u_{ro}} - \frac{w_{ro}\Delta u_r}{u_{ro}^2}\right) - mg\cos\theta_{ro}\Delta\theta_r \\
&+ \alpha_o\frac{M_\infty^2-2}{M_\infty^2-1}\frac{2q_\infty b}{\ell_\infty V_\infty}\sum_{j=1}^{N}\int_{-c/2}^{c/2}\Phi_j(x)dx(\Delta\dot{\eta}_i) + \alpha_o\frac{2q_\infty b}{\ell_\infty}\sum_{j=1}^{N}\int_{-c/2}^{c/2}\Phi_j'(x)dx(\Delta\eta_i) \\
m\Delta\dot{w}_r &= mu_{ro}\Delta q_r - q_\infty S\left(\frac{4}{\ell_\infty} + \frac{8\alpha_o^2}{\ell_\infty}\right)\left(\frac{\Delta w_r}{u_{ro}} - \frac{w_{ro}\Delta u_r}{u_{ro}^2}\right) \\
&- \frac{q_\infty S}{V_\infty}\left(\frac{8\alpha_o^3}{\ell_\infty} + \frac{8\alpha_o^2}{\ell_\infty}\right)\left(\frac{w_{ro}}{V_\infty}\Delta w_r + \frac{u_{ro}}{V_\infty}\Delta u_r\right) - mg\sin\theta_{ro}\Delta\theta_r \\
&- \frac{M_\infty^2-2}{M_\infty^2-1}\frac{2q_\infty b}{\ell_\infty V_\infty}\sum_{j=1}^{N}\int_{-c/2}^{c/2}\Phi_j(x)dx(\Delta\dot{\eta}_i) - \frac{2q_\infty b}{\ell_\infty}\sum_{j=1}^{N}\int_{-c/2}^{c/2}\Phi_j'(x)dx(\Delta\eta_i) \\
I_{yy}\Delta\dot{q}_r &= \frac{2q_\infty b}{\ell_\infty V_\infty}\frac{M_\infty^2-2}{M_\infty^2-1}\sum_{j=1}^{N}\int_{-c/2}^{c/2}x\Phi_i(x)dx(\Delta\dot{\eta}_i) \\
&+ \frac{2q_\infty b}{\ell_\infty}\sum_{j=1}^{N}\int_{-c/2}^{c/2}x\Phi_i'(x)dx(\Delta\eta_i) + \frac{q_\infty Sc}{V_\infty}C_{mq}\Delta q_r \\
\int_{-c/2}^{c/2}\rho_b\Phi_i(x)^2dV(\Delta\ddot{\eta}_i) &= -q_\infty b\left(\frac{4}{\ell_\infty} + \frac{8\alpha_o^2}{\ell_\infty}\right)\int_{-c/2}^{c/2}\Phi_i(x)dx\left(\frac{\Delta w_r}{u_{ro}} - \frac{w_{ro}\Delta u_r}{u_{ro}^2}\right) \\
&- \frac{q_\infty b}{V_\infty}\left(\frac{8\alpha_o^3}{\ell_\infty} + \frac{8\alpha_o^2}{\ell_\infty}\right)\int_{-c/2}^{c/2}\Phi_i(x)dx\left(\frac{w_{ro}}{V_\infty}\Delta w_r + \frac{u_{ro}}{V_\infty}\Delta u_r\right) - \sum_{j=1}^{N}\int_{-c/2}^{c/2}EI\left(\Phi_i''(x)\right)^2dx(\Delta\eta_i) \\
&- \frac{2q_\infty b}{\ell_\infty}\sum_{j=1}^{N}\int_{-c/2}^{c/2}\Phi_i(x)\Phi_j'(x)dx(\Delta\eta_j) \\
\Delta\dot{\eta}_k &= \Delta\dot{\eta}_k \\
\Delta\dot{\theta}_r &= \Delta q_r
\end{aligned}
\tag{7.35}
$$

7.5 Sensor Placement Strategies for the Very Thin Supersonic Flexible Beam

This section makes some suggestions for the placement of velocity and pitch sensors (equivalently accelerometers and rate gyros) for the theoretical supersonic flexible beam. Knowledge of the modeshapes will be used to guide the placement of these sensors.

For most subsonic aircraft in most flight conditions, the internal flexible and rigid dynamics are mostly decoupled (i.e. the dynamics in the A matrix). What can become a more important issue is the tendency of sensors to "feel" these flexible motions (i.e. the dynamics in the C matrix). These flexible dynamics will inevitably be fed back into the closed loop and could potentially cause instability of A-

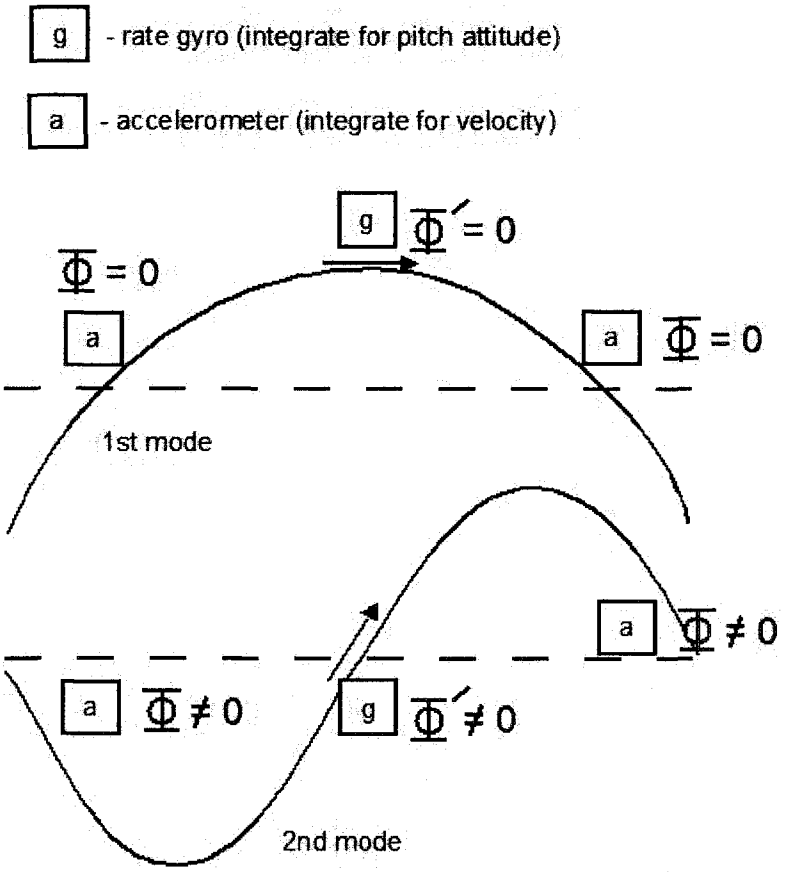


Fig. 7.8. Sensor placement strategy to eliminate feeding back the 1st mode. Here, we put the accelerometers near the nodes of the 1st mode so that there is no displacement there. We also put the rate gyro where the slope of the 1st modeshape is nearly 0. However, the 2nd mode will become noticeable to the sensors.

BKC if a high enough gain, K, is used. Parts of this section will show some sensor placement ideas to make flexible modes mostly unobservable, so that they do not cause instability.

We visually show a strategy to avoid picking up oscillations of the first mode by the sensors in Fig. 7.8. One may try to put the accelerometers near the nodes of the first mode such that there will be very little displacement there. Also, since the first mode has no slope at the center, $\Phi_1'(0) = \partial\Phi_1/\partial x(0) \approx 0$, it would be a good spot to put the rate gyro. However, notice that this sensor scheme will inevitably pick up vibrations from the second mode.

Thus, one can see that this strategy will work well to eliminate feeding back a well known single mode. Suppose that accelerometers do not cost very much and there are many available. Now, one can use the mean axis property such that $\int_V \rho_b(\Phi_k(x))dV \approx \sum_j \rho_{bj}(\Phi_k(x_j))dV_j \approx \sum_j \rho_{bj}(a_j)dV_j \approx 0$ for each mode. This will of course use many accelerometers, so this will inevitably be more expensive than just using one or two accelerometers.

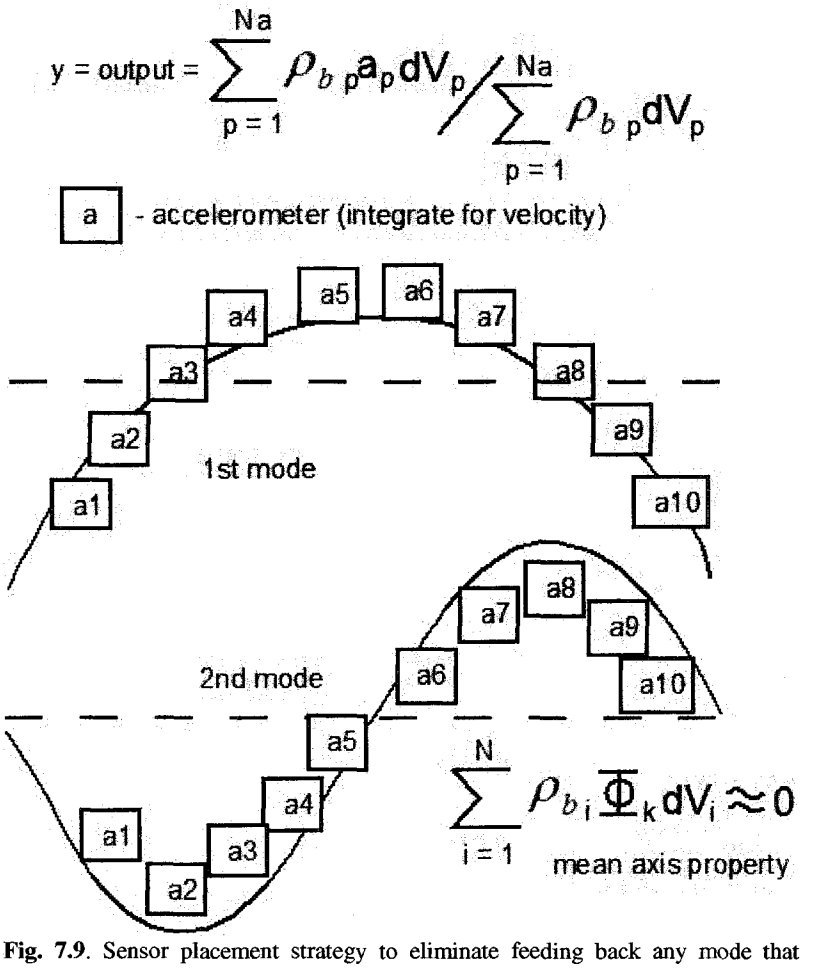


Fig. 7.9. Sensor placement strategy to eliminate feeding back any mode that exhibits mean axis properties. This procedure can be extended to non-uniform density and complex geometries with many accelerometers.

This strategy is shown in Fig. 7.9, where the mean axis property could theoretically be used to diminish the feed back of any flexible mode. Of course, as seen in section 7.3 the mean axis assumption is not strictly true, but is a pretty good approximation.

The author is not aware of this strategy being employed on any aircraft. Recent sensor blending techniques for flexible aircraft have focused on phase margin criteria [7.4,7.5], but not on a physical impression of the mean axis assumptions. Earlier efforts by trial and error have been employed experimentally [3.12]. Another analytical result for flexible structures *in a vacuum* is that of co-located actuators and sensors [7.6]. Theoretically, if an accelerometer was placed at the end of the elevator the dynamics should be stable and minimum phase. Unfortunately, this creates low frequency NMP zeros from rigid body aerodynamics. A different scheme shown in Fig. 7.9 might be an option.

Neither the approach shown in Fig. 7.9 nor the collocated actuator and sensor method works for pitch attitude sensors because they are effected by the derivative of the modeshape $\Phi'(x)$. Unfortunately, one cannot use any mean axis properties to combat the problem of by rate gyros sensing flexible modes.

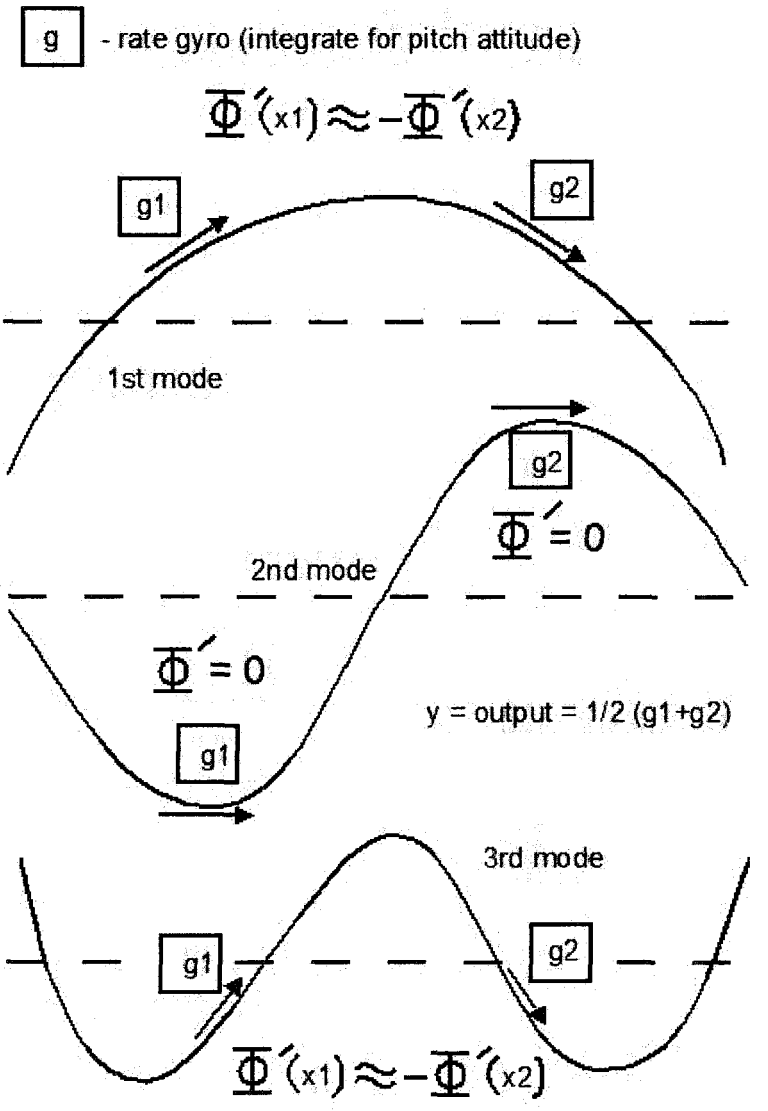


Fig. 7.10. Sensor placement strategy to eliminate feeding back flexible modes from rate gyros. This has not been very successful so far.

To place a single attitude sensor one can simply mitigate $\Phi'(x)$ across one particular mode as seen in Fig. 7.8. Another option is shown in Fig. 7.10. Here, we see that we can mitigate the influence of the first three modes with two rate gyros by simply placing them in the right spot and then summing the rate gyros together. With the supersonic beam of uniform density, one can easily exploit symmetry. Of course, with an irregular geometry, rate gyro placement may be more difficult.

Now we will investigate what these strategies in Fig. 7.9 and Fig. 7.10 can do for feedback control of the supersonic beam. We add a fictionally fast "elevator" point force at the trailing edge of the beam. Using the dirac delta function, this force can be added into the equations of motion. The A matrix used here is taken from eq. 7.34, and the B matrix is listed in eq. 7.37. The C matrix is listed in eq. 7.39.

Downward velocity, $w(x_p)$, and pitch attitude, $\theta_p(x_p)$, at point x_p is given in eq. 7.36.

$$\alpha(x_p) = {}^{1}\!/_{U_0}\left(w_r - q_r x_p + \sum_{k=1}^{N}\Phi_k(x_p)\dot{\eta}_k\right),\quad \theta(x_p) = \theta_r - \sum_{k=1}^{N}\Phi_k'(x_p)\eta_k \qquad (7.36)$$

The relations in eq. 7.36 help form the C matrix. Remember that we will be lumping together the velocity and pitch attitude at several points, not just one.

The B matrix is listed below in eq. 7.37

$$B = M^{-1}\begin{bmatrix} 0 \\ F_{z\delta e} \\ F_{z\delta e}\, c/2 \\ F_{z\delta e}\Phi_1\left(-c/2\right) \\ \vdots \\ F_{z\delta e}\Phi_N\left(-c/2\right) \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{7.37}$$

Here, $F_{z\delta e} = -1000$ is assigned arbitrarily as the force per radian of the fictionally fast elevator. The M matrix is the mass matrix and is formed by the total mass of the beam and the individual modeshapes. The M matrix is shown in eq. 7.38.

$$M = \begin{bmatrix} m_r & 0 & 0 & 0 & \cdots & 0 \\ 0 & m_r & 0 & \int_V \rho_b \Phi_1 dV & \cdots & \int_V \rho_b \Phi_N dV \\ 0 & 0 & I_{yyr} & -\int_V \rho_b x \Phi_1 dV & \cdots & -\int_V \rho_b x \Phi_N dV \\ 0 & \int_V \rho_b \Phi_1 dV & -\int_V \rho_b x \Phi_1 dV & \int_V \rho_b \Phi_1 \Phi_1 dV & \cdots & \int_V \rho_b \Phi_1 \Phi_N dV \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \int_V \rho_b \Phi_1 dV & -\int_V \rho_b \Phi_N dV & \int_V \rho_b \Phi_N \Phi_1 dV & \cdots & \int_V \rho_b \Phi_1 \Phi_N dV \end{bmatrix} \tag{7.38}$$

The C matrix can be realized below in eq. 7.39.

$$C=\begin{bmatrix} C_\theta \\ C_\alpha \end{bmatrix} \quad C_\theta=\begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 & -\sum_{p=1}^{Ng}\Phi_1'(x_p) & \dots & -\sum_{p=1}^{Ng}\Phi_N'(x_p) & 1 \end{bmatrix}$$

$$C_\alpha = \Big/\left(U_o\sum_{p=1}^{Na}\rho_{bp}dV_p\right)\begin{bmatrix} 0 & \sum_{p=1}^{Na}\rho_{bp}dV_p & -\sum_{p=1}^{Na}\rho_{bp}x_pdV_p & \sum_{p=1}^{NG}\rho_{bp}\Phi_1(x_p)dV_p & \dots \end{bmatrix} \tag{7.39}$$

$$\sum_{p=1}^{NG}\rho_{bp}\Phi_N(x_p)dV_p \quad 0 \quad \dots \quad 0 \quad 0\Big]$$

$$x=[\Delta u_r \quad \Delta w_r \quad \Delta q_r \quad \Delta\dot{\eta}_1 \quad \dots \quad \Delta\dot{\eta}_N \quad \Delta\eta_1 \quad \dots \quad \Delta\eta_N \quad \Delta\theta_r]$$

Data for the state space of the supersonic beam from section 7.3 with the first 4 modes is listed below in eq.'s 7.40-42. Here, *Na*=11 evenly spaced accelerometers and *Ng*=2 rate gyros are used at x = +1 and -1 meters.

$$A=\begin{bmatrix} -0.01 & 0.0 & -1.4113 & 0 & 0 & 0 \\ 0.0144 & -6.9511 & 680.3728 & -14.9732 & 2.1434 & -0.0458 \\ -0.0 & 0.0 & -1.222 & -1.2121 & 0.1735 & -0.0037 \\ -0.0 & -0.0 & 0.8385 & -15.8605 & 0.2803 & -0.006 \\ 0.0 & -0.0 & 0.2173 & 0.7195 & -14.005 & 0.0022 \\ 0.0 & -0.0 & 0.0613 & 0.1740 & -0.0249 & -13.9015 \\ 0.0 & 0 & 0.0203 & 0.1012 & -0.0145 & 0.0003 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \dots$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -9.81 \\ 0.1813 & -4.63x10^3 & 4.93x10^3 & -0.4051x10^3 & -4.382x10^3 & -0.0203 \\ 0.0147 & -0.3783x10^3 & 0.397x10^3 & -0.0329x10^3 & 0.3532x10^3 & 0.0 \\ 0.0237 & -0.0491x10^5 & 637.544 & -51.3606 & 569.0519 & 0.0 \\ -0.0087 & 232.55 & -0.3226x10^5 & 17.3091 & -206.9202 & 0.0 \\ -0.0021 & -3.345 & -88.4725 & -1.231x10^5 & -69.0065 & 0.0 \\ -13.903 & 42.6278 & -25.1577 & -0.1524 & -3.365x10^5 & 0.0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{7.40}$$

$$B = [0 \quad -1.6144 \quad -2.1647 \quad 0.0456 \quad -3.6363 \quad -4.5721 \quad 4.7185 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \quad (7.41)$$

$$C_\theta = [0 \quad \ldots \quad 0 \quad -0.6933 \quad 0.7277 \quad -1.0102 \quad -1.603 \quad 1.00]$$

$$C_\alpha = [0.0 \quad 1.469x10^{-3} \quad 0.0 \quad -6.79x10^{-4} \quad 9.20x10^{-5} \quad -1.97x10^{-4} \quad 1.389x10^{-5} \quad 0 \quad \ldots \quad 0] \quad (7.42)$$

The benefit of using multiple accelerometers can be clearly seen by inspecting Fig. 7.11 and Fig. 7.12. With a single accelerometer at the leading edge, the high frequency dynamics clearly exhibit non-minimum phase behavior with very many unstable zeros as in Fig. 7.11.

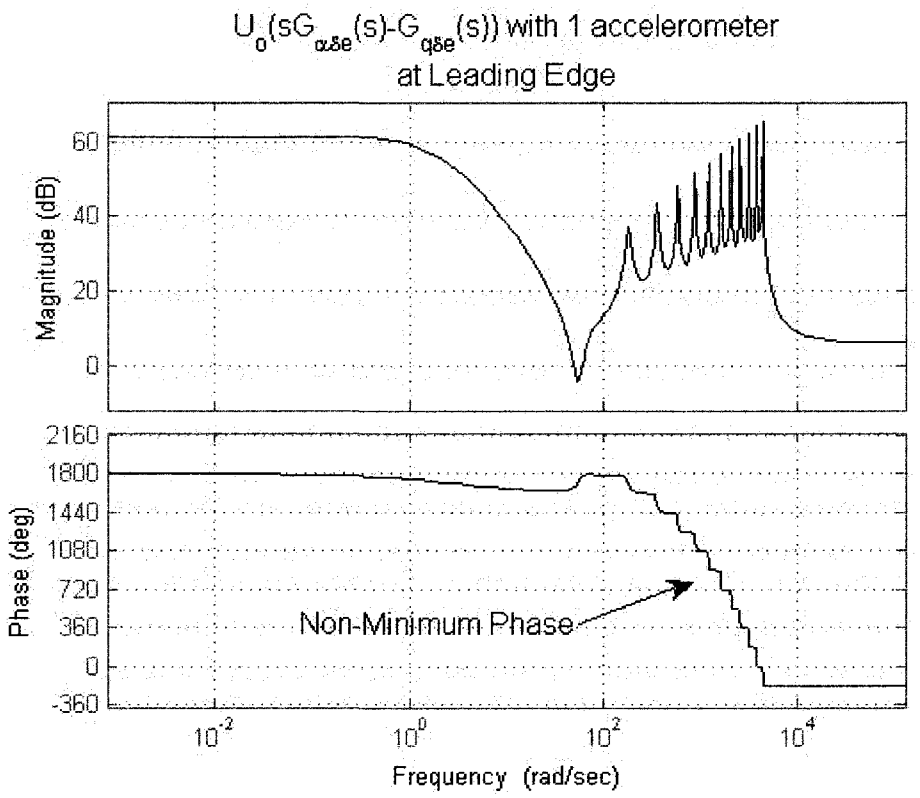


Fig. 7.11. Placing one accelerometer at the leading edge is not a good spot. Almost all the flexible modes are picked up here and almost all of them are non-minimum phase.

By contrast, a mostly minimum phase behavior can be seen in Fig. 7.12 where 101 accelerometers were used. However, there is still one unstable zero that shows up in Fig. 7.12, but it is at a low frequency and

is associated with the rigid body mode.

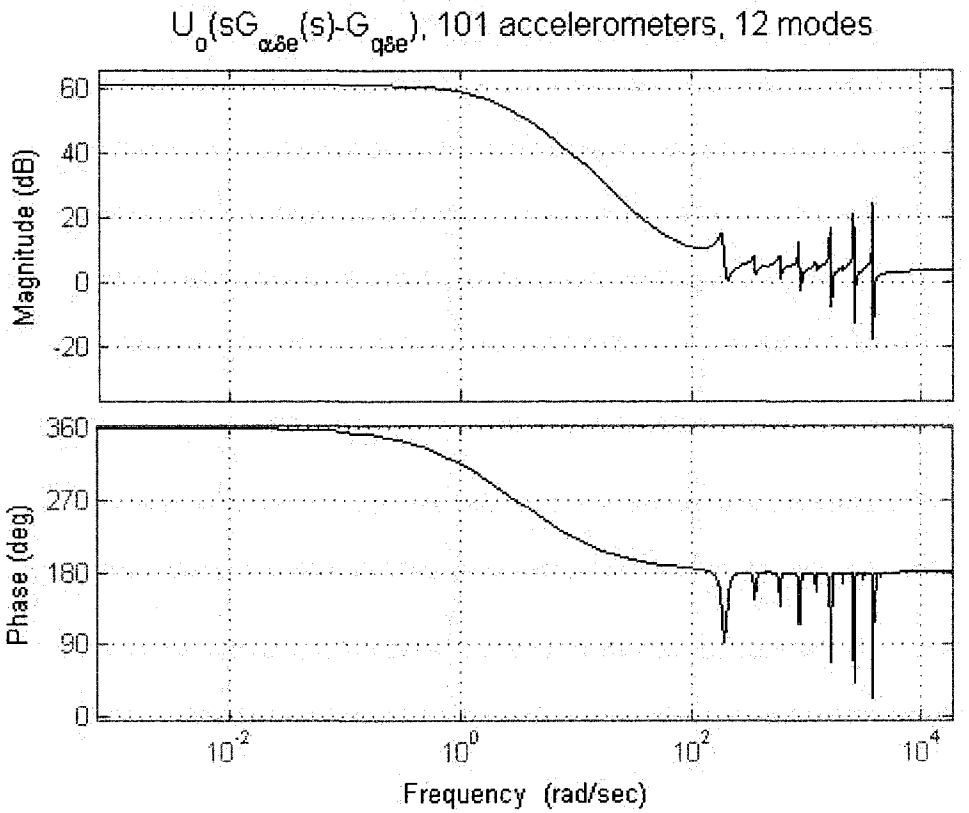


Fig. 7.12. Many flexible modes have closely matched pole-zero pairs with this scheme of 101 accelerometers placed evenly across the uniform beam. There is one unstable zero here which is hard to spot. To see it, notice that there is a -20db/dec. slope with a 180 degree drop in phase from 1 rad./sec. to 100 rad./sec.

Things are different for rate gyro placement. There appears not to be much benefit to using multiple rate gyros and averaging them together. It appears to be more important to place them away from the leading edge and more towards the trailing edge. It is unlikely that one will be able to make flexible modes nearly unobservable to rate gyros. It is more important to try to have as few unstable zeros as possible. Figure 7.13 shows the bode plot with two rate gyros, one at x = 1 and one at x = -1.

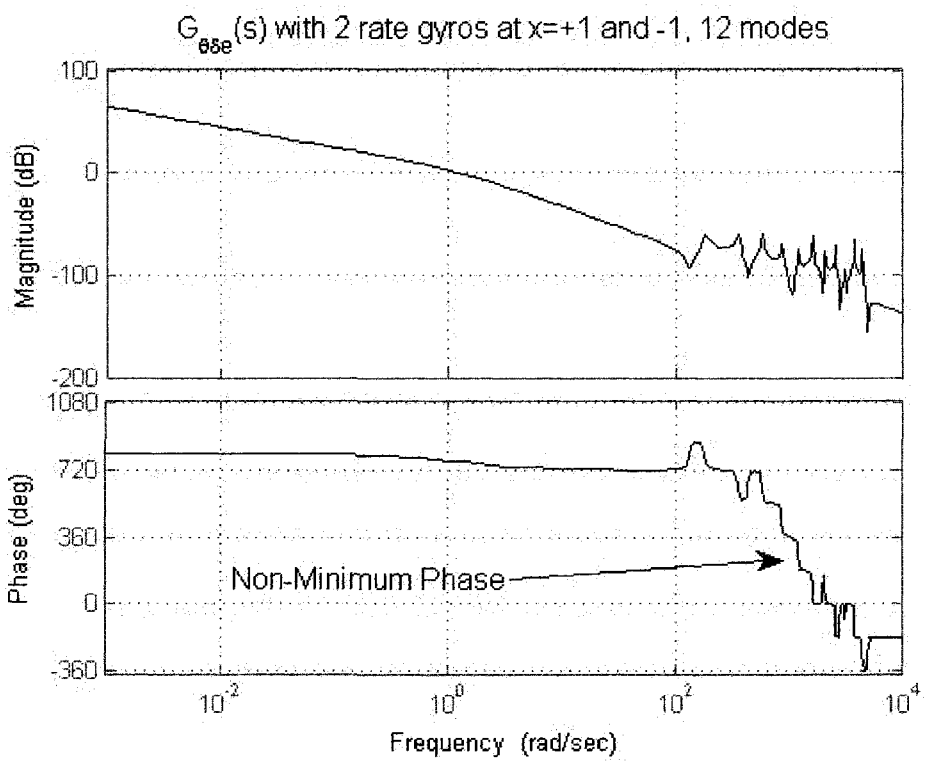


Fig. 7.13. This placement scheme produces many unstable zeros, which is undesirable. This only seems to give good behavior for the first two modes. The flexible modes are still very observable. X position are in meters.

This produces a highly non-minimum phase behavior, there are many unstable zeros.

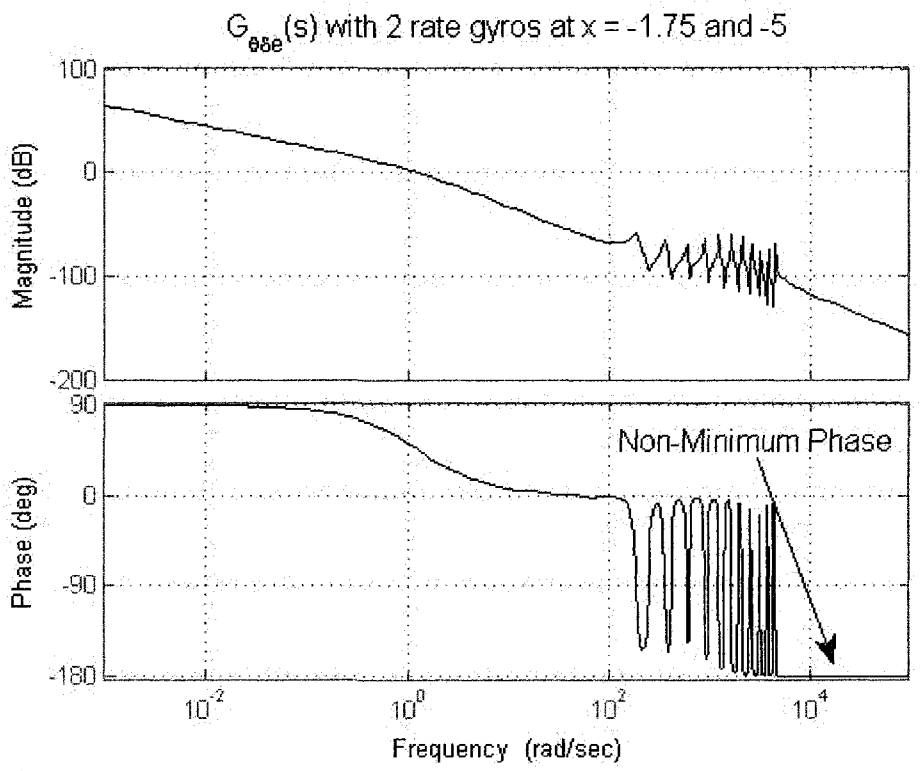


Fig. 7.14. This placement scheme produces a better result, with only one unstable zero. It is better to put the rate gyros closer to trailing edge, where the fictionally fast elevator is located. X position is in meters.

An improvement can be seen by placing the rate gyros closer to the trailing edge as in Fig 7.14. Here, there is only one unstable zero at very high frequency.

From this data, it seems as though it is not possible to mask all flexible deformations by sensor placement alone. Using many accelerometers is helpful, but not perfect. For accelerometer placement, there is no convenient solution.

For a vehicle with an elevator at the tail, one might consider want to placing rate gyro sensors toward the rear of the vehicle, not the front.

Additionally, one may want to factor in the cost associated with 100 accelerometers, making some schemes fairly unreasonable.

7.6 Digital Control Strategy for Very Thin Supersonic Flexible Beam

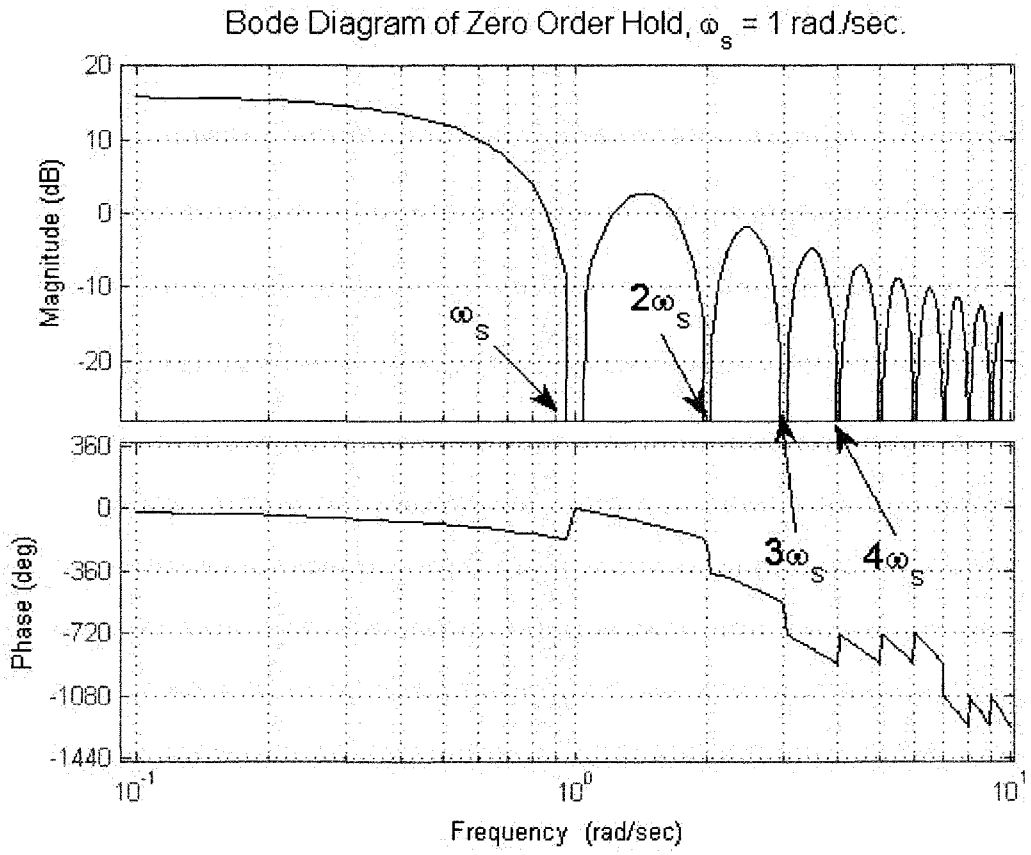


Fig. 7.15. This is the bode plot of the zero order hold, $zoh(s) = {1 - e^{-Ts}}/{s}$, used in digital control. Here, $\omega_s = {2\pi}/{T}$, and T is the sampling time. At integer values of the sampling frequency, there will be infinite attenuation of the output signal.

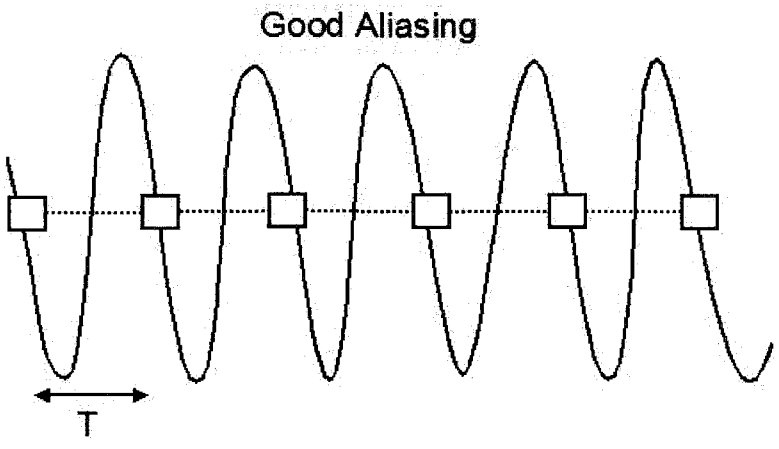


Fig. 7.16. With regard to flexible dynamics, it is best not to be able to sense them at all. This figure shows what good aliasing would be like in the time domain. The flexible mode and the sampling rate, *T*, both have the same period.

In the event that crossover frequencies are about a factor 6 to 10 below the lowest structural mode one can simply use aliasing to avoid feeding back some, but not all, of these structural modes. Refer to Fig. 7.15 which shows the bode plot of the zero order

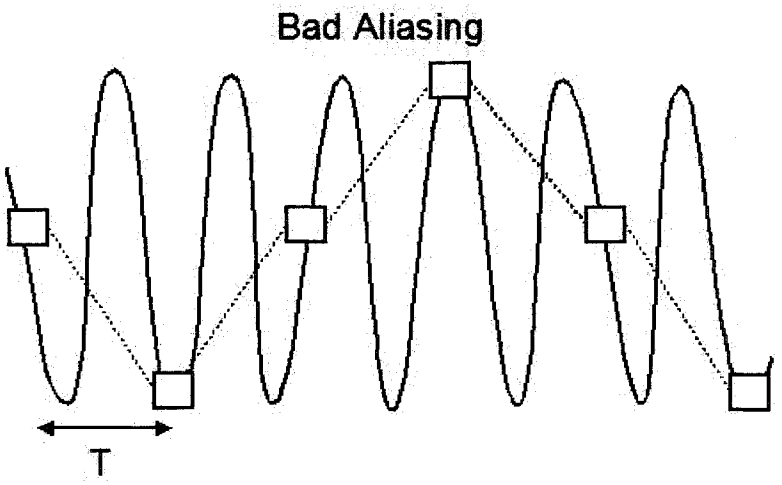


Fig. 7.17. Unless one gets lucky, as one does with this supersonic beam, some flexible modes will be aliased in a bad way. One remedy is to use an analog "anti-aliasing" filter.

hold. Notice that at integer values of the sampling frequency, ω_s, the bode plot is attenuated to 0. Any dynamics faster than the sampling frequency, ω_s, will be aliased. Depending upon whether they are closer to ω_s or $\frac{3}{2}\omega_s$ will effect whether those dynamics are amplified or attenuated. Flexible dynamics that occur at ω_s will have a good kind of aliasing. Flexible dynamics that occur at $\frac{3}{2}\omega_s$ will exhibit a bad kind of aliasing. To help attenuate the dynamics at $\frac{3}{2}\omega_s$ and the other peaks in between, one can use an *analog* filter to help attenuate these aliased dynamics further. Of course, such an analog filter often called an "anti-aliasing filter" is not absolutely necessary for digital control, but it can help smooth out oscillations should they occur.

The continuous dynamics of the supersonic beam with four modes and rate gyros at x = -1.75m and -5m is listed in eq. 7.43.

$$G_{\theta\delta_e}(s)=\frac{6.1227\left(s^2+15.43s+4964\right)\left(s-168.3\right)\left(s+163.3\right)}{s\left(s+1.284\right)\left(s^2+15.78s+4901\right)\left(s^2+14.03s+3.226x10^4\right)}\cdots$$
$$\frac{\left(s^2+15.71s+7.098x10^4\right)\left(s^2+14.76s+2.442x10^5\right)}{\left(s^2+13.9s+1.231x10^5\right)\left(s^2+13.9s+3.365x10^5\right)} \tag{7.43}$$

The first four structural poles occur at frequencies of about 70, 180, 350, and 580 rad./sec., which is a very auspicious set of numbers! We decide to sample at T = 0.1 seconds, approximately $\omega_s \approx 60$ rad./sec. We notice that we can use the good kind of aliasing here (70 is close to 60, 180 is thrice 60, 350 is close to 6 times 60).

Using MATLAB's z-transform function, c2d( ), we use the zero order hold to transform the plant into the z-domain. The result is shown in eq. 7.44.

$$G_{\theta\dot{\delta}e}(z,0.1)=\frac{-0.01045(z+0.9956)(z^2+0.8595z+0.245)(z^2-0.7107z+0.2064)}{(z-1)(z-0.8795)(z^2+0.8648z+0.249)(z^2-0.708z+0.2065)}\cdots$$
$$\frac{(z^2-0.6146z+0.2444)(z^2-0.1158z+0.2487)}{(z^2-0.6155z+0.2459)(z^2-0.1160z+0.2490)} \tag{7.44}$$

To design a controller we simply use MATLAB's sisotool( ) function to design a controller that achieves a bandwidth of about 6 rad./sec. and is listed below.

$$C_{\dot{\delta}e}(z,0.1)=\frac{-17(z-0.88)}{z}\quad L(z,0.1)\approx\frac{-0.178(z+0.9956)}{z(z-1)} \tag{7.45}$$

Simulink implementation of the control loop with a fictionally fast elevator is shown below in Fig. 7.18.

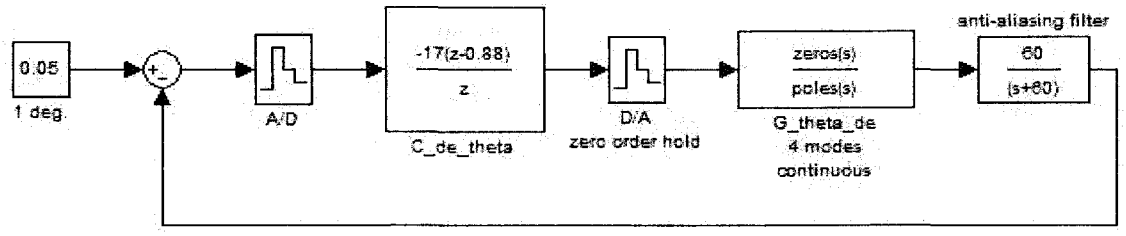


Fig. 7.18. Simulink implementation of the simple control loop is shown here with an "anti-aliasing filter".

7.7 3-D Dynamics of a Maneuvering Flexible Vehicle with a Mean Axis

This section details the derivation of the 3-D dynamics of a maneuvering flexible vehicle. We will first use the mean axis assumptions [3.25] and then later add more terms in with the general case from the unified theory.

The specific coordinates will be denoted as follows: $r=r_r+r_e$, $r=\begin{bmatrix}x_t & y_t & z_t\end{bmatrix}^T$, $r_e=\begin{bmatrix}x_e & y_e & z_e\end{bmatrix}^T$ and $r_r=\begin{bmatrix}x_r & y_r & z_r\end{bmatrix}^T$. The inertial coordinates in the North-East-Down frame will be labeled as $R_O=\begin{bmatrix}X_O & Y_O & Z_O\end{bmatrix}^T$ with an orientation of $\Theta=\begin{bmatrix}\phi_r & \theta_r & \psi_r\end{bmatrix}^T$. The body has a density labeled ρ_b.

Primary assumptions for the mean axis system are listed below.

A1 – Linear momentum of the elastic body is conserved $\int_V \rho_b(\dot{r}_e)dV = 0$. *This is true of the free response of an ideal elastic body in a vacuum.*

A2 – Angular momentum of the elastic body is conserved as well $\int_V \rho_b(r_r \times \dot{r}_e)dV = 0$. *This is also true of the free response of an ideal elastic body in a vacuum.*

A3 – The origin O is located at the center of mass of the un-deflected structure.

$$\int_V \rho_b(r_r)dV = 0 \quad \int_V \rho_b(\tilde{r}_r)dV = 0 \text{ and } r_e(0) = 0$$

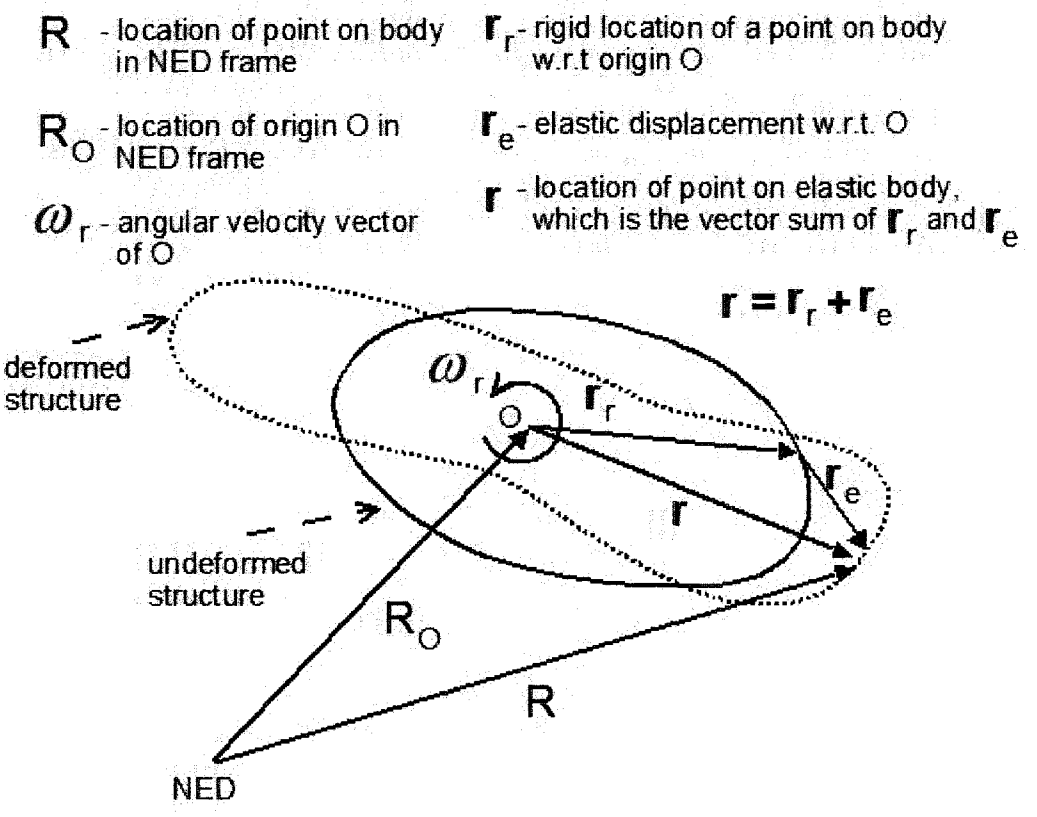


Fig. 7.19. Elastic body above whose origin is O and has an inertial reference frame in the North East Down (NED) coordinate system. We will primarily be interested in the inertial quantities expressed in the frame of O.

The total kinetic energy, T, of the body can be written in the form below.

$$T = \tfrac{1}{2}\int_V \rho_b(\dot{R}^T \dot{R})dV \tag{7.46}$$

The total velocity vector can be represented below

$$\dot{R} = \dot{R}_O + \dot{r} + \omega_r \times r = \dot{R}_O + \dot{r} + \tilde{\omega}_r r = \dot{R}_O + \dot{r} + \tilde{r}^T \omega_r \tag{7.47}$$

The tilde symbol represents a skew-symmetric matrix which is used to represent the cross product of two vectors. Clarification of this convention is listed below in eq. 7.48 with

$\omega_r = [p_r \quad q_r \quad r_r]^T$ and $r = [x_t \quad y_t \quad z_t]^T$

$$\omega_r \times r = \tilde{r}^T \omega_r = \begin{bmatrix} 0 & -z_t & y_t \\ z_t & 0 & -x_t \\ -y_t & x_t & 0 \end{bmatrix}^T \begin{bmatrix} p_r \\ q_r \\ r_r \end{bmatrix} = \tilde{\omega}_r r = \begin{bmatrix} 0 & -r_r & q_r \\ r_r & 0 & -p_r \\ -q_r & p_r & 0 \end{bmatrix} \begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix} \tag{7.48}$$

Expanding eq. 7.47 to include the elastic terms yields eq. 7.49.

$$\begin{aligned} T &= \int_V \rho_b \left(\dot{R}_O^T \dot{R}_O + \dot{r}^T \dot{r} + (\tilde{\omega}_r r)^T (\tilde{\omega}_r r) + 2\dot{R}_O^T \dot{r} + 2\dot{R}_O^T (\tilde{\omega}_r r) + 2\dot{r}^T (\tilde{\omega}_r r) \right) dV \\ T &= T_1 + T_2 + T_3 + T_4 + T_5 + T_6 \\ T_1 &= \int_V \rho_b \left(\dot{R}_O^T \dot{R}_O \right) dV \quad T_2 = \int_V \rho_b \left(\dot{r}^T \dot{r} \right) dV \quad T_3 = \int_V \rho_b (\tilde{\omega}_r r)^T (\tilde{\omega}_r r) dV \\ T_4 &= \int_V \rho_b \left(2\dot{R}_O^T \dot{r} \right) dV \quad T_5 = \int_V \rho_b \left(2\dot{R}_O^T (\tilde{\omega}_r r) \right) dV \quad T_6 = \int_V \rho_b \left(2\dot{r}^T (\tilde{\omega}_r r) \right) dV \end{aligned} \tag{7.49}$$

Fundamental assumptions for the mean axis are that the terms T_4, T_5 and T_6 are 0. Thus, we will look at these two terms in close detail. T_4 can be rewritten as follows in eq. 7.50 because $\dot{R}_O^T$ is constant along the body.

$$\frac{T_4}{2} = 2\dot{R}_O^T \int_V \rho_b (\dot{r}) dV = 2\dot{R}_O^T \int_V \rho_b (\dot{r}_r + \dot{r}_e) dV \tag{7.50}$$

Now, $\dot{r}_r = 0$ by definition. For an unrestrained elastic body *in a vacuum,* $\int_V \rho_b (\dot{r}_e) dV = 0$, by conservation of momentum. So, for an unrestrained elastic body *in a vacuum,* $T_4 = 0$.

We now re-write T_6 as follows in eq. 7.51 and notice that ω_r is constant along the body, so the same algebraic step can be taken again.

$$\begin{aligned} \frac{T_6}{2} &= \int_V \rho_b \left(\dot{r}^T (\tilde{\omega}_r r) \right) dV = \int_V \rho_b \left(\dot{r}^T \left(\tilde{r}^T \omega_r \right) \right) dV = \int_V \rho_b \left(\dot{r}^T \left(\tilde{r}^T \right) \right) dV \omega_r = \int_V \rho_b (r \times \dot{r}) dV \omega_r \\ \frac{T_6}{2} &= \int_V \rho_b (r_e \times \dot{r}_e + r_r \times \dot{r}_e + r_r \times \dot{r}_r + r_e \times \dot{r}_r) dV \omega_r = \int_V \rho_b (r_e \times \dot{r}_e + r_r \times \dot{r}_e) dV \omega_r \end{aligned} \tag{7.51}$$

We expect that $\int_V \rho_b (r_e \times \dot{r}_e) dV$ should be 0 because the elastic displacement and elastic displacement rate should be in the same direction.

The justification for making $\int_V \rho_b (r_r \times \dot{r}_e) dV = 0$ is that an unrestrained elastic body *in a vacuum* should exhibit conservation of angular momentum. Thus, one would expect $T_6 = 0$ for an elastic body *in a vacuum.*

Now, to make $T_5 = 0$, we will have to assume that the origin O, is located at the center of mass of the un-deflected body. Previously, we assumed that for an unrestrained elastic body *in a vacuum,* $\int_V \rho_b (\dot{r}_e) dV = 0$. Now, if we assume the initial condition of zero deflection, $r_e(0) = 0$, then $\int_V \rho_b (r_e) dV = 0 \quad \forall \quad t$. Thus, we can also say that $\int_V \rho_b (\tilde{r}_e) dV = 0 \quad \forall \quad t$.

$$\frac{T_5}{2} = \int_V \rho_b \dot{R}_O^T (\tilde{\omega}_r) dV = \int_V \rho_b \dot{R}_O^T (\tilde{r}^T \omega_r) dV = \dot{R}_O^T \left(\int_V \rho_b (\tilde{r}^T) dV \right) \omega_r$$
$$\frac{T_5}{2} = \dot{R}_O^T \left(\int_V \rho_b (\tilde{r}_e^T + \tilde{r}_r^T) dV \right) \omega_r = 0 \tag{7.52}$$

One really needs to look at the modeshapes to figure out whether or not the mean axis assumptions can be used. We re-iterate the assumptions A1 and A2 using modal notation, noting that $r_e = \sum_{k=1}^{\infty} \Phi_k \eta_k$ where each Φ_k would be a 3 x 1 vector and each η_k would be a scalar.

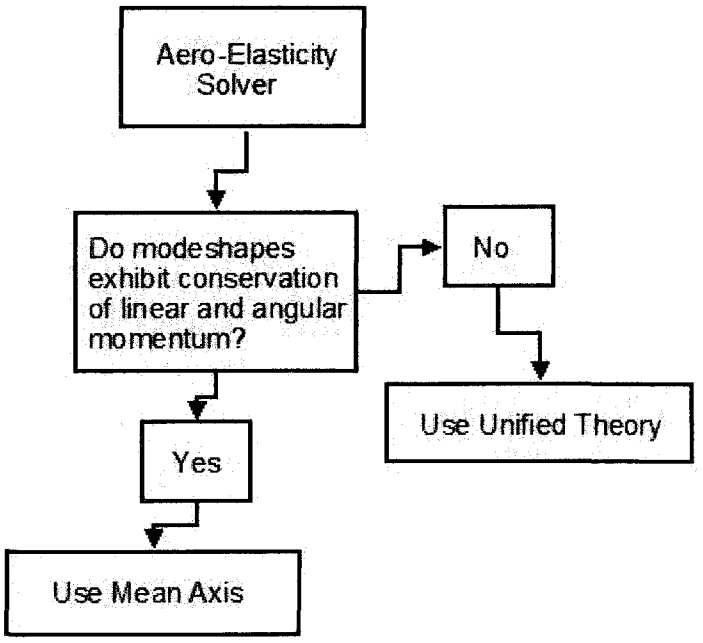


Fig. 7.20. If the modeshapes of the aircraft look as though momentum will be conserved, one should use the mean axis equations of motion. Otherwise, one may consider using the more complicated unified theory.

A1 – Linear momentum of the elastic body is conserved $\sum_{k=1}^{\infty}\left(\int_V \rho_b(\Phi_k)dV\right)\eta_k = 0$. Since each η_k can be completely arbitrary, this means that the integral $\int_V \rho_b \Phi_k dV = 0$ for each mode k. This is true of the free response of an ideal elastic body *in a vacuum.*

A2 – Angular momentum of the elastic body is conserved as well $\sum_{k=1}^{\infty}\int_V \rho_b(r_r \times \Phi_k)dV\eta_k = 0$.

Once again, due to the arbitrariness of η_k, $\int_V \rho_b(r_r \times \Phi_k)dV\eta_k = 0$. This is also true of the free response of an ideal elastic body *in a vacuum.*

Aircraft do not fly in a vacuum. Therefore, we cannot expect that assumptions A1 and A2 will be strictly true. However, at low dynamic pressures for fairly stiff aircraft, the influence of aerodynamics on the structure is not very strong. So, the following advisements can be made about when to apply the mean axis assumptions in Fig. 7.20.

The equations of motion under the mean axis assumption from Ref. [3.25] are listed in eq. 7.53.

$$
\begin{aligned}
m_r(\dot{u}_r - r_r v_r + q_r w_r + g\,\boldsymbol{sin}\,\theta_r) &= \sum X_{aero} \\
m_r(\dot{v}_r - p_r w_r + r_r u_r - g\,\boldsymbol{sin}\,\phi_r\,\boldsymbol{cos}\,\theta_r) &= \sum Y_{aero} \\
m_r(\dot{w}_r - q_r u_r + p_r v_r - g\,\boldsymbol{cos}\,\phi_r\,\boldsymbol{cos}\,\theta_r) &= \sum Z_{aero}
\end{aligned}
$$

$$
\begin{aligned}
&I^r_{xx}\dot{p}_r - \left(I^r_{xy}\dot{q}_r + I^r_{xz}\dot{r}_r\right) + \left(I^r_{zz} - I^r_{yy}\right)q_r r_r + \left(I^r_{xy}r_r - I^r_{xz}q_r\right)p_r + \left(r_r^2 - q_r^2\right)I^r_{yz} = \sum L_{aero} \\
&I^r_{yy}\dot{q}_r - \left(I^r_{xy}\dot{p}_r + I^r_{yz}\dot{r}_r\right) + \left(I^r_{xx} - I^r_{zz}\right)p_r r_r + \left(I^r_{xy}p_r - I^r_{yx}r_r\right)q_r + \left(p_r^2 - r_r^2\right)I^r_{xz} = \sum M_{aero} \\
&I^r_{zz}\dot{r}_r - \left(I^r_{xz}\dot{p}_r + I^r_{yz}\dot{q}_r\right) + \left(I^r_{yy} - I^r_{xx}\right)p_r q_r + \left(I^r_{xz}q_r - I^r_{yz}p_r\right)r_r + \left(q_r^2 - p_r^2\right)I^r_{xy} = \sum N_{aero}
\end{aligned}
\qquad (7.53)
$$

$$
\ddot{\eta}_i + 2\zeta\omega_{ni}\dot{\eta}_i + \omega_{ni}^2\eta_i = \sum Q_{aero}
$$

Here, u_r, v_r, w_r represent the rigid, body fixed velocities *at the center of mass*, in the x, y, and z axes respectively. Also, p_r, q_r, r_r represent the rigid, body fixed angular rates *at the center of mass*, in the x, y, and z axes respectively. Aerodynamic forces are represented as $\sum X_{aero}, \sum Y_{aero}, \sum Z_{aero}$ which are summed in the x, y, and z axes respectively. Aerodynamic moments are represented as $\sum L_{aero}, \sum M_{aero}, \sum N_{aero}$ which are summed about the x, y, and z axes respectively. The η_i terms represent modal coordinates of the flexible subsystem and the $\sum Q_{aero}$ term would represent aerodynamic forces and moments expressed in terms of modal coordinates.

A block diagram of how to model under the mean axis assumption with an aero-elastic model is shown in Fig. 7.21.

Structural mass and stiffness matrices alone are always symmetric. Therefore, the eigenvectors (i.e. natural modes) are always orthogonal. This includes the rigid body modes. In a vacuum, the rigid body modes will be independent from the elastic modes and each elastic mode will be independent of every other elastic mode.

Notice that the dynamic pressure, $\overline{q}$, acts as a constant gain in the feedback diagram shown in Fig. 7.21. At low dynamic pressure the aerodynamic influence coefficients are not that important and the mean axis assumption becomes very meaningful. However, at high dynamic pressures, the aerodynamic influence coefficients, AIC_s, exert a stronger influence on the overall stiffness of the aero-elastic system. There is no guarantee that the AIC_s will be symmetric. Thus, one can expect that the orthogonality of the rigid and elastic modes would degrade.

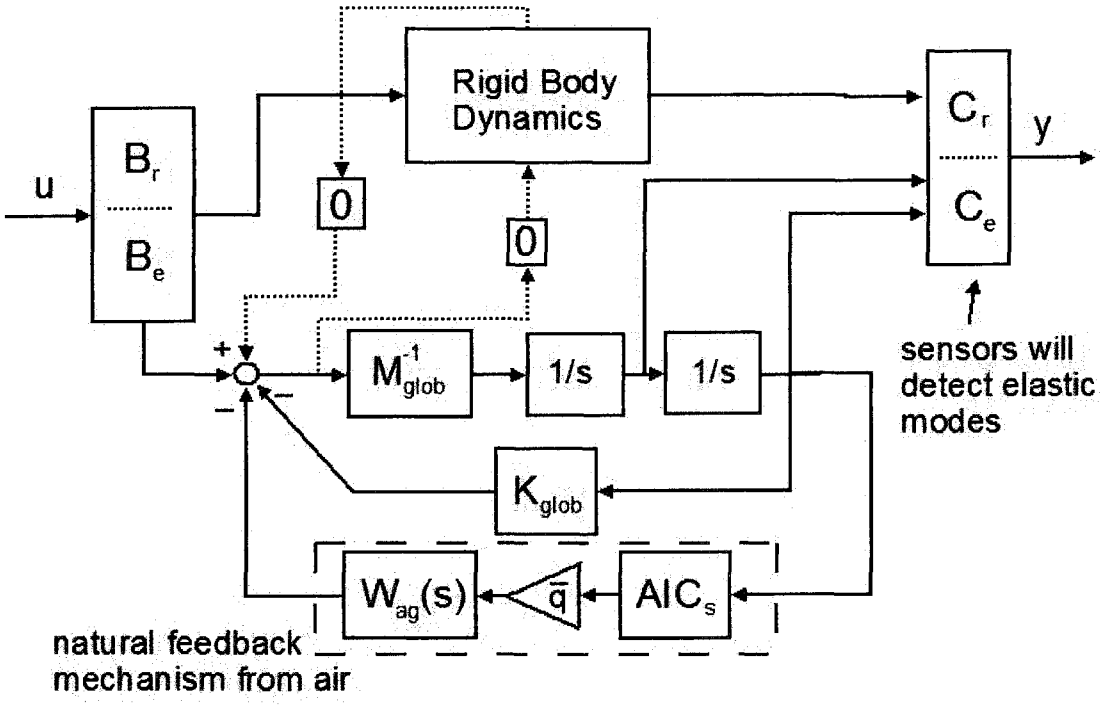


Fig. 7.21. The modeling strategy to integrate a rigid body model with a finite element aero-elastic model. It is assumed that there is a global mass matrix, M_{glob}, a global stiffness matrix, K_{glob}, a steady aerodynamic influence coefficient matrix, AIC_s, dynamic pressure, $\bar{q}$. The dynamic pressure, $\bar{q}$, is just a scalar. The author decided to use Wagner's lift growth function, $W_{ag}(s)$, to approximate unsteady aerodynamic lift generation. The B matrix is partitioned into an elastic component, B_e, and a rigid component, B_r. The C matrix is partitioned into an elastic component, C_e, and a rigid component, C_r.

7.8 Creating a Simplified Structural Aircraft Model with Beam Elements

Creating an accurate aero-elastic model of an aircraft is a significant challenge and would probably result in a higher level of detail than is really necessary for feedback control. The model presented in the following sections is not intended for a precise flutter analysis or design. This model is made purely to experiment with sensor placement and feedback techniques.

A significant challenge with creating any finite element model is keeping track of element connectivity, geometry and orientation. This section will detail some simple arrays that can be used to help keep things organized [3.20].

The first step is to draw a picture of the structural model with nodes along the fuselage and the wings. The next step is to number those nodes in a pattern that hopefully makes sense. A beam element

has two nodes associated with it. Next, one would number beam elements in a repeating pattern, placing those element numbers in between the corresponding nodes. After this step is done, one should arrive at something that looks like Fig. 7.22.

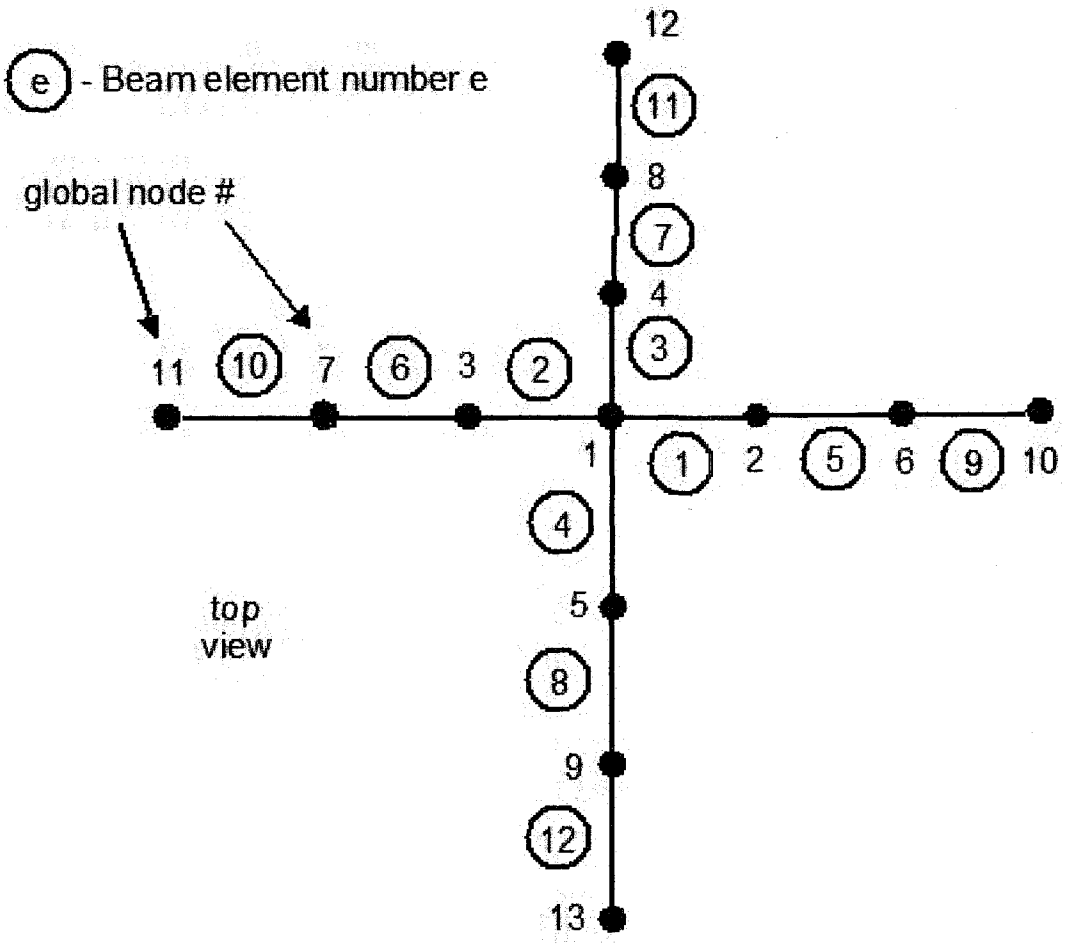


Fig. 7.22. The modeling strategy with beam elements

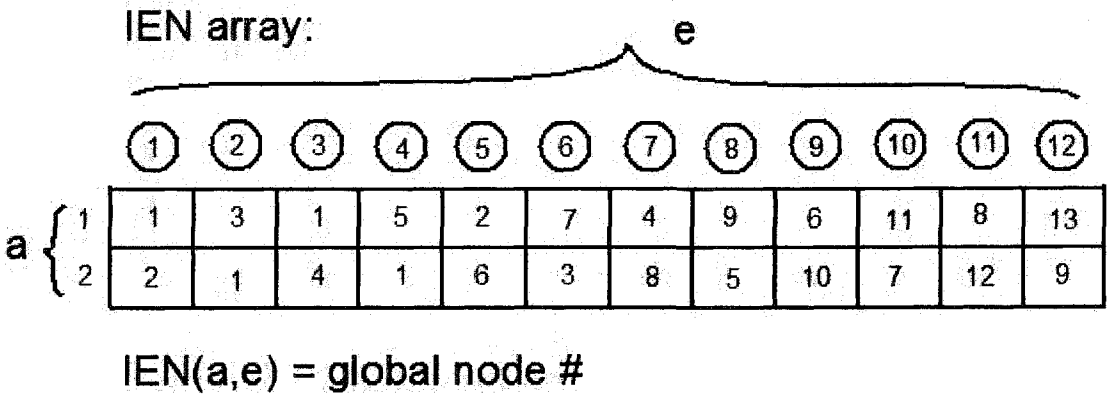


Fig. 7.23. The IEN array maps element numbers and local node numbers to global node numbers

Now to help with the programming, an array that associates node numbers with element numbers is

made. This is called the IEN array. For beam elements the IEN array should be of dimension 2 x n_ele. The entry IEN(1,9) would contain the 1st local node of the 4th element in global coordinates, from Fig. 7.22, IEN(1,9) = 6. The IEN array is populated in Fig. 7.23

ID array:

g global node #

dof # \ v	1	2	3	4	5	6	7	8	9	10	11	12	13
1	1	4	7	10	13	16	19	22	25	28	31	34	37
2	2	5	8	11	14	17	20	23	26	29	32	35	38
3	3	6	9	12	15	18	21	24	27	30	33	36	39

ID(v,g) = stiffness entry #

Fig. 7.24. The ID array maps global node numbers to matrix entries.

After the IEN array is populated, one goes about creating an ID array which maps global node numbers and degrees of freedom to a particular number which will be used for placement into the global stiffness and global mass matrices. Since this is an aircraft, the free-free boundary condition will be employed and one can simply count upwards until everything is accounted for. The parameter, v, will refer to a degree of freedom (dof). A three dof beam element would have a bending displacement, bending slope and twist angle from torsion at each node. The ID array for the model of Fig. 7.23 is shown in Fig. 7.24.

One can also create an LM array which would be an array whose entries would be populated in triple as LM(v,a,e) = ID(v,IEN(a,e)). The LM array is just a combination of the ID and IEN arrays and is only modestly helpful in programming though it may help with readability. MATLAB code to create these arrays is shown on the next page.

Elemental beam stiffness with 3-dof on the fuselage is listed below in eq. 7.54.

$L_e = element \quad length \quad EI_e = bending \quad stiffness \quad GJ_e = torsional \quad stiffness$

$$K^e = \begin{bmatrix} \frac{12EI_e}{L_e^3} & \frac{6EI_e}{L_e^2} & 0 & \frac{-12EI_e}{L_e^3} & \frac{6EI_e}{L_e^2} & 0 \\ \frac{6EI_e}{L_e^2} & \frac{4EI_e}{L_e} & 0 & \frac{-6EI_e}{L_e^2} & \frac{2EI_e}{L_e^2} & 0 \\ 0 & 0 & \frac{GJ_e}{L_e} & 0 & 0 & \frac{-GJ_e}{L_e} \\ \frac{-12EI_e}{L_e^3} & \frac{-6EI_e}{L_e^2} & 0 & \frac{12EI_e}{L_e^3} & \frac{-6EI_e}{L_e^2} & 0 \\ \frac{6EI_e}{L_e^2} & \frac{2EI_e}{L_e^2} & 0 & \frac{-6EI_e}{L_e^2} & \frac{4EI_e}{L_e} & 0 \\ 0 & 0 & \frac{-GJ_e}{L_e} & 0 & 0 & \frac{GJ_e}{L_e} \end{bmatrix} \quad x^e = \begin{bmatrix} z_{e1} \\ \theta_{e1} \\ \phi_{e1} \\ z_{e2} \\ \theta_{e2} \\ \phi_{e2} \end{bmatrix} \tag{7.54}$$

MATLAB code to form IEN, ID and LM arrays

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Structural Portion,
% Use beam elements to make up global stiffness, K_global
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
n_ele=4*3; %number of elements
node_per_ele=2;
n_node=n_ele+1; %number of nodes
n_dof=3; %number of degrees of freedom per node, displacement, slope
         %and twist

%form IEN array
IEN=[1 3 1 5;2 1 4 1];

for e=5:n_ele
    if ( mod(e,2) > 0 )
        IEN(1,e)=e-3;
        IEN(2,e)=e+1;
    else
        IEN(1,e)=e+1;
        IEN(2,e)=e-3;
    end
end

%form ID array
count=0;
for g_node=1:n_node
    for v=1:n_dof
        count=count+1;
        ID(v,g_node)=count;
    end
end

%form LM array
for e=1:n_ele
    for a=1:node_per_ele
        for v=1:n_dof
            LM(v,a,e)=ID(v,IEN(a,e));
        end
    end
end
```

In eq. 7.54, z_{e1} would be downward displacement at the first local node, θ_{e1} would be the bending slope at the first local node and ϕ_{e1} would be a twist angle at the first local node.

Now, with the elemental stiffness one has to add up the elemental portions of the stiffness and mass matrices into global stiffness and mass matrices. Assembling a global stiffness matrix is quite simple, provided that one uses the IEN and ID arrays. Elemental stiffness entries sum together at their global node locations. The only difficulty is mapping local nodes to global nodes, which is made much

easier with usage of the IEN and ID arrays. MATLAB code to assemble the global stiffness matrix this is shown on the next page.

MATLAB code to form global stiffness matrix

```
%elemental stiffness parameters
EI=4e9; GJ=4e9; L=50/(n_ele/2);
Ke_fus=[12*EI/L^3 6*EI/L^2 0 -12*EI/L^3 6*EI/L^2 0;
    6*EI/L^2 4*EI/L 0 -6*EI/L^2 2*EI/L 0;
    0 0 GJ/L 0 0 -GJ/L;
    -12*EI/L^3 -6*EI/L^2 0 12*EI/L^3 -6*EI/L^2 0;
    6*EI/L^2 2*EI/L 0 -6*EI/L^2 4*EI/L 0;
    0 0 -GJ/L 0 0 GJ/L];

EI=1e9; GJ=1e9; L=40/(n_ele/2);
Ke_wing=[12*EI/L^3 0 6*EI/L^2 -12*EI/L^3 0 6*EI/L^2;
    0 GJ/L 0 0 -GJ/L 0;
    6*EI/L^2 0 4*EI/L -6*EI/L^2 0 2*EI/L;
    -12*EI/L^3 0 -6*EI/L^2 12*EI/L^3 0 -6*EI/L^2;
    0 -GJ/L 0 0 GJ/L 0;
    6*EI/L^2 0 2*EI/L -6*EI/L^2 0 4*EI/L];

%transform elements into global coords for straight wing
%would be more complicated for wing sweep or dihedral
for e=1:n_ele
    if ( (mod(e,4) < 3) & (mod(e,4) > 0) )
        Ke_global(:,:,e)=Ke_wing;
    else
        Ke_global(:,:,e)=Ke_fus;
    end
end

%form global stiffness
K_global=zeros(count-1);
for e=1:n_ele
    for a=1:node_per_ele
    for b=1:node_per_ele
        for i=1:n_dof
        for j=1:n_dof
            dummya=a-1;
            dummyb=b-1;
            P1=LM(i,a,e);
            P2=LM(j,b,e);
            row=dummya*n_dof+i;
            col=dummyb*n_dof+j;
            K_global(P1,P2)=K_global(P1,P2)+Ke_global(row,col,e);
        end
        end
     end
     end
end
```

The elemental mass matrix for an element on the fuselage is listed below in eq. 7.55. The density would be a unit mass per unit length measurement. One should note that it is reasonable to use a diagonal lumped mass matrix or to simply add mass at nodal locations where an engine might be. For example, if an 500 kg engine were attached at node 6, then a point mass of 500 kg could be simply added to the global mass in the (16,16), (17,17) and (18,18) entries of the global mass matrix.

$$L_e = element\ \ length\ \ \rho_e = \ mass\ \ per\ \ unit\ \ length$$

$$M^e = \frac{\rho_e L_e}{420}\begin{bmatrix} 156 & 22L_e & 0 & 54 & -13L_e & 0 \\ 22L_e & 4L_e^2 & 0 & 13L_e & -3L_e^2 & 0 \\ 0 & 0 & \frac{420}{3} & 0 & 0 & \frac{420}{6} \\ 54 & 13L_e & 0 & 156 & -22L_e & 0 \\ -13L_e & -3L_e^2 & 0 & -22L_e & 4L_e^2 & 0 \\ 0 & 0 & \frac{420}{6} & 0 & 0 & \frac{420}{3} \end{bmatrix} \quad x^e = \begin{bmatrix} z_{e1} \\ \theta_{e1} \\ \phi_{e1} \\ z_{e2} \\ \theta_{e2} \\ \phi_{e2} \end{bmatrix} \tag{7.55}$$

7.9 Steady Aerodynamic Influence Coefficients (AIC)

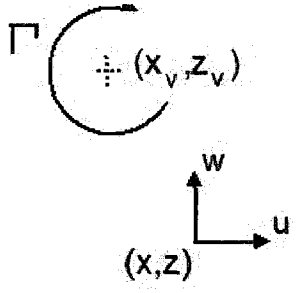


Fig. 7.25. Circulation Γ, creates lift

Obtaining accurate AIC is a difficult task for complex airfoil geometries. This section makes the simplifying assumptions of an incompressible and inviscid airflow. This allows the usage of vortex lattice methods which are reasonable for making lift and moment calculations [3.29].

Consider the vortex with circulation, Γ, at the point of (x_v, z_v) as in Fig. 7.25. It is shown in Ref. [3.29, 3.28] that the flow can be written as in eq. 7.56.

$$u = \frac{\Gamma}{2\pi}\frac{(z - z_v)}{r^2} \quad w = -\frac{\Gamma}{2\pi}\frac{(x - x_v)}{r^2} \qquad r^2 = (x - x_v)^2 + (z - z_v)^2 \tag{7.56}$$

In this section, the variables u and w are flow velocities and not necessarily associated with the velocity of the aircraft.

We will be considering splitting an aerofoil into two chordwise elements as shown in Fig. 7.26. According to the vortex lattice theory, there will be zero normal flow boundary condition at the three-quarter chord point. This three quarter chord point is called a control point for vortex lattice. The zero normal flow boundary condition is reflected in eq. 7.57. Here, $cbr1$ and $cbr2$ represent camber angles associated with the airfoil, α is the angle of attack. The velocities $w1$ and $w2$ represent vertical velocities at control points 1 and 2 in Fig. 7.26. The velocities $u1$ and $u2$ represent horizontal velocities at control points 1 and 2 in Fig. 7.26.

$$\begin{aligned} n1 = 0 &= (w1 + V_\infty \sin\alpha)\cos(cbr1) + (u1 + V_\infty \cos\alpha)\sin(cbr1) \\ n2 = 0 &= (w2 + V_\infty \sin\alpha)\cos(cbr2) + (u2 + V_\infty \cos\alpha)\sin(cbr2) \end{aligned} \tag{7.57}$$

We insert the vortex relationship of eq. 7.56 into eq. 7.57 which results in eq. 7.58. Now, we can relate airfoil geometry to circulation. With the circulation known, one can calculate the lift and moment forces at the elastic axis.

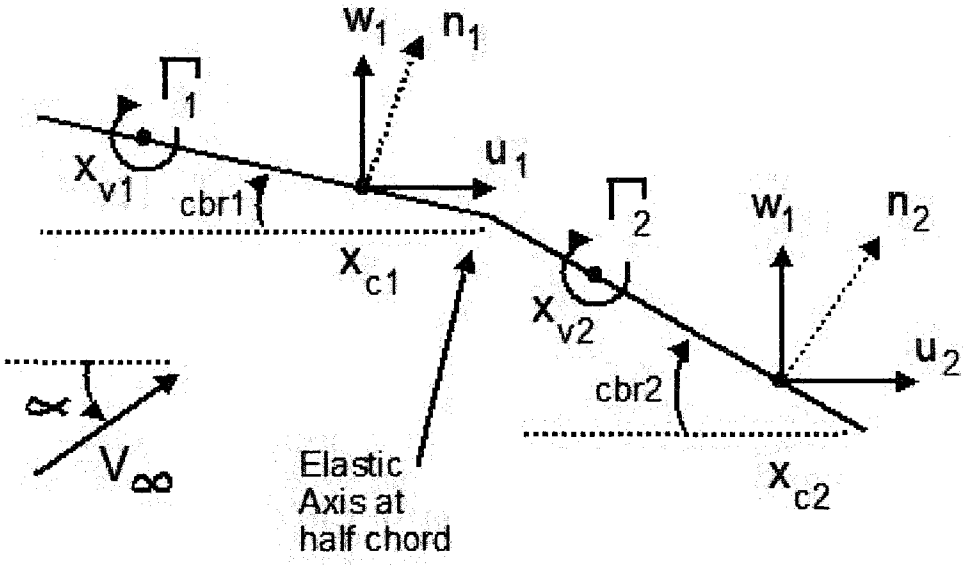


Fig. 7.26. The vortex lattice method consists of aerodynamic panels which are split up along the chord.

$$\frac{1}{2\pi}\begin{bmatrix} \frac{-(x_{c1}-x_{v1})}{r_{11}^2}\cos(cbr1)+\frac{(z_{c1}-z_{v1})}{r_{11}^2}\sin(cbr1) & \cdots \\ \frac{-(x_{c2}-x_{v1})}{r_{21}^2}\cos(cbr2)+\frac{(z_{c2}-z_{v1})}{r_{21}^2}\sin(cbr2) & \\ \frac{-(x_{c1}-x_{v2})}{r_{12}^2}\cos(cbr1)+\frac{(z_{c1}-z_{v2})}{r_{12}^2}\sin(cbr1) & \\ \frac{-(x_{c2}-x_{v2})}{r_{22}^2}\cos(cbr2)+\frac{(z_{c2}-z_{v2})}{r_{22}^2}\sin(cbr2) & \end{bmatrix}\begin{bmatrix}\Gamma_1\\ \Gamma_2\end{bmatrix} = \Theta\begin{bmatrix}\Gamma_1\\ \Gamma_2\end{bmatrix} = -V_\infty\begin{bmatrix}\sin\alpha\cos(cbr1)+\cos\alpha\sin(cbr1)\\ \sin\alpha\cos(cbr2)+\cos\alpha\sin(cbr2)\end{bmatrix} \tag{7.58}$$

To simplify eq. 7.58, we assume that the camber angles are small, vortices are placed at quarter chord points, the control points are placed at three quarter chord points and that the airfoil is thin. What results is shown in eq. 7.59, where c is the total chord length.

$$\frac{2}{c\pi}\begin{bmatrix}-1 & 1\\ -1/3 & -1\end{bmatrix}\begin{bmatrix}\Gamma_1\\ \Gamma_2\end{bmatrix} = -V_\infty\begin{bmatrix}\alpha\\ \alpha\end{bmatrix} \tag{7.59}$$

Now, given eq. 7.59, we can solve for the lift since $L_{ift} = \rho_{air}V_\infty\Gamma$.

$$\begin{bmatrix}L_{ift1}\\ L_{ift2}\end{bmatrix} = \rho_{air}V_\infty\begin{bmatrix}\Gamma_1\\ \Gamma_2\end{bmatrix} = -\rho_{air}V_\infty^2\frac{c\pi}{2}\begin{bmatrix}-1 & 1\\ -1/3 & -1\end{bmatrix}^{-1}\begin{bmatrix}\alpha\\ \alpha\end{bmatrix} = \rho_{air}V_\infty^2\frac{c\pi}{8}\begin{bmatrix}3 & 3\\ -1 & 3\end{bmatrix}\begin{bmatrix}\alpha\\ \alpha\end{bmatrix} \tag{7.60}$$

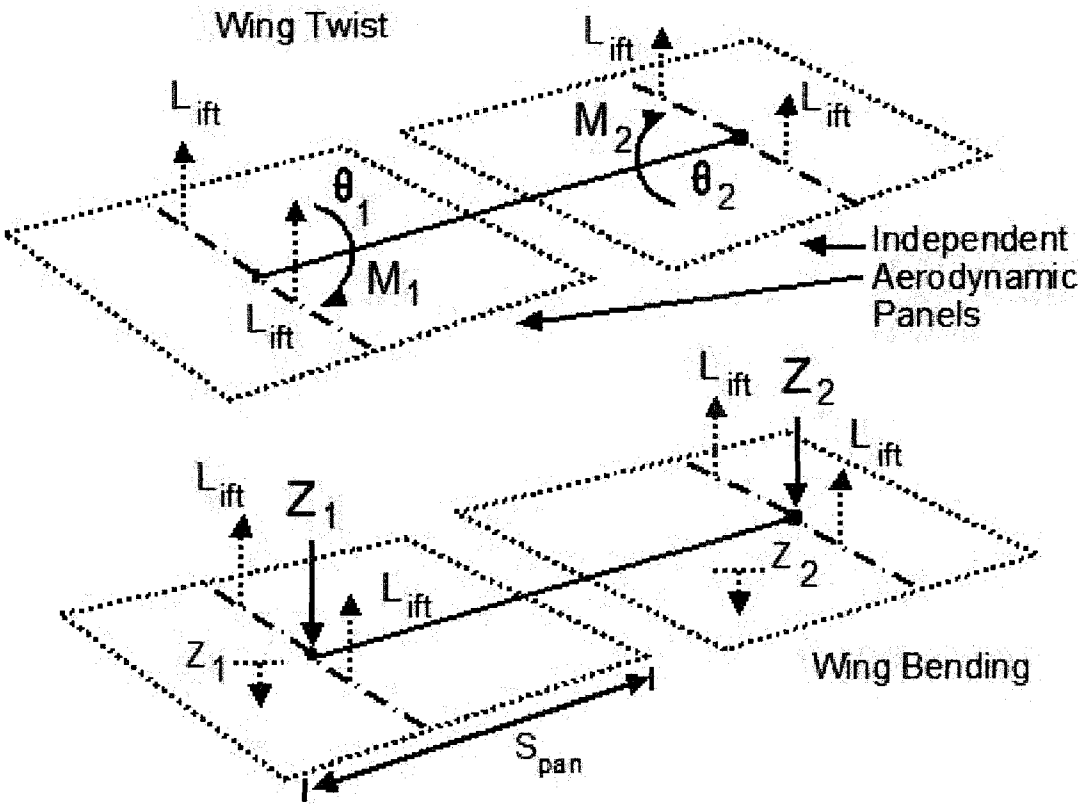


Fig. 7.27. This modeling strategy will consider independent aerodynamic panels, not considering 3D aerodynamic effects.

With the lift calculated from eq. 7.60, we can now translate these forces into an aerodynamic stiffness term assuming $\alpha \approx \theta_{wing\ \ twist}$. Figure 7.27 shows graphically the aerodynamic panels and the coordinate system for the elastic system. Some transformation is needed because the elastic system was created with the z-axis downward. The following transformation matrix, T_1, is used to translate forces from the upward lift forces to the downward z-axis forces and to create twisting moments from lift. The matrix T_2 relates elastic deflections to the angle of attack.

$$\begin{bmatrix} Z \\ M \\ L \end{bmatrix} = \frac{\rho_{air} V_\infty^2}{2} \begin{bmatrix} -1 & -1 \\ 3c/8 & -c/8 \\ 0 & 0 \end{bmatrix} s_{pan} \begin{bmatrix} 3c/4\pi & 3c/4\pi \\ -c/4\pi & 3c/4\pi \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix} = \bar{q} T_2 s_{pan} \Theta^{-1} T_1 \tag{7.51}$$

Since the steady AIC is treated as a stiffness term multiplied by dynamic pressure, $\bar{q}$, then the sign must be reversed and we have the following formula in eq. 7.52.

$$AIC_s = -T_2 \Theta^{-1} T_1 \tag{7.52}$$

7.10 Wagner's Lift Growth Function

Wagner's lift growth function originated from experiments dating back to 1925, where a wing was rapidly accelerated to constant velocity and the lift was measured as a function of time [3.25]. The idea behind Wagner's lift growth function is shown to the left in Fig. 7.28.

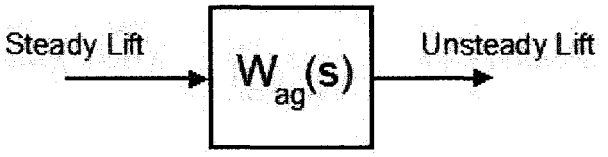


Fig. 7.28. The modeling strategy to integrate a rigid body model with a finite element aero-elastic model. It is assumed that there is a global

To approximate unsteady lift, one would do a steady lift calculation and then make that steady lift an input to wagner's lift growth transfer function, which would give a reasonable approximation to unsteady lift.

The linear approximation to a build up of lift amounts to a mild lag whose state space is listed below in eq. 7.53.

$$
\begin{aligned}
&W_{ag}(s) = C_{wag}\left(sI - A_{wag}\right)^{-1} B_{wag} + D_{wag} \\
&r_o = 0.0455\left(V_\infty / 2c\right) \quad r_1 = 0.3\left(V_\infty / 2c\right) \\
&A_{wag} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -r_o r_1 & -(r_o + r_1) \end{bmatrix} \quad B_{wag} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \\
&C_{wag} = \begin{bmatrix} 0 & 0.5 r_o r_1 & 0.165 r_o + 0.335 r_1 \end{bmatrix} \quad D_{wag} = 0.5
\end{aligned}
\tag{7.53}
$$

The bode plot of Wagner's lift growth function is shown in Fig. 7.29. As can be seen from the bode plot, the steady state gain is 1 unsurprisingly and the high frequency gain is 0.5, which corresponds to the D_{wag} value. This means that the lift will instantly reach ½ of its steady value and slowly increase until it reaches its steady value. Do notice that there is a 15 degree phase lag which can manifest itself in a reasonable frequency range even for rigid body dynamics at low speeds. At high speeds, the 15 degree phase lag is less noticeable.

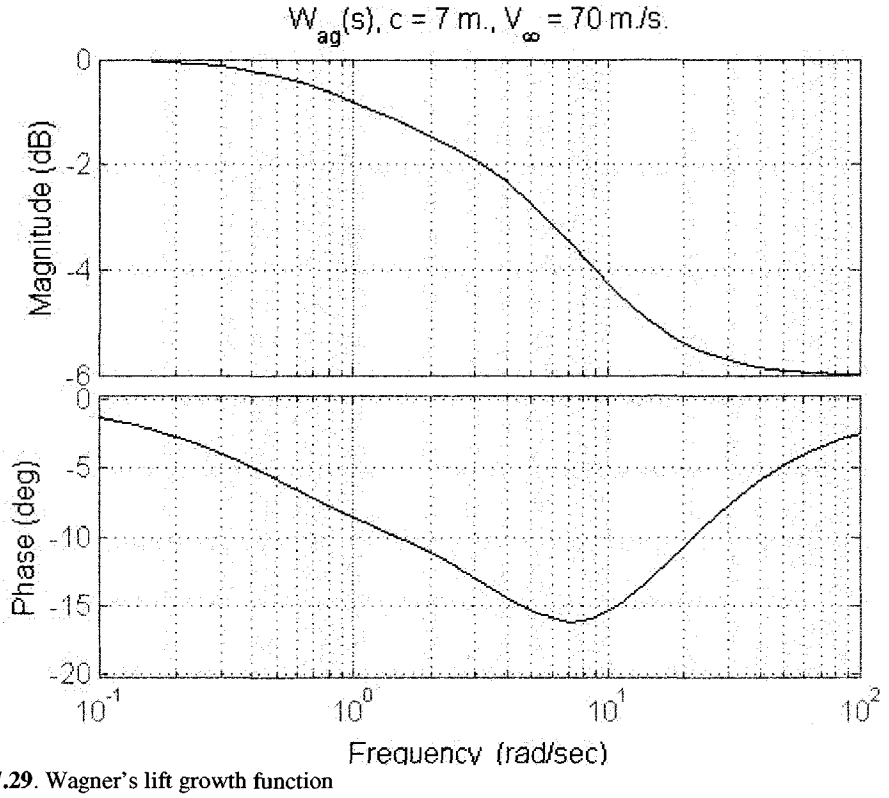


Fig. 7.29. Wagner's lift growth function

7.11 Digital Issues With Slow Flexible Dynamics

The 0.1 second digital sampling rate is very popular with aircraft control.

> "For systems with human input commands where the system response is critical (such as a fly-by-wire flight control), the time delay alone suggests that the sample period be kept to a small fraction of the rise time. A pilot flying an airplane with digital fly-by-wire flight control will complain if the sampling delay is on the order of a tenth of a second from input action to the beginning of the response. Assuming we wish to keep the time delay to be 10% of the rise time, a 10-Hz sample frequency should be used for 1 sec. rise time or, in terms of the non-dimensional sampling multiple, $\omega_s / \omega_{BW} \geq 20$." –Ref. [7.7]

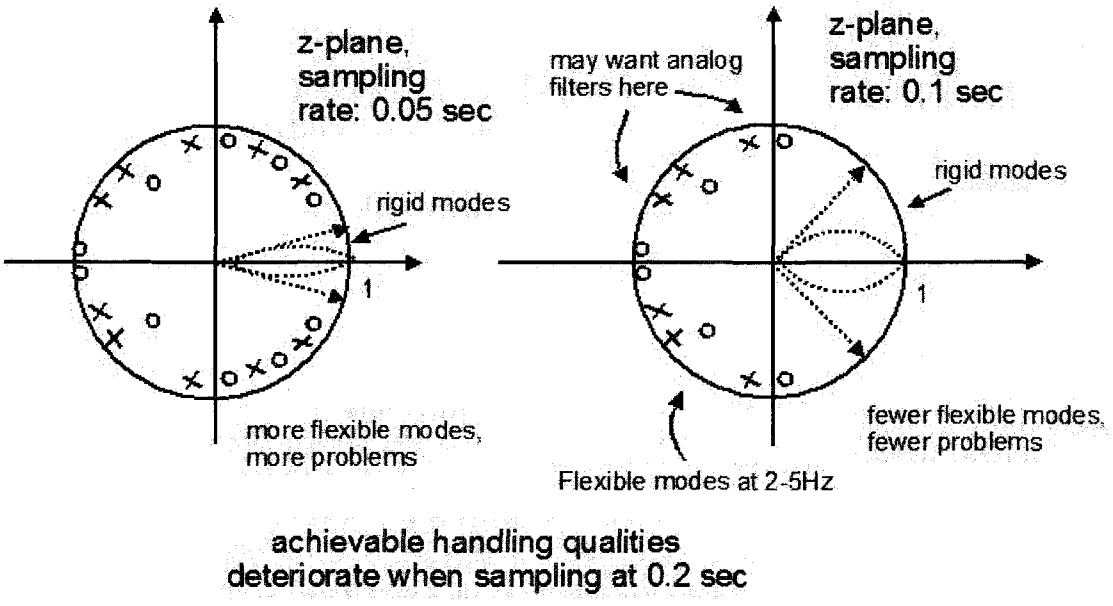


Fig. 7.30. Choosing a sampling period slower than 0.1 seconds may be advantageous because it allows one to deal with fewer flexible modes. However, it is simply not option. Also, one may consider using a faster sampling rate because one can use simpler Z transform methods to emulate controllers without significant distortion in phase or magnitude. However, there are unfortunately more flexible modes to deal with in this case. What a nuisance!

The same conclusion was reached in Ref. [2.20].

As long as the slowest flexible dynamics are above 10 Hz, or reasonably close, then filtering out flexible modes becomes a fairly painless task. The digital sampling rate of 0.1 seconds naturally allows some filtering of those flexible modes. Additionally, one can use an anti-aliasing filter, which could just be a first order analog RC-circuit.

For aircraft with slower flexible modes around 1-5 Hz, notch filtering of those flexible modes can become necessary. Digitizing a continous time notch filter design by a zero order hold Z transform might not be such a good idea in this case unless a quicker sampling rate of 0.05 seconds or faster is used. One may want to consider digitizing the continuous time notch filter design by different Z transform methods, such as the Bilinear/Tustin Z transform, matched pole-zero Z transforms, or use second order modules [7.7,7.8]. One can use the MATLAB "notchdemo" to quickly assess tradeoffs between sampling

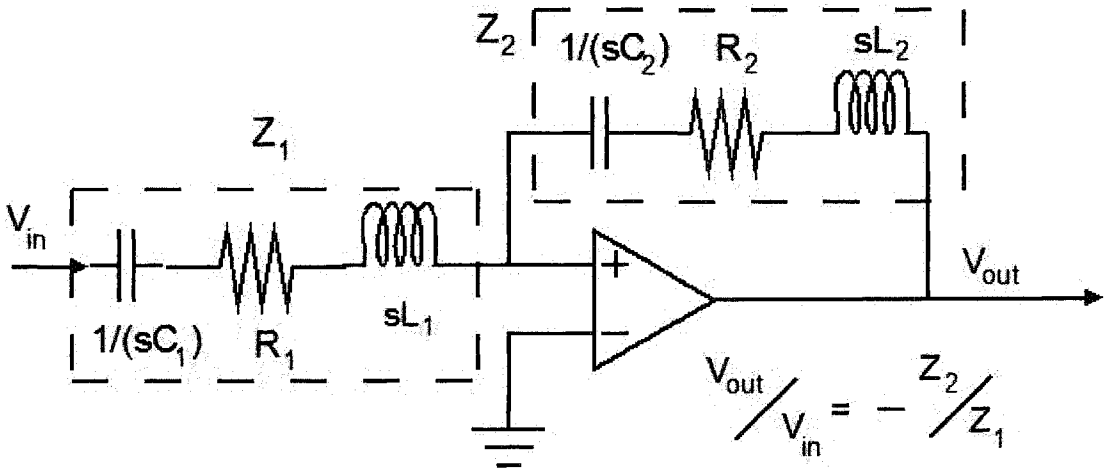


Fig. 7.31. Using an inverting op amp, one can implement an analog notch filter without resorting to sophisticated Z transforms. Note that there would be a negative sign in front of the filter because this is an inverting op amp. One may want to use a second inverting op amp in series to correct the sign change.

rate and Z transform methods. Alternatively, one may also prefer to do a direct digital design. The MATLAB "sisotool()" function allows for an easy and graphical means of doing direct digital design.

One may find these sophisticated Z transforms rather intimidating, and one may wish to use purely analog notch filters which were originally designed in the continuous time domain. This way, one does not need to worry about the distortion due to digitization.

Whatever the method, there is certainly an extra design effort involved that can become a nuisance when dealing with slower flexible dynamics in the 1-5 Hz range.

7.12 Sensor Placement for P17

The author was hesitant to use 10 or 100 accelerometers across the entire body of the aircraft because the aircraft will be subject to damage. It was perceived that fewer sensors meant fewer susceptibilities to damaged sensors.

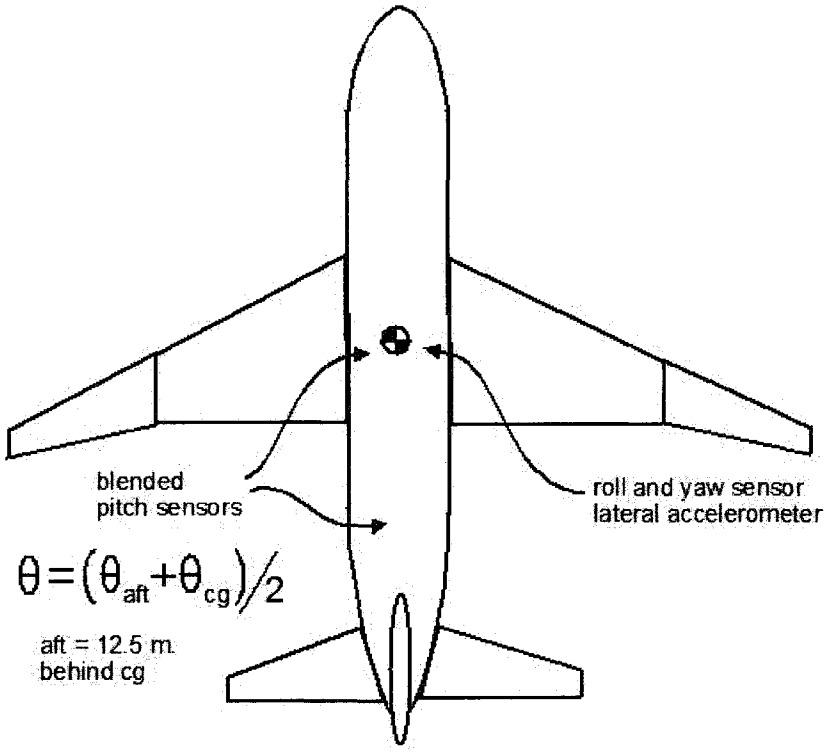


Fig. 7.32. Lateral sensors were simply placed at the c.g. Longitudinal sensors for pitch were placed at the c.g. and 12.5 meters aft and averaged.

The author then considered blending vertical and lateral accelerometers at the c.g. and at the tail together much like the squaring down procedure in section 2.31. Unfortunately, the author was not able to obtain a significant benefit from blending accelerometers at the c.g. and rear of the aircraft.

The author was capable of maintaining healthy gain margins with lateral sensors placed only at the c.g.

Sensor blending for pitch rate gyros was needed. The author was not capable of meeting acceptable gain margins and meeting reasonable bandwidths without resorting to multiple sensor placement techniques for the pitch angle output. One pitch angle sensor was placed at the c.g. and another pitch angle sensor was placed 12.5 meters aft of the c.g. The pitch sensors were simply averaged to use as an output for pitch angle.

One should note that the fuselage of the P17 is most likely too flexible. The first flexible modes of the fuselage appear at 1.5 Hz. *If one were to simply divide the frequency first flexible mode by 6, one would get* 0.3 Hz, *which corresponds to about* 1.5 rad./sec. *This is very close to the pitch angle bandwidth that the author was able to comfortably achieve while maintaining a gain margin of 10dB.*

7.13 Lateral Control Design for Flexible and Undamaged P-17

The mega-aileron will be used here which consists of aileron, asymmetric spoiler and differentially deflected tail. The mega-aileron was described in section 5.7.

The outputs were selected to be bank angle, ϕ, and $\dot{\beta} \approx {a_y^{cg}}/{U_o} - r$. A reduced order Mega-Aileron to Rudder Interconnect (MARI) was designed using feedforward techniques described in chapter 5.

$$MARI(s) = -\frac{G_{\beta\delta ma}(s)}{G_{\dot{\beta}\delta r}(s)} \approx \frac{-0.3(s-1.3)}{(s+1.5)} \quad 0.5 \le \omega \le 5 \quad rad./sec.$$

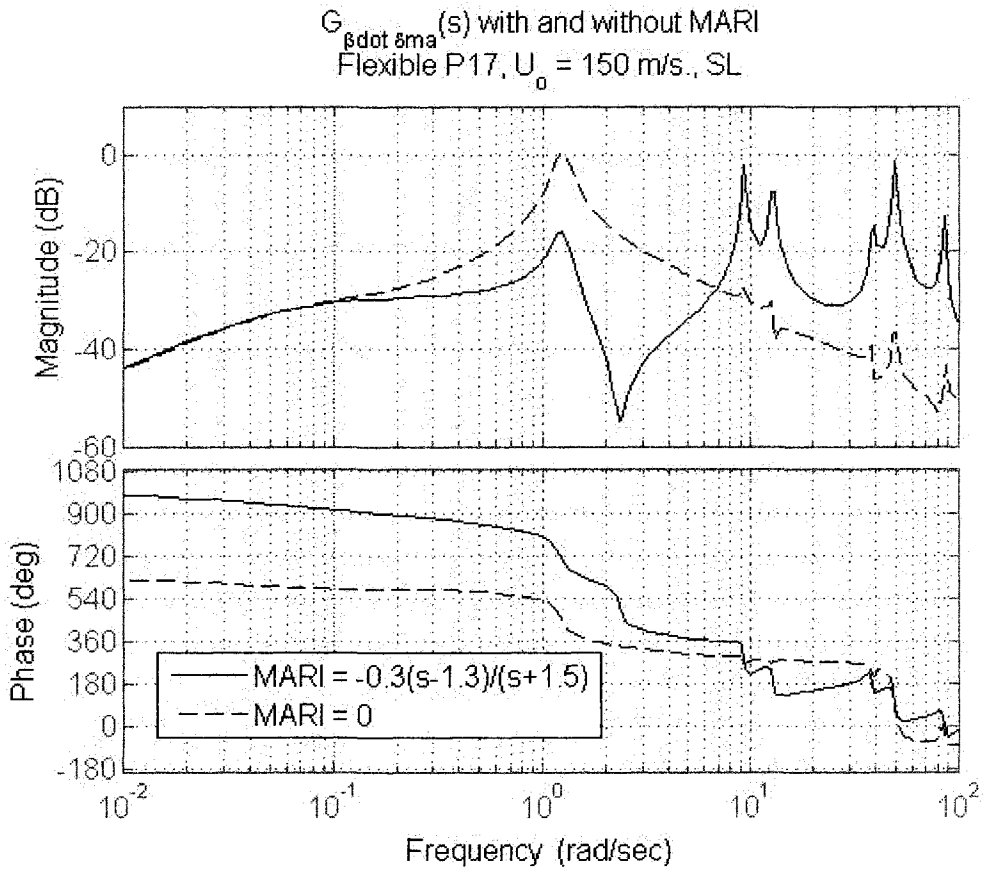


Fig. 7.33. The Mega-Aileron to Rudder Interconnect (MARI) reduces coupling of $\dot{\beta}$ and δma near the dutch roll frequency which is near 1 rad./sec. This helped significantly reduced cross coupling between the wing leveler and dutch roll damping loops. However, it increases coupling of higher frequency flexible dynamics.

After the MARI was designed, $\dot{\beta}$ was chosen to pair with the rudder to help with damping the dutch roll mode, $\dot{\beta} \rightarrow \delta r$. Because $\dot{\beta} \approx {a_y^{cg}}/{U_o} - r$ consists of lateral acceleration, a_y^{cg}, the output is fairly "high-pass" and so some filtering was done to inhibit the influence of high frequency flexible modes.

Simply feeding back a gain of 0.5 to 1 is probably sufficient to add damping to the dutch roll mode. A notch filter was added to increase the phase margin at high frequency. Without the notch filter, phase margins of about 20 degrees or less would be found at about 10 rad./sec.

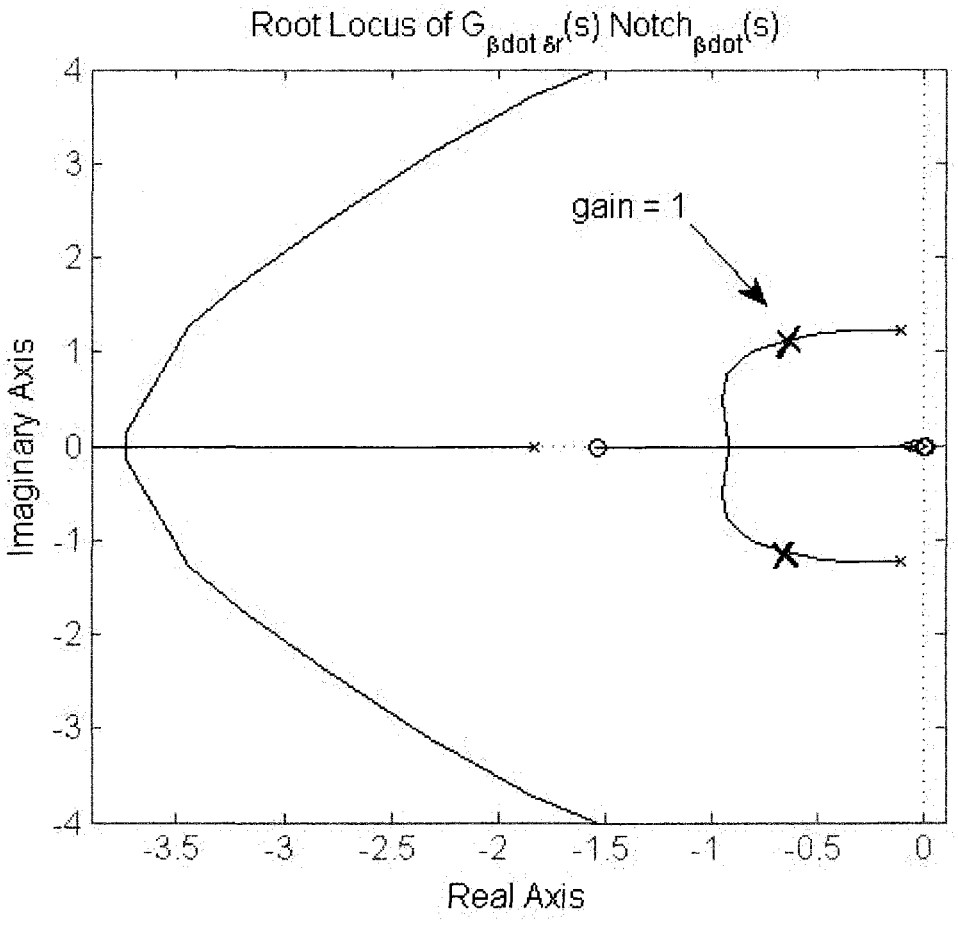


Fig. 7.34. Feedback of filtered $\dot{\beta}$ to the rudder, δr. A gain of 1 adds sufficient damping to the dutch roll mode.

Above is the root locus from shown close to the rigid body modes. However, high frequency dynamics are not highly visible from this plot.

A bode plot shows the high frequency dynamics and the gain and phase margins associated with higher frequency dynamics. The phase margin around 10 rad./sec. became moderately worrisome, so servo dynamics were included. Notice that there is a third gain margin associated with a high frequency flexible mode. This third gain margin is about 10-15 dB. Because this is close to 60 rad./sec., which will be the sampling frequency, additional filtering will be present purely as a result of the digital sampling.

$$Notch_{\dot{\beta}}(s) = \frac{3.25\left(s^2 + 3s + 123.25\right)}{\left(s^2 + 40s + 400\right)}$$

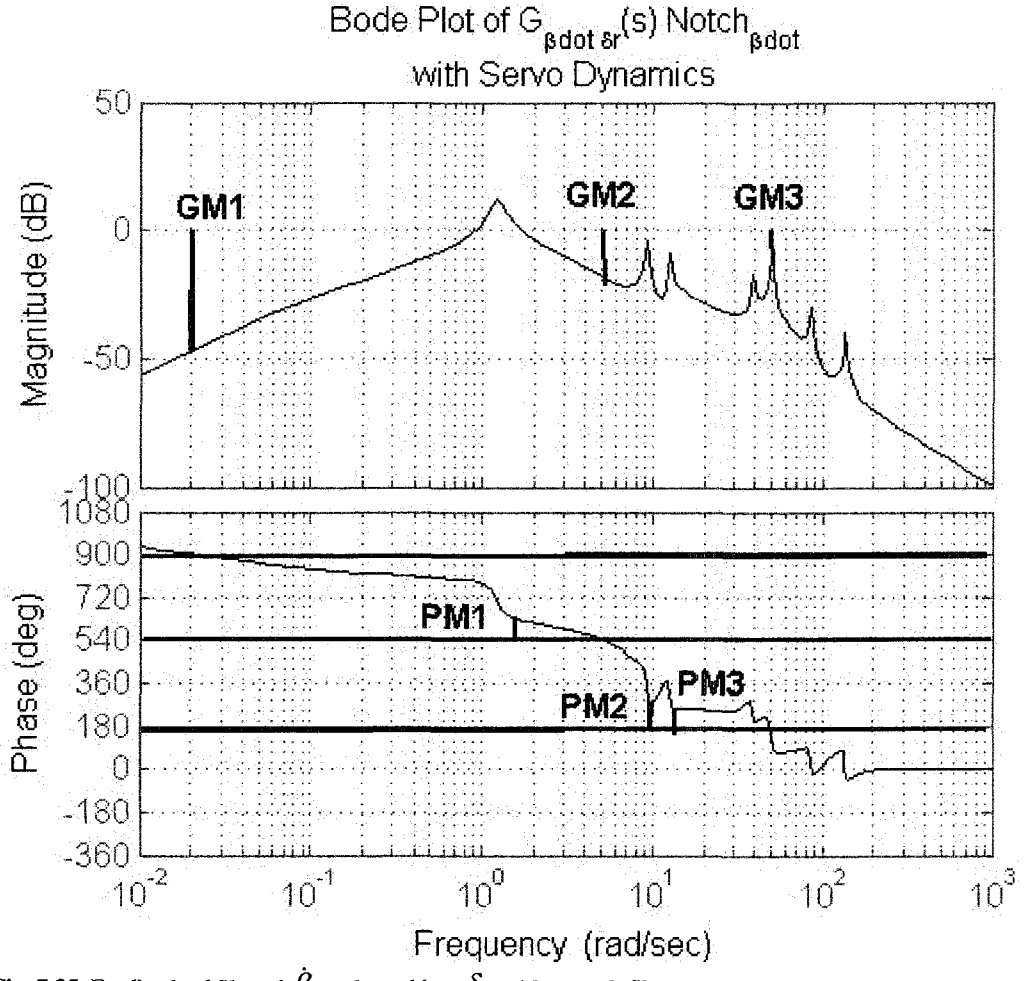


Fig. 7.35. Feedback of filtered $\dot{\beta}$ to the rudder, δr with a notch filter.

This notch filter significantly improves the phase around 10 rad./sec. In the P17 model, there were structural modes of the fuselage corresponding to about 1.7 Hz, which is pretty slow. It is reasonable to suggest that the fuselage would actually be stiffer than this.

We move on to the next step in our design which is the wing leveler, which will feed bank angle, ϕ, into the mega-aileron δma and into the rudder via the MARI.

Figure 7.36 shows an open loop bode plot of the $G_{\phi\delta ma}(s)\big|_{\dot{\beta}\to\delta r}$ loop. Notice that there is a pair of oscillatory zeros which precede the flexible mode at around 40 rad./sec. This is most likely a twist mode along the wings.

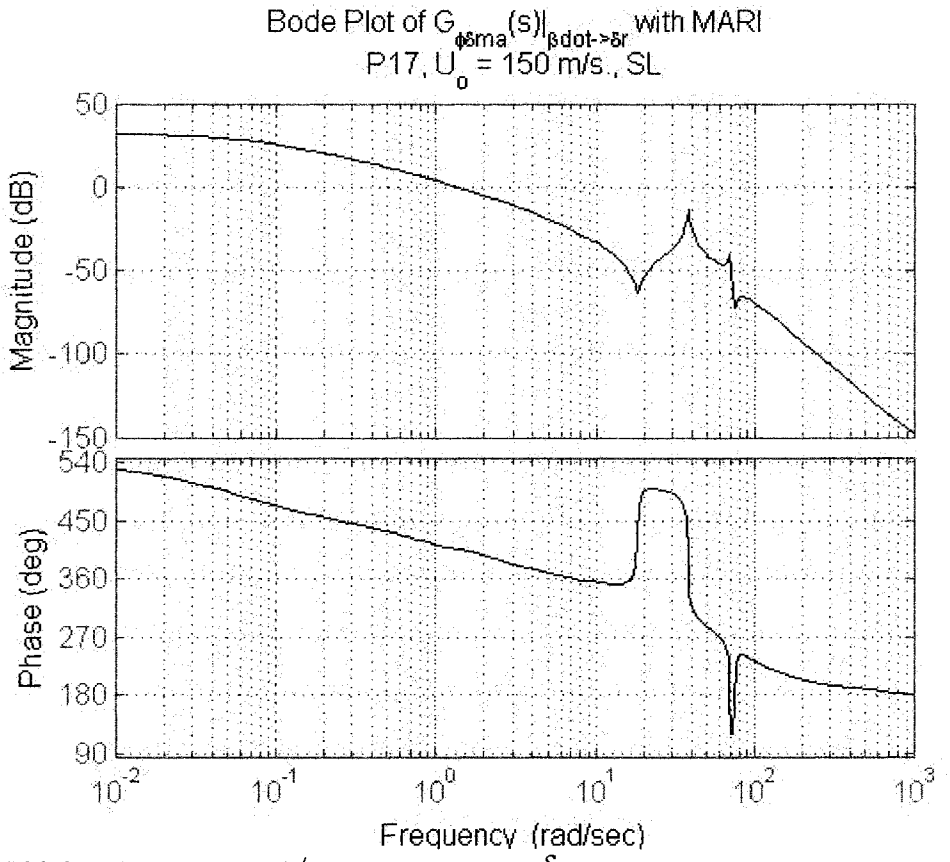


Fig. 7.36. Open loop bode plot of ϕ to the mega-aileron, δma, with the Mega-Aileron Interconnect (MARI) and the $\dot{\beta} \to \delta r$ loop closed.

A notch filter for the $\phi \to \delta ma$ loop was designed below using root locus and bode plots.

$$Notch_{\phi}(s) = \frac{1.65\left(s^2 + 3.5s + 1228.1\right)}{\left(s^2 + 80s + 2025\right)}$$

After the notch filter was designed, a typical PID wing leveler with a filter was designed after the notch filter was designed.

$$C_{\delta ma\phi}(s)=\frac{-20(s+0.06)(s+1.5)}{s(s+20)}$$

This achieves a bandwidth of about 2 rad./sec. which should be sufficient to achieve adequate handling qualities for the P17.

The wing leveler was converted to a digital with the bilinear, or tustin transform at 0.1 seconds with the "c2d( )" subroutine.

$$C_{\delta ma\phi}(z,0.1)=\frac{-10.7822(z-0.994)(z-0.8605)}{z(z-1)}$$

It is assumed that the notch filters will remain in the continuous domain and will be implemented with an analog circuit. The PID wing leveler and the MARI will be implemented in the digital domain.

It might also be possible to emulate the analog notch filters more accurately by using a higher samping rate. However, the author knows that this could create more painful problems as more flexible modes will become visible to the controller. A sampling rate of 0.1 sec. is already a faster sampling rate than the author would prefer. If pilot opinion were of no consequence, then a sampling rate of 0.25 sec. or slower would be excellent because most flexible modes would not be an issue.

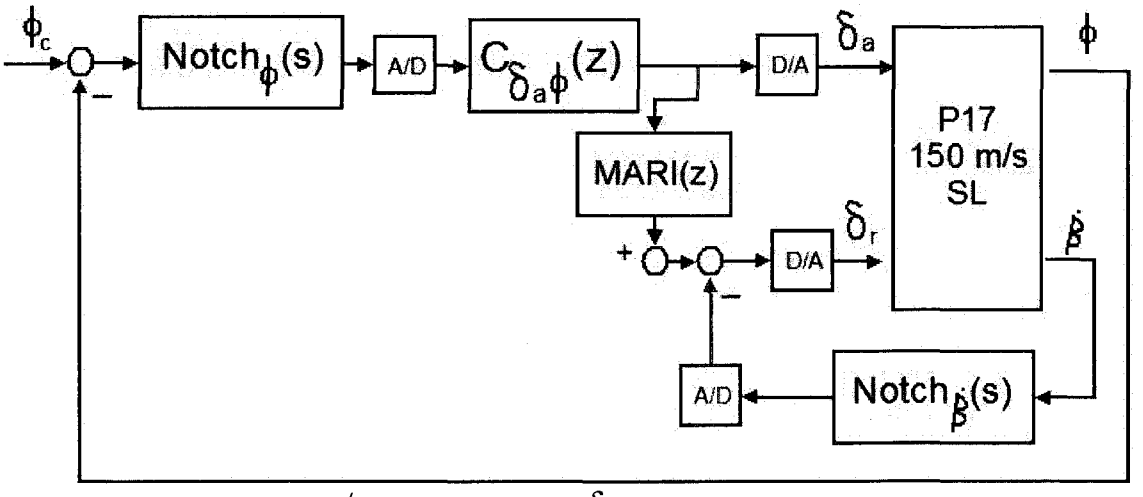
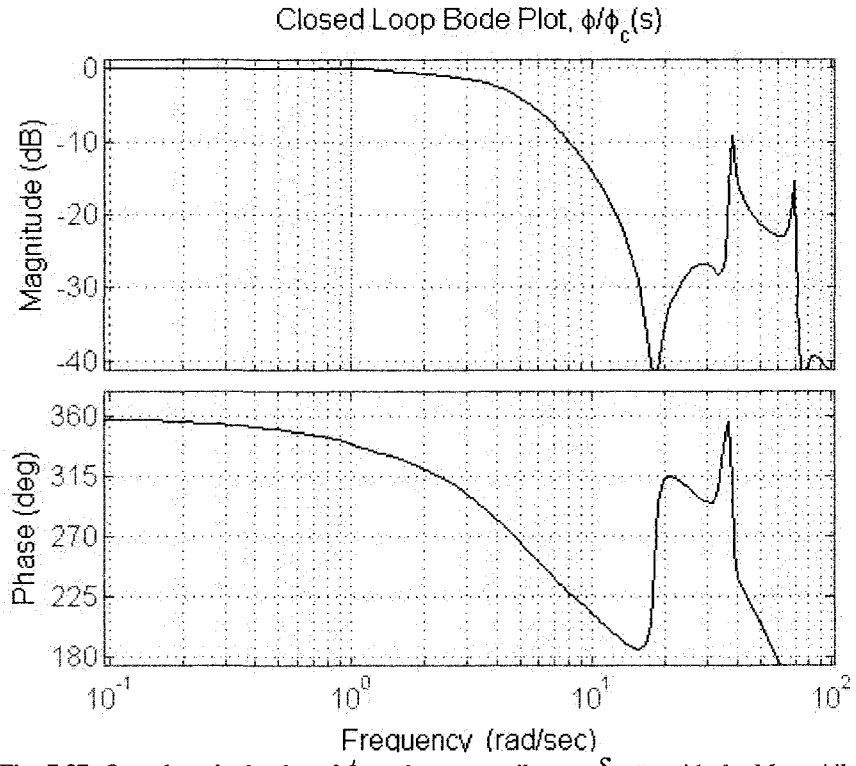


Fig. 7.37. Open loop bode plot of ϕ to the mega-aileron, δma, with the Mega-Aileron Interconnect (MARI) and the $\dot{\beta} \rightarrow \delta r$ loop closed.

$$MARI(z,0.1)=\frac{-0.26093(z-1.139)}{(z-0.8605)}$$



A closed loop bode plot in the continuous domain is shown to the left. The achieved bandwidth is approximately 2.0 rad./sec., which should hopefully fulfill handling qualities requirements.

Fig. 7.37. Open loop bode plot of ϕ to the mega-aileron, δma, with the Mega-Aileron Interconnect (MARI) and the $\beta \rightarrow \delta r$ loop closed.

7.14 Longitudinal Control Design for Flexible and Undamaged P17

The longitudinal control design for the flexible P17 was a very challenging task. The fuselage was simply too flexible. The first flexible mode was slow and appeared at 1.5 Hz followed by two others at 2.5 Hz and 5 Hz. These first three to four flexible modes were very bothersome when performing the control design. The controllers were designed to meet a gain margin of at least 8dB to 10dB first and foremost. A secondary objective was meeting handling qualities requirements, which was most likely not met with the final control designs. Other studies have shown that handling qualities will deteriorate with slow flexible modes [7.9,3.18].

Note that sensor blending was needed in this case. The author could not simultaneously achieve a reasonable bandwidth and maintain a gain margin of 10dB with only the pitch sensor placed at the center of gravity, c.g. A pitch sensor placed 12.5 meters aft of the c.g. and a pitch sensor at the c.g. were simply averaged as $\theta = \left(\theta_{cg} + \theta_{aft}\right)/2$.

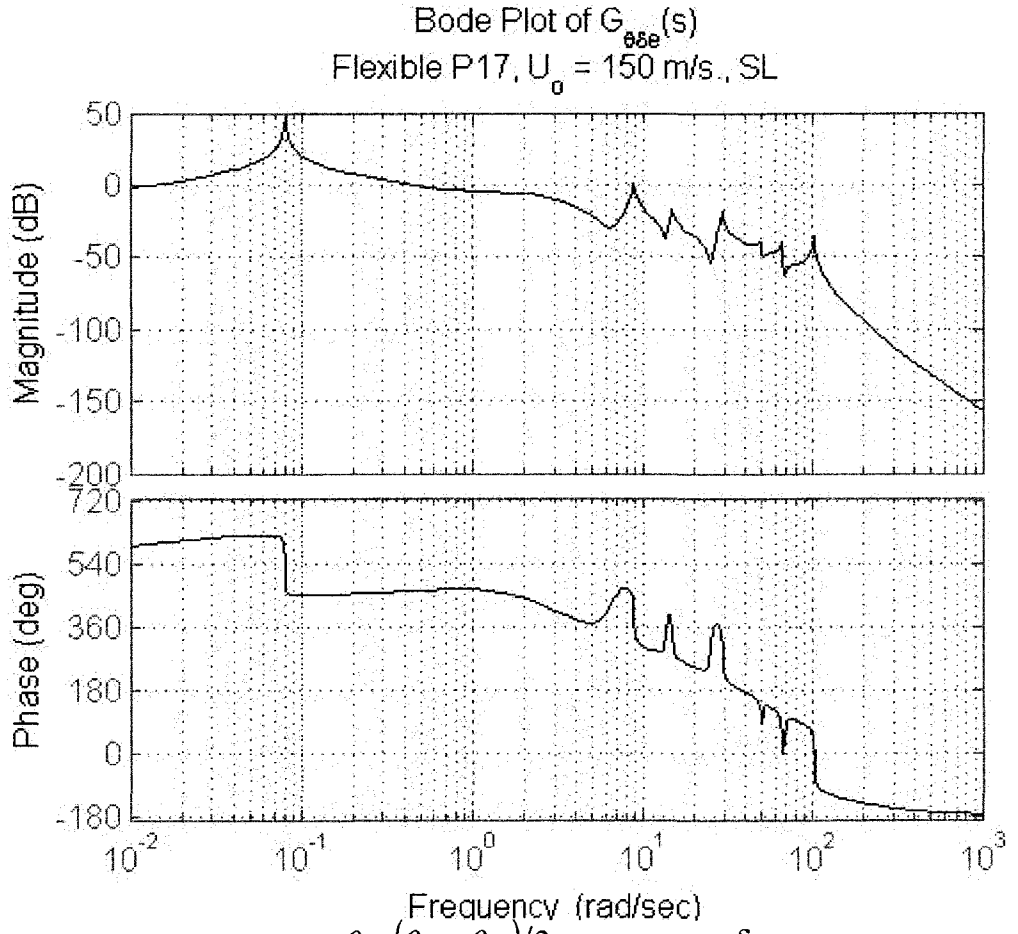


Fig. 7.38. Open loop bode plot of $\theta = (\theta_{cg} + \theta_{afi})/2$ to the elevator, δe. A gain margin of 10dB could not be achieved using only the pitch sensor at the c.g. Therefore, some sensor blending was necessary.

Even with sensor blending, a notch filter was necessary and is shown below.

$$Notch_{\theta}(s) = \frac{2.371(s^2 + s + 46.5)}{(s+10.5)^2}$$

Using the notch filter in series with the following controller, marginal handling qualities were achieved.

$$C_{\delta e\theta}(s) = \frac{-14(s+0.1)(s^2 + 3s + 6.25)}{(s+0.9)(s+20)}$$

A closed loop bode plot is shown below. Notice the awful influence of flexible dynamics in the closed loop response. Not much can robustly be done about this problem. One could perform close pole-zero cancellations of flexible modes near the $j\omega$ axis. However, this will not be very robust.

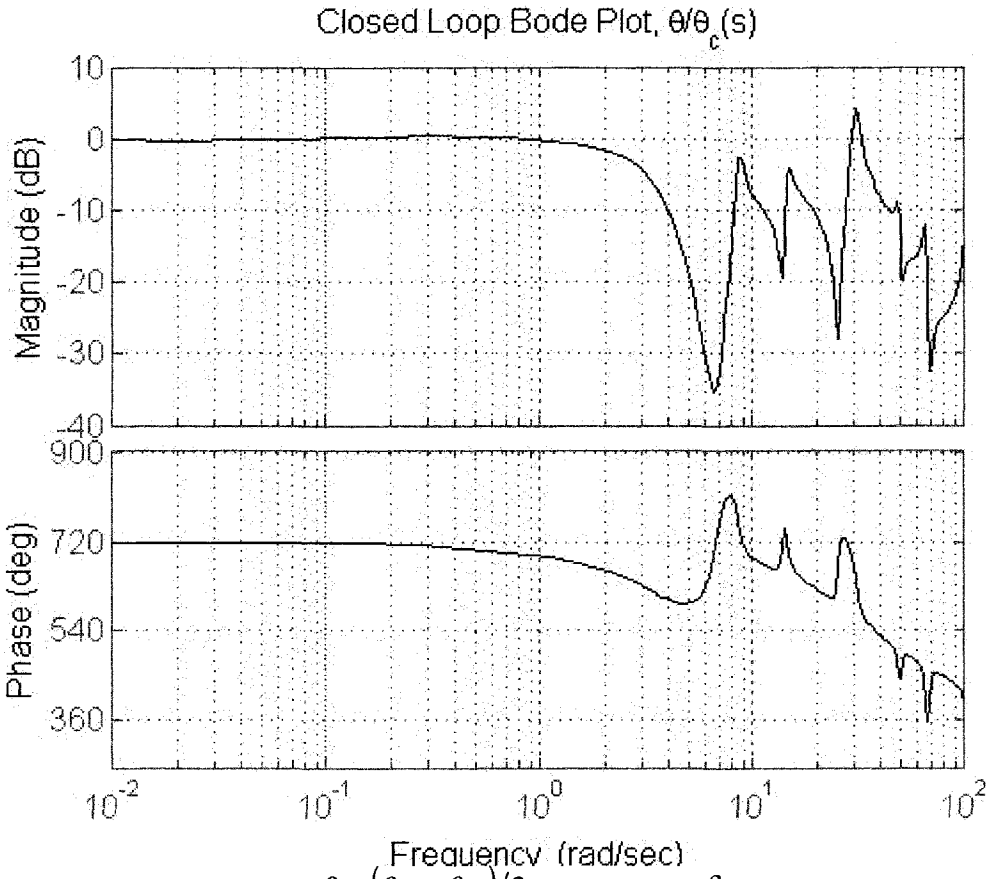


Fig. 7.39. Open loop bode plot of $\theta = (\theta_{cg} + \theta_{aft})/2$ to the elevator, δe. A gain margin of 10dB could not be achieved using only the pitch sensor at the c.g. Therefore, some sensor blending was necessary.

7.15 Stability Investigation of Flexible and Damaged P17

Large variation in structural frequencies could potentially destabilize notch filtered controllers for flexible aircraft. Therefore, the variation in structural frequencies with wing damage was investigated.

Table. 7.3. Variation of Structural Frequencies for P-17 with Damage

	dmg = 0	dmg = 0.10	dmg = 0.20	dmg = 0.30	dmg = 0.40
Mode 1 (Hz)	1.40	1.43	1.46	1.49	1.50
2	1.47	1.48	1.49	1.50	1.51
3	2.01	2.13	2.23	2.30	2.37
4	2.31	2.55	2.86	3.24	3.7
5	4.16	4.62	4.62	4.63	4.64
6	4.62	4.68	5.24	5.62	5.79
7	6.01	6.30	6.65	7.33	7.57
8	6.16	6.65	7.06	7.50	8.04

Closed loop stability with the previously designed controller was also investigated in the range of 0 to 40% wing damage. All damage scenarios were stable in the closed loop.

With partial wing loss, it is more likely that *static* instability will onset before before *dynamic* instability (i.e. an unstable eigenvalue). The rolling moments become quite large at about 20 to 30% damage. With the P17 data, it is unlikely the aircraft will survive at 40% wing damage.

Table. 7.4. Closed Loop Stability of flexible P17 with Damage

Dmg	Closed loop stable	Largest real part of closed loop poles
0	Yes	-0.0076
0.10	Yes	-0.0072
0.20	Yes	-0.0069
0.30	Yes	-0.0066
0.40	Yes	-0.0061

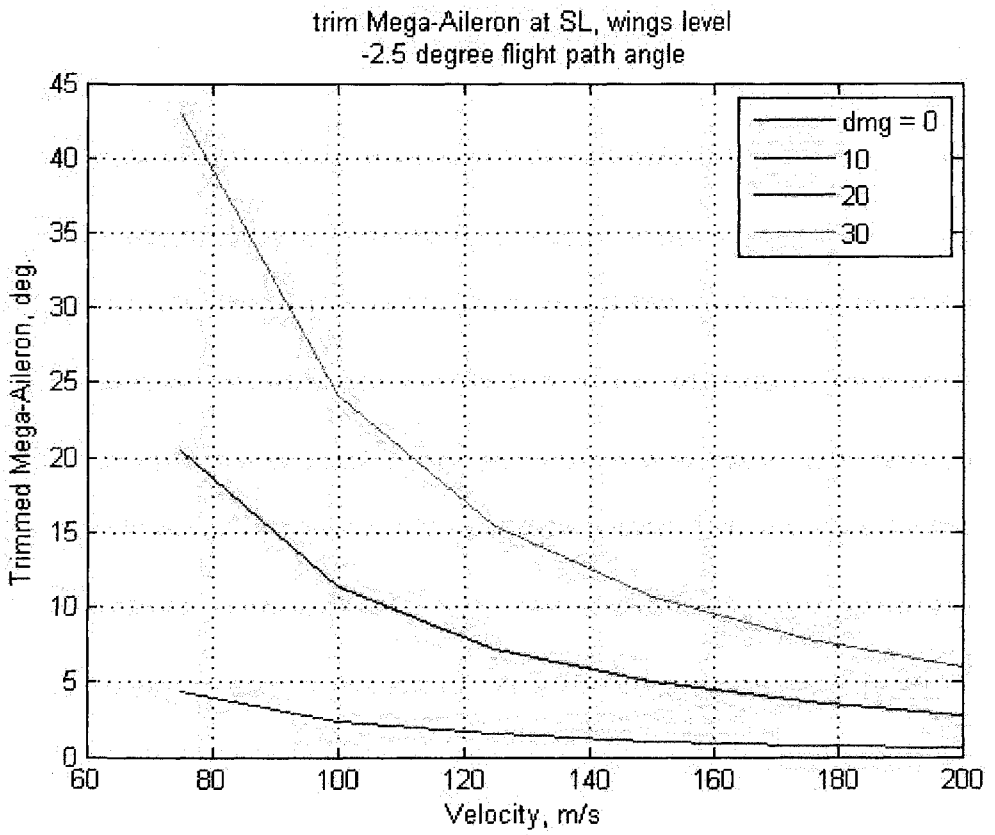


Fig. 7.40. It takes a lot of effort from the mega-aileron to counteract basic rolling moments produced by wing damage. This data comes from rigid body dynamics. Trimmed sideslip angles were less than 1 degree. Trimmed angle of attack angles did deviate by more than 2 degrees with the damage less than 30%. This is for the aircraft without any cargo. With cargo, it is expected that the trimmed mega-aileron deflection would increase by about 5 to 10 degrees at low speed because the lateral c.g. shift would be less pronounced.

Chapter 8 – Time Varying Gain and Switching Control

8.1 Stability of Swichted Linear Systems

Stability of linear time varying systems is different than standard linear time invariant systems [1.28]. In this section, we discuss the stability behavior of two switched linear systems in free response.

We define the matrix exponential below in eq. 8.1.

$$e^{At} \equiv I + {(At)}/{1!} + {(At)^2}/{2!} + {(At)^3}/{3!} + {(At)^4}/{4!} + \ldots \tag{8.1}$$

We now suppose that there are two linear A matrices, $A1$ and $A2$, which are switched. $A1$ is used from times t_o to t_1, $A2$ is used from times t_1 to t_2, $A1$ is used from times t_2 to t_3, and $A2$ is used from times t_3 to t_4. We can then write out the state history as follows below in eq. 8.2.

$$\begin{aligned}
x(t_1) &= e^{A1(t_1-t_o)}x(t_o) \\
x(t_2) &= e^{A2(t_2-t_1)}x(t_1) = e^{A2(t_2-t_1)}e^{A1(t_1-t_o)}x(t_o) \\
x(t_3) &= e^{A1(t_3-t_2)}x(t_2) = e^{A1(t_3-t_2)}e^{A2(t_2-t_1)}e^{A1(t_1-t_o)}x(t_o) \\
x(t_4) &= e^{A2(t_4-t_3)}x(t_3) = e^{A2(t_4-t_3)}e^{A1(t_3-t_2)}e^{A2(t_2-t_1)}e^{A1(t_1-t_o)}x(t_o)
\end{aligned} \tag{8.2}$$

It is not in our best interests to generalize, so we use,

$t_n - t_{n-1} = t_{n-1} - t_{n-2} = \ldots = t_1 - t_o = {T_{sw}}/{2}$, where T_{sw} is the switching period. We also define the switching matrix, $SWM(T_{sw})$, below in eq. 8.3.

$$SWM(T_{sw}) = e^{A2(T_{sw}/2)}e^{A1(T_{sw}/2)} \tag{8.3}$$

We will now decompose the switching matrix in terms of a diagonal matrix of eigenvalues, D_λ, and eigenvectors V.

$$SWM(T_{sw}) = VD_\lambda V^{-1} \tag{8.4}$$

Now, we can calculate the state vector at time t_{2n} in terms the switching matrix in eq. 8.5.

$$x(t_{2n}) = [SWM(T_{sw})]^n x(t_o) \tag{8.5}$$

Using the eigenvector representation from earlier, we can write out the switching matrix raised to the n^{th} power in eq. 8.6.

$$\begin{aligned}[SWM(T_{sw})]^n &= VD_\lambda V^{-1}VD_\lambda V^{-1}VD_\lambda V^{-1}\ldots VD_\lambda V \\ [SWM(T_{sw})]^n &= V[D_\lambda]^n V^{-1}\end{aligned} \tag{8.6}$$

Now, we want to figure out what happens at infinite time and we just write it out in eq. 8.7.

$$\lim_{n\to\infty} x(t_{2n}) = \lim_{n\to\infty}[SWM(T_{sw})]^n x(t_o) = \lim_{n\to\infty} V[D_\lambda]^n V^{-1}x(t_o) \tag{8.7}$$

$$\begin{aligned}&\text{if} \quad \rho[SWM(T_{sw})] = \left\|\bar{\lambda}[SWM(T_{sw})]\right\| < 1 \quad \text{then} \quad x(\infty) = 0 \\ &\text{if} \quad \rho[SWM(T_{sw})] = \left\|\bar{\lambda}[SWM(T_{sw})]\right\| > 1 \quad \text{then} \quad x(\infty) = \infty\end{aligned} \tag{8.8}$$

The maximum eigenvalue of a matrix is also called the spectral radius and is given the symbol, ρ as in eq. 8.9.

$$\rho[SWM(T_{sw})] = \left\|\bar{\lambda}[SWM(T_{sw})]\right\| \tag{8.9}$$

Example 8.1

A large widebodied transport had a yaw regulator shown above [2.21]. There was no feedback to ailerons at all. Notice that there is a switch depending on whether or not the flaps are up or down. We will see what happens when this autopilot is switched at different frequencies. We show a frequency domain plot of the spectral radius, or maximum eigenvalue of the switching matrix. We notice that the maximum eigenvalue, or spectral radius definitely exceeds 1 around 1 rad./sec. This is close to the dutch roll frequency of the system. When the switching frequency is right around 1 rad./sec., instability develops. This is not to be expected in actual service.

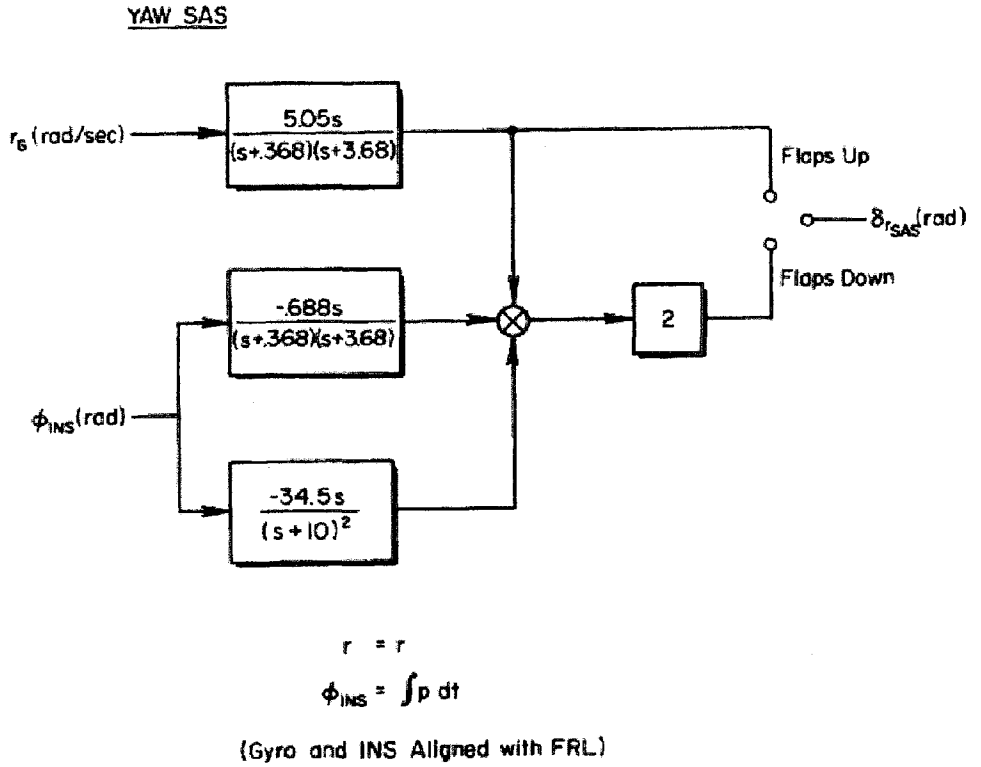


Fig. 8.1. There was a switch in the yaw regulator for a large widebodied transport. Taken from Ref. [2.21] and used with permission from NASA.

If all of the eigenvalues of the switching matrix are all below one, then the series will converge. However, if any one of the eigenvalues of the switching matrix is above one, then the series will diverge.

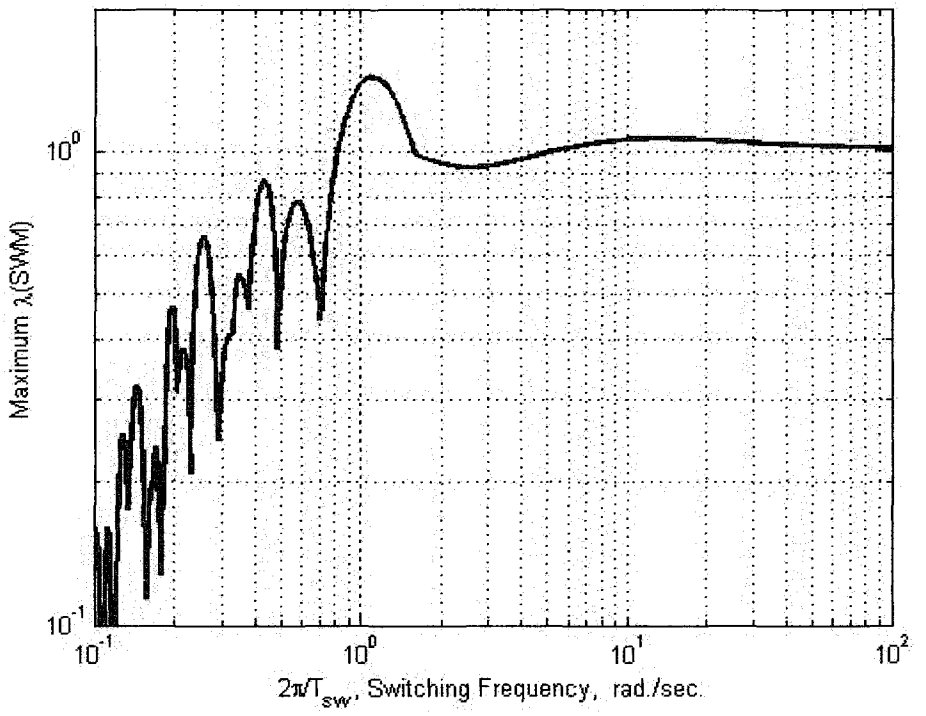


Fig. 8.2. Notice that the maximum eigenvalue of the switching matrix, SWM, exceeds 1 at around 1 rad./sec. This indicates an instability which is brought about purely by switching controllers.

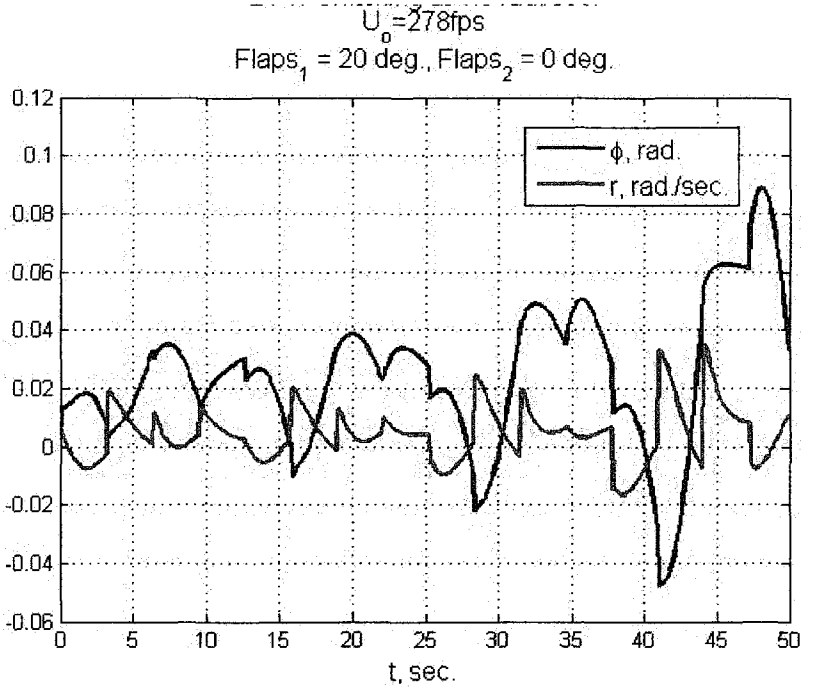


Fig. 8.3. Simulation confirms that instability will develop with switching at 1 rad./sec.

The two controllers are listed below.

$$C_2(s)=\begin{bmatrix} 0 \\ \dfrac{-5.05s}{(s+0.368)(s+3.68)} \end{bmatrix}$$

$$C_1(s)=\begin{bmatrix} \dfrac{-10.10s}{(s+0.368)(s+3.68)}+\dfrac{-34.5s}{(s+10)(s+10)} \\ \dfrac{-5.05s}{(s+0.368)(s+3.68)} \end{bmatrix}$$

The open-loop state space of the lateral dynamics of large wide-bodied transport is listed below.

Open Loop Lateral Dynamics of large wide-bodied transport at Sea-Level, 278 feet per sec.

$$A=\begin{bmatrix} -0.0997 & 0 & -1.0000 & 0.1457 \\ -1.6300 & -1.1000 & 0.1980 & 0 \\ 0.2470 & -0.1250 & -0.2290 & 0 \\ 0 & 1.0000 & 0 & 0 \end{bmatrix}$$

$$B=\begin{bmatrix} 0 & 0.0182 \\ 0.3180 & 0.1100 \\ 0.0300 & -0.2330 \\ 0 & 0 \end{bmatrix}$$

$$C=\begin{bmatrix} 0 & 0 & 0 & 1.0000 \\ 0 & 0 & 1.0000 & 0 \end{bmatrix}$$

Connecting the two controllers in feedback, yields the two closed loop state space matrices A1 and A2 below.

$$A1=\begin{bmatrix} -0.0608 & -0.7883 & 2.2528 & 0.5318 & 0.4904 & -0.8050 & -0.5444 & 0.6377 \\ -13.3808 & -9.4738 & 0.7502 & -4.1907 & -6.9733 & -2.6168 & 0.6460 & 3.6166 \\ 0.0782 & 2.8316 & -8.6745 & 1.9564 & -3.2208 & 1.8786 & -2.0441 & -1.8289 \\ -0.4434 & 0.3351 & -1.6676 & -0.1729 & -1.1118 & 0.3042 & 1.6750 & -0.4334 \\ 0.0189 & 0.9462 & -3.3864 & -0.5384 & -3.1806 & 1.2919 & -1.0510 & -1.3325 \\ 0.3842 & 0.0030 & -0.1151 & -1.0052 & -0.0530 & -0.5130 & 0.6126 & 0.3439 \\ -0.0198 & 0.0543 & -0.1888 & -0.0807 & 0.0479 & -0.4711 & -2.8998 & -0.5202 \\ -0.2688 & 0.0630 & -0.1520 & 0.5449 & -0.0839 & 0.1463 & -1.2241 & -0.5012 \end{bmatrix}$$

$$A2=\begin{bmatrix} -0.5504 & -0.8007 & -0.3524 & 1.0189 & -1.6176 & -0.8115 & 0.1704 & -1.5021 \\ -1.9079 & -7.9554 & 3.3776 & -2.5346 & -0.9615 & -1.2098 & 0.1687 & -0.4096 \\ -0.0577 & 3.4509 & -5.4177 & -0.6580 & -1.5557 & -0.2483 & 3.0662 & 0.2330 \\ -0.0364 & -0.3846 & -2.1686 & -5.3849 & 2.5120 & 1.5321 & 3.3397 & 1.0282 \\ 0.0127 & 0.0606 & -0.0126 & 0.6190 & -1.7609 & -0.7997 & 0.1470 & -0.1784 \\ -0.0031 & -0.0711 & 0.3899 & 0.2863 & 0.3950 & 0.1103 & -0.1815 & 0.1767 \\ -0.0864 & -0.9519 & 3.8000 & 2.7305 & 1.0234 & 0.5316 & -3.2579 & 0.1607 \\ -0.0631 & -0.2503 & -0.2833 & -0.2782 & -0.0071 & -0.1154 & -1.6016 & -1.2598 \end{bmatrix}$$

MATLAB code used to generate the previous figures is shown below.

Example 8.1 – MATLAB code for switching large Widebodied Transport

```
clc
clear all

%B747 lateral stability derivatives
%at Mach 0.249, SL, flaps 20 deg

Uo=221;
bank=0/180*pi;
cbank=cos(bank);
Yv=-.0997;
Lbeta=-1.63;
Nbeta=.247;
Lp=-1.10;
Np=-.125;
Lr=0.198;
Nr=-0.229;

Ysda=0.0;
Lda=0.318;
Nda=0.03;
Ysdr=.0182;
Ldr=0.110;
Ndr=-.233;

Alat=[Yv 0 -1 32.2/Uo*cbank;Lbeta Lp Lr 0;
Nbeta Np Nr 0;0 1 0 0];
Blat=[Ysda Ysdr;Lda Ldr;Nda Ndr;0 0];
Clat=[0 0 0 1;0 0 1 0;Uo*Yv 0 -Uo*0 32.2*cbank];
Dlat=[zeros(2,2);Ysda*Uo    Ysdr*Uo];

syslat1=ss(Alat,Blat,[Clat(1,:);Clat(2,:)],zeros(2));

syslat2=syslat1;

s=tf('s');

f1=zpk([0],[-10 -10],-34.5);
f2=zpk([0],[-3.68 -0.368],-0.688);
x587=f1+f2;
H1=[0.01 0;2*x587 2*-5.05*s/((s+0.368)*(s+3.68))];
H2=[0.01 0;0.01*x587 1*-5.05*s/((s+0.368)*(s+3.68))];

GregK1=minreal(feedback(syslat1,H1));
GregK2=minreal(feedback(syslat2,H2));

A1=GregK1.a;
A2=GregK2.a;
```

Example 8.1 – MATLAB code for switching Large Wide-bodied Transport

```
%simulation to show instability around 1.0 rad/sec
tfin=50;
dt=0.1;
nt=tfin/dt+1;
t=linspace(0,tfin,nt);
sig_sw=square(1.0*t);

sys1d=c2d(GregK1,dt,'zoh');
sys2d=c2d(GregK2,dt,'zoh');

%IC
[order order]=size(sys1d.a);
x(:,1)=.01*ones(order,1);

for k=1:nt
    if ( sig_sw(k) == 1 )
        Acl=sys1d.a;
        Ccl=sys1d.c;
    else
        Acl=sys2d.a;
        Ccl=sys2d.c;
    end
    x(:,k+1)=Acl*x(:,k);
    y(:,k+1)=Ccl*x(:,k);
end

figure(4);
plot(t,y(:,1:nt));
grid on
legend('\phi, rad.','r, rad./sec.','FontSize',12);
xlabel('t, sec.','FontSize',12);
f789={'U_o=278fps',
    'Flaps_1 = 20 deg., Flaps_2 = 0 deg.'};
title(f789,'FontSize',12);
```

8.2 Basic Lyapunov Stability

One way to justify the stability of both linear and nonlinear systems is Lyapunov stability. In order to use Lyapunov stability, one must first have a valid Lyapunov function. Then, one can show that

1. The Lyapunov function, $L(x)$, is bounded and positive-definite everywhere:

   $c_o\|x\| \le L(x) \le c_1\|x\|, \quad c_o > 0 \quad , c_1 > 0, \quad \forall x < \infty$, where c_o and c_1 is an arbitrary positive constant.

2. The Lyapunov function is radially unbounded $\lim_{x\to\infty} L(x) = \infty$

3. The time derivative of the Lyapunov function is negative everywhere, $\dot{L}(x) < 0 \quad \forall x$.

Although it can be extended to non-linear systems, Lyapunov stability is difficult to use for non-linear systems analysis. We show the following result for unforced linear systems analysis.

$$\begin{aligned} &\dot{x} = Ax \\ &L(x) = x^T Px \quad P = P^T > 0 \\ &\dot{L}(x) = x^T \left(A^T P + PA\right)x \end{aligned} \tag{8.10}$$

Now, in order to show that the system is stable, we must solve the following Linear Matrix Inequality (LMI). Where the matrix, $M_1 = A^T P + PA$, has all of its eigenvalues in the LHP, $\mathbf{Re}\{\lambda[M_1]\} < 0$, which means that M_1 would be negative definite. We also require that P would have all of its eigenvalues to be positive, $\mathbf{Re}\{\lambda[P]\} > 0$. Thus, we say that P would be positive definite.

$$\begin{aligned} &A^T P + PA < 0 \\ &P^T = P > 0 \end{aligned} \tag{8.11}$$

These LMI are oftentimes very difficult to solve by hand. Software is available which solves these equations using constrained minimization subroutines.

8.2.1 Misapplication of Lyapunov Stability

Suppose one considered the following first order and unstable system in free response.

$$x(t) = e^t x_o \tag{8.12}$$

We propose the following time varying Lyapunov function below.

$$L(x,t) = \frac{1}{2} x^2 P(t), \quad P(t) = e^{-3t} \tag{8.13}$$

We then check the time derivative of the Lyapunov function below in eq. 8.14. Notice that the following time derivative of the Lyapunov function is negative definite, which indicates that the system is stable. However, the system is actually unstable.

$$\dot{L}(x,t)=\dot{x}(t)x(t)P(t)+\frac{1}{2}x(t)^2\dot{P}(t),\quad \dot{x}(t)=e^t x_o \quad \dot{P}(t)=-3e^{-3t}$$
$$\dot{L}(x,t)=x_o^2e^{2t}e^{-3t}-\frac{3}{2}x_o^2e^{2t}e^{-3t}=-\frac{1}{2}x_o^2e^{-t}\le 0 \tag{8.14}$$

Going back and checking our work, we see that we actually have not met property 1 of the Lyapunov function checklist. At infinite time, the Lyapunov function approaches 0 with x being non-zero, $\lim_{t\to\infty} L(x,t)=0 \quad \forall x$.

8.2.2 Application of Lyapunov Stability

Using complicated Lyapunov functions for complicated time varying or non-linear systems is difficult and prone to misinterpretation. For linear time varying systems, one can definitely show stability if the following condition is met below.

$$\begin{aligned}&\dot{x}=A(t)x\\&PA(t)+A(t)^TP<0 \quad \forall t\\&P^T=P>0\end{aligned} \tag{8.15}$$

Generally, it is best to avoid making the P matrix a function of time. The preceding equations imply a large and fairly difficult LMI feasibility problem which is discussed in Ref. [2.10].

An important thing to keep in mind about Lyapunov stability for non-linear and time varying systems is that it is a sufficient but not necessary condition for stability. This means that one does not absolutely need to have found a Lyapunov function to show stability. Experimental means can always show stability.

However, for linear time invariant systems, it is both necessary and sufficient that a positive definite P satisfies $A^TP+PA<0$. But it is a lot easier to simply check the eigenvalues of the A matrix.

8.3 SISO Strict Positive Realness

Prior to studying Lur'e systems, we will discuss the result of Strict Positive Realness (SPR) and what this means. The Meyer-Kalman-Yakubovich (MKY) lemma for SISO systems is listed below.

$$\begin{aligned} &\text{if} \quad Re\{G(j\omega) = C(j\omega I - A)B + D\} > 0 \quad \forall \quad 0 \le \omega \le \infty \\ &\text{then} \quad \exists \quad P^T = P > 0 \quad : \quad \begin{cases} A^T P + PA < 0 \\ C = B^T P \\ D \ge 0 \end{cases} \end{aligned} \tag{8.16}$$

SPR qualities are used in direct adaptive control stability proofs that basically use infinite gain or nearly infinite gain. In the real world, this is just cannot happen. However, we will briefly show the mathematical Lyapunov stability result below as a warm up and a purely theoretical exercise.

$$\dot{k} = y^2, \quad u = -ky$$

$$\begin{aligned} &L = x^T Px + k^2 \\ &\dot{L} = x^T (A^T P + PA)x - 2x^T PBkCx + 2k\dot{k} \\ &\dot{L} = x^T (A^T P + PA)x - 2x^T PBkCx + 2ky^2, \quad y = Cx = B^T Px, \quad y^2 = y^T y \\ &\dot{L} = x^T (A^T P + PA)x - 2kx^T PBCx + 2kx^T PBCx, \quad \text{MKY} \\ &\dot{L} = x^T (A^T P + PA)x < 0 \end{aligned} \tag{8.17}$$

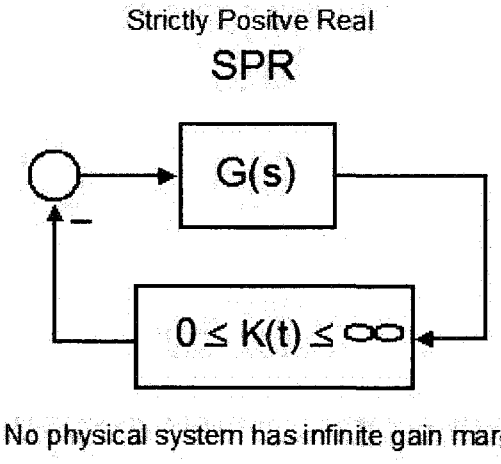


Fig. 8.4. If any system is SPR, then any nonlinear feedback law , *NL*, which varies between 0 and infinity will work. Obviously, this cannot be.

One should note that it would be a very bad idea to actually use this direct adaptive control algorithm without gain limiting. Almost all implementations of direct MRAC will basically use infinite or very high gain.

One can quickly assess the strict positive realness of a transfer function by either a nyquist plot or a bode plot. Basically, the phase must remain between -90 and 90 degrees across all frequencies. For

a system with even the smallest time delay, this will be violated. A bode plot of $1/(s+1)$ with a time delay at 100Hz is shown below in Fig. 8.5.

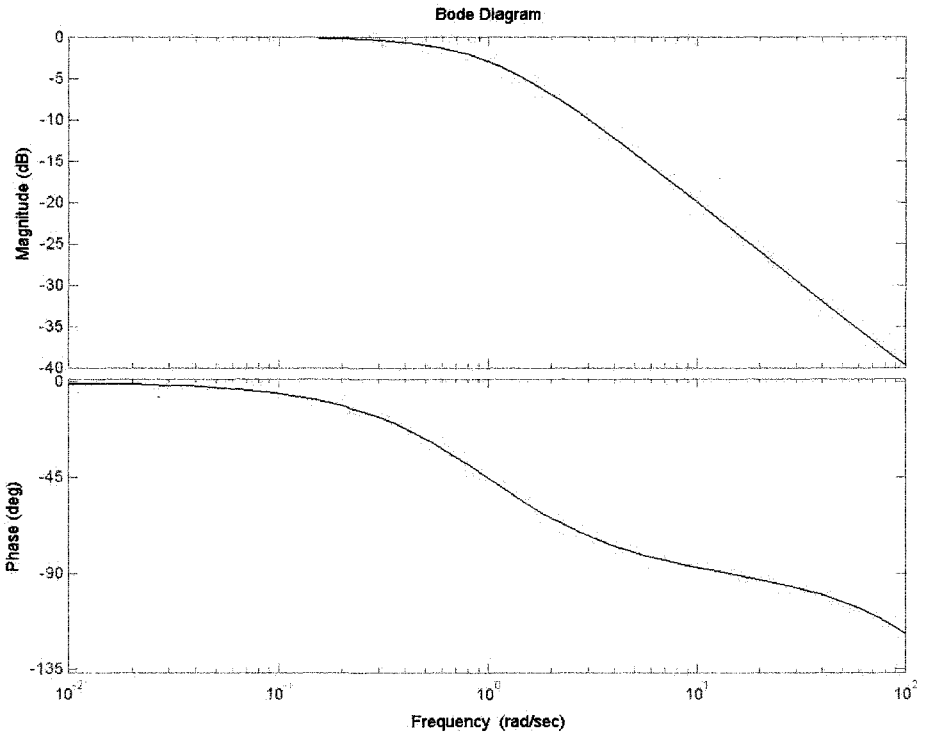


Fig. 8.5. The bode plot of $G(s)=\frac{1}{s+1}\frac{-(s-200)}{(s+200)}\approx\frac{e^{-0.01s}}{s+1}$ is not SPR. Since all physical systems have a time delay at some frequency, there is no such thing as a physical SPR system.

8.4 Lur'e Systems: SISO Circle Criterion

Strict positive realness is too strict. Infinite gain is too big. It is reasonable to suppose that if one used a gain less than infinity that things would be stable. Sure enough, when one uses non-linear or time varying gains less than infinity, then one can show stability for a physical system. This is called the circle criterion.

We will reiterate the circle criterion from ref. [2.16]. We make the following assumptions below.

1. The linear system is stable and has a nyquist plot that lies strictly to the right of $-1/k_2$, i.e. $\boldsymbol{Re}\{C(j\omega I-A)^{-1}B+D\}<-1/k_2$
2. The non-linearity fulfills the following property, $0\le yNL\le k_2 y^2 \quad \forall y$. This means that the non-linearity can be thought of as a gain which is less than k_2. The input is given by $u=-NL$. k_2 is a positive scalar quantity.

We now use the following Lyapunov function below.

$$\begin{aligned} L &= x^T Px \\ \dot{L} &= x^T \left(A^T P + PA\right)x + 2x^T PBu \end{aligned} \tag{8.18}$$

Using some clever algebra, we notice that the following is true below.

$$\begin{aligned} u\left(k_2 y + u\right) - u\left(k_2 y + u\right) &= 0 \\ u\left(k_2 Cx + kDu + u\right) - u\left(k_2 y + u\right) &= 0 \end{aligned} \tag{8.19}$$

We subtract eq. 8.19 from eq. 8.18 to give eq. 8.20.

$$\dot{L} = x^T \left(A^T P + PA\right)x + 2x^T PBu - uk_2 Cx - u\left(1 + k_2 D\right)u + u\left(k_2 y + u\right) \tag{8.20}$$

Since k_2 and u are scalars, the following manipulation can be made.

$$\dot{L} = x^T \left(A^T P + PA\right)x + 2x^T \left(PB - C^T k_2\right)u - u\left(1 + k_2 D\right)u + u\left(k_2 y + u\right) \tag{8.21}$$

We use the inequality of assumption 1 below.

$$\begin{aligned} u\left(k_2 y + u\right) &\le 0 \\ -NL\left(k_2 y\right) + NL^2 &\le 0 \end{aligned} \tag{8.22}$$

We can then use this inequality to refine eq. 8.21 below.

$$\dot{L} \le x^T \left(A^T P + PA\right)x + 2x^T \left(PB - C^T k_2\right)u - u\left(1 + k_2 D\right)u \tag{8.23}$$

This implies the following in the frequency domain below.

$$\boldsymbol{Re}\left\{C(j\omega I - A)^{-1} B + D\right\} < -1/k_2 \tag{8.24}$$

If one were to look at a nyquist plot, we would want the contour to lie to the right of $-1/k_2$.

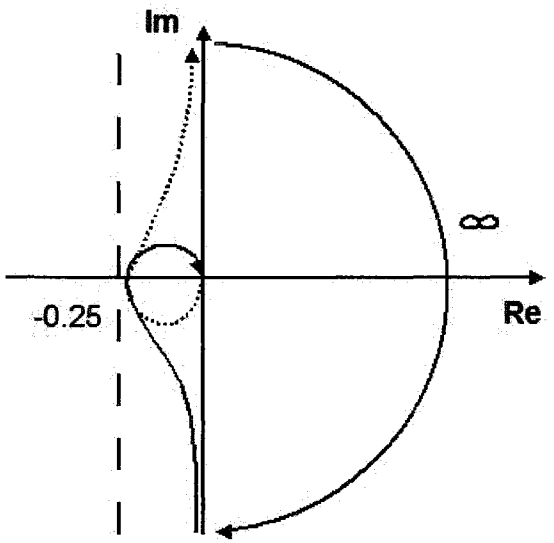
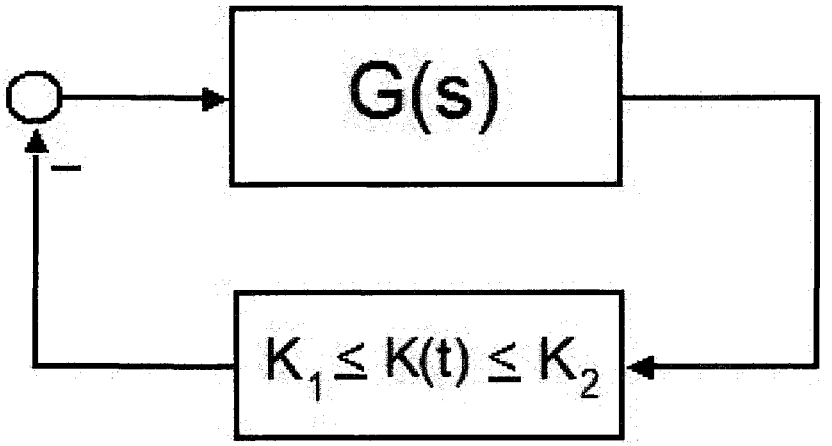


Fig. 8.6. From this nyquist contour, we can assume that a non-linear gain in the ranging between 0 and 4 will be stable. This can work on a physical system with a time delay.

Now, suppose we change the assumptions such that the following are true below.

1. The linear system is stable and has a nyquist plot that lies strictly to the right of $-1/(k_2-k_1)$, i.e. $\boldsymbol{Re}\{C(j\omega I-A)^{-1}B+D\}<-1/(k_2-k_1)$.

2. The non-linearity fulfills the following property, $k_1y^2 \le yNL \le k_2y^2$, $\forall y$. This means that the non-linearity can be thought of as a gain which is less than k_2 and greater than k_1. The input is given by $u=-NL$. k_2 and k_1 are positive scalar quantities.



Possible to use on a physical system!

Fig. 8.7. With gain limiting, one can guarantee that there is stability with time varying gain or non-linearity

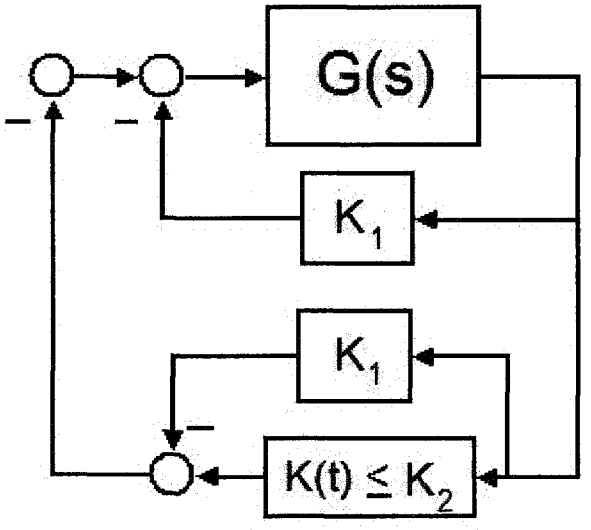


Fig. 8.8. One can use the above loop transformation to make the limited system described work like equations 8.18 to 8.24

This is equivalent to the following block diagram to the left. After some algebraic manipulations, one can show that as long as the nyquist plot does not intersect a circle with points defined by $-1/k_1$ and $-1/k_2$, then the closed loop will be stable.

Figure 8.9 shows the circle criterion applied to the Navion $\theta \rightarrow \delta_e$ control loop with a 0.1 second time delay in the loop. Notice that the margin to increase the gain is not very large, only 60% at most.

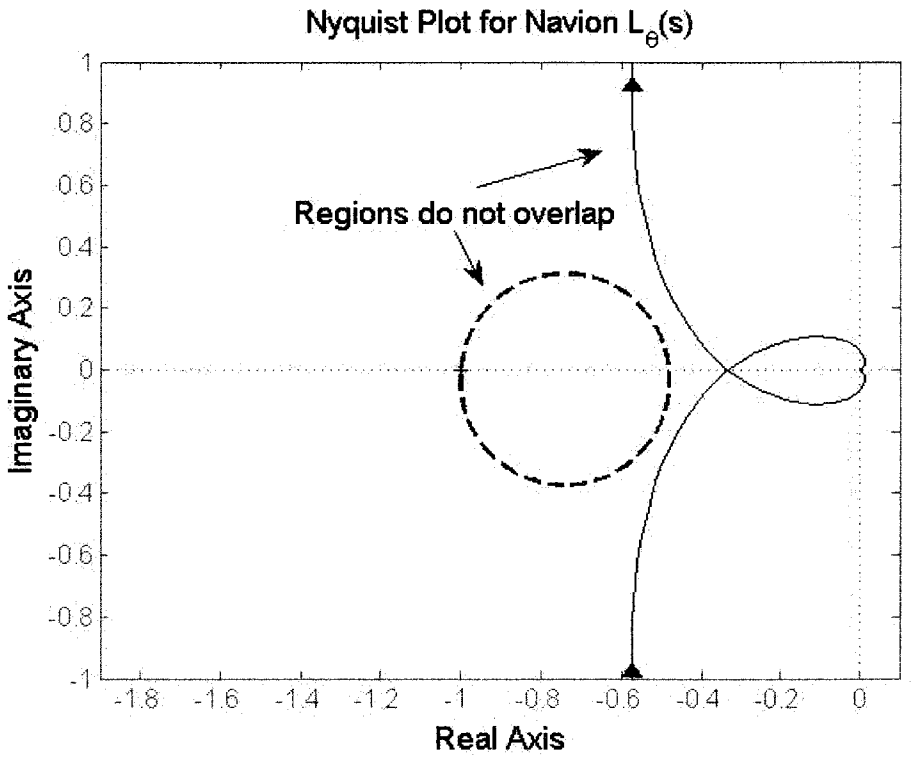


Fig. 8.9. We can see that the envelope for increasing the gain is really quite small after the controller is designed. We see that we can accomodate a random non-linear gain increase by at most 50 to 60% in flight. The gain margin of the system is around 250% or 8dB. This is uses data from the navion control design from chapter 4

8.5 Lur'e Systems: MIMO Circle Criterion for Square Systems

Conditions for MIMO Lur'e systems (i.e. a linear system with a time-varying gain) exist, though there is some variability within the academic community. For SISO systems, there is uniform agreement upon the circle criterion. We will only repeat the MIMO result from Ref. [8.1].

1. The non-linearity fulfills the following property, $(-NL-K_1 y)^T(-NL-K_2 y)\le 0$, $\forall y$. This means that the non-linearity can be thought of as an *m x m* gain which is less than K_2 and an *m x m* greater than K_1. The input is given by $u=-NL$. K_2 and K_1 are positive scalar quantities.
2. The transfer function $W(s)=(I+K_2G(s))(I+K_1G(s))^{-1}$ will be strictly positive real across all frequencies, $\lambda\{W^*(j\omega)+W(j\omega)\}>0 \quad \forall \quad 0\le\omega\le\infty$. This means

that we have to check all of the eigenvalues of $W(j\omega)$ plus its conjugate transpose, $W^*(j\omega)$ and make sure that they are positive.

We have seen that the switched analysis showed instability near the dutch roll frequency from section 8.1. The Lur'e analysis shows a similar tendency for instability near the dutch roll frequency. Below is a plot of the eigenvalues of $W(j\omega)+W^*(j\omega)$ using the data from the previous section on switched linear systems.

This has important ramifications for any type of time-varying control. We have seen in section 8.1 that if control is switched at the wrong frequency, then instability will result. This resulted even though both of the controllers would stabilize the aircraft if there were no switching.

The Lur'e analysis is different from the switching analysis in that it assumes an arbitrary norm bounded non-linearity or time varying effect. Nonetheless, it still indicates that stability can unravel at the dutch roll frequency.

This characteristic is not unique and not isolated to the large wide-bodied transport from section 8.1. Many aircraft will exhibit this problem with time varying gain at the dutch roll frequency. It does not always cause instability, but it definitely excites some strange behavior.

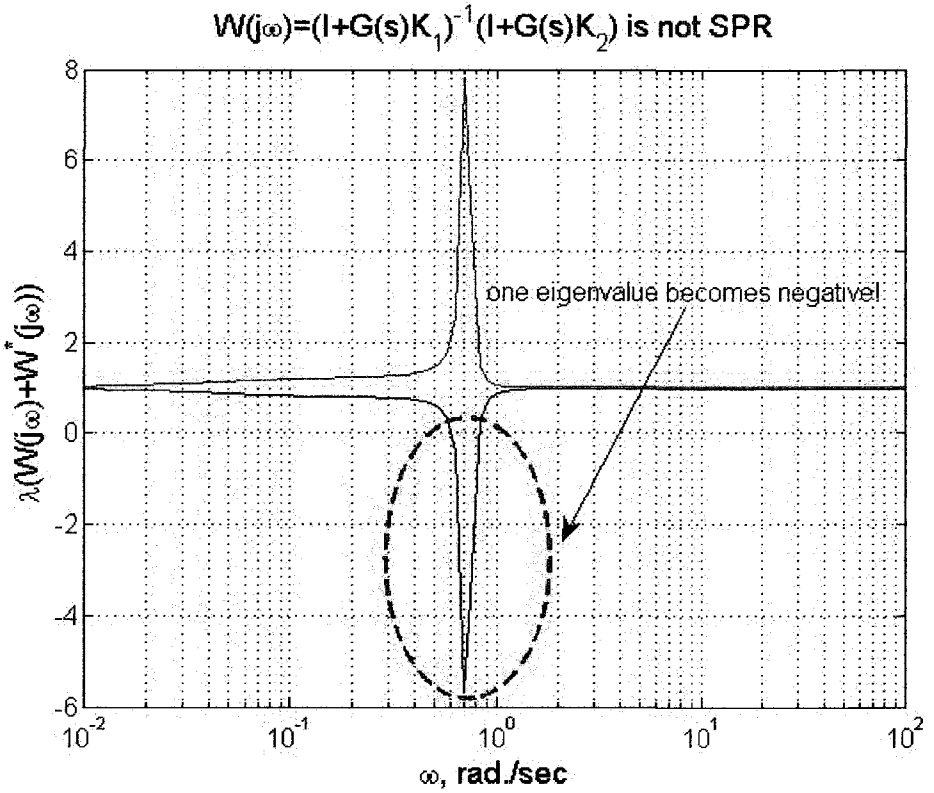


Fig. 8.10. We can see that we have not met the Lur'e condition for MIMO stability for the B-747 on approach.

We used the values of $K_1 = \begin{bmatrix} 1.1 & 0 \\ 0 & 1.0 \end{bmatrix}$ and $K_2 = \begin{bmatrix} 1.0 & 0 \\ 0 & 1.1 \end{bmatrix}$, which is a very minor variation in gain.

Note that this is not a definite indication of instability because the circle criterion is only a sufficient condition, not a necessary condition. However, we have seen from evidence proposed earlier that there is in fact an instability near the dutch roll frequency.

8.6 Relevance to Adaptive and Reconfigurable Flight Control

Adaptive and reconfigurable flight control almost always involves some kind of time-varying gain. We have seen that time varying gain can cause instability in aircraft. However, certain styles of adaptive control are more prone to instability by time varying gain than others.

Reconfigurable Control

This is the most practical method and is currently used on some aircraft. With testing, one can show that a failure rate of $1x10^{-9}$ per hour for undamaged aircraft is met.

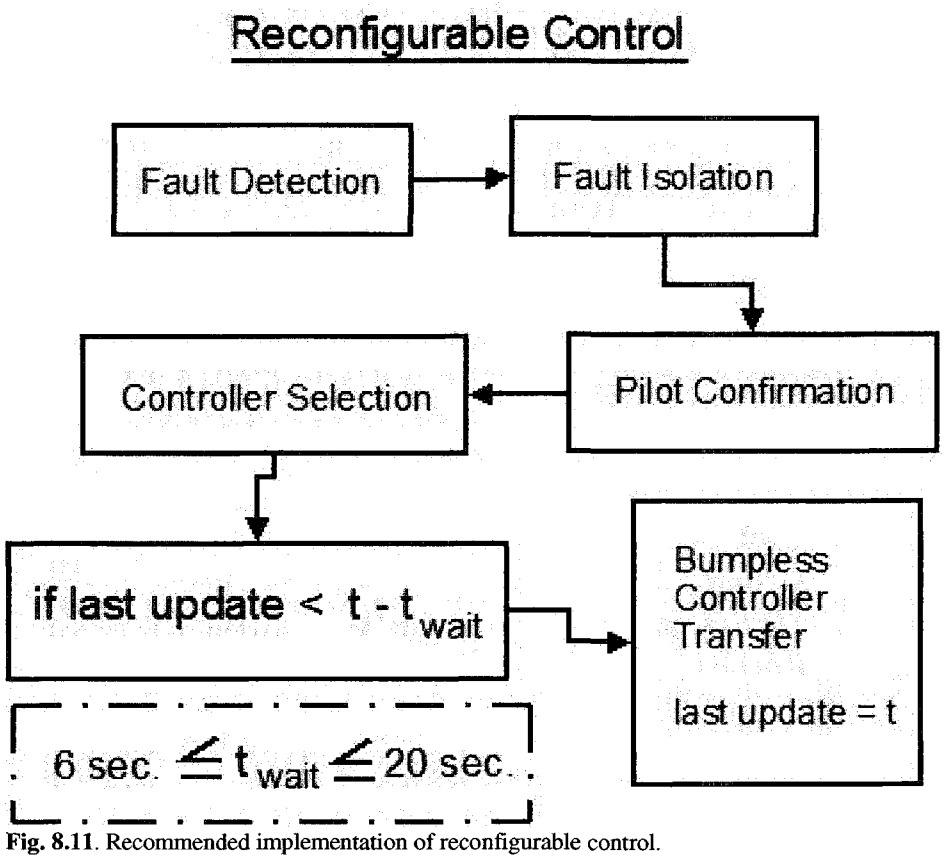


Fig. 8.11. Recommended implementation of reconfigurable control.

Recommendations for Reconfigurable Control:

1. As soon as a problem is diagnosed, one can switch the controller. However, if one wants to switch again, a waiting time of at least 6 to 20 seconds is recommended to avoid time varying gain instability for the lateral system. This corresponds to a switching period of 12 to 40 seconds.

2. Use a bumpless controller transfer, discussed in section 2.28.

3. With testing, shorter waiting times can be achieved.

Iterative Identification and Control / Iterative Learning Control

If the adaptive control is a kind of switching control, then it is somewhat reasonable to suggest that slowly switching controllers would work. It is also somewhat reasonable to suggest that it would not work. Testing and verification of this method makes it somewhat difficult. Catastrophic failure rates are usually designed to meet $1x10^{-9}$ per hour for undamaged aircraft. It is entirely likely that redesigning the autopilot in flight will cause accidents for undamaged aircraft.

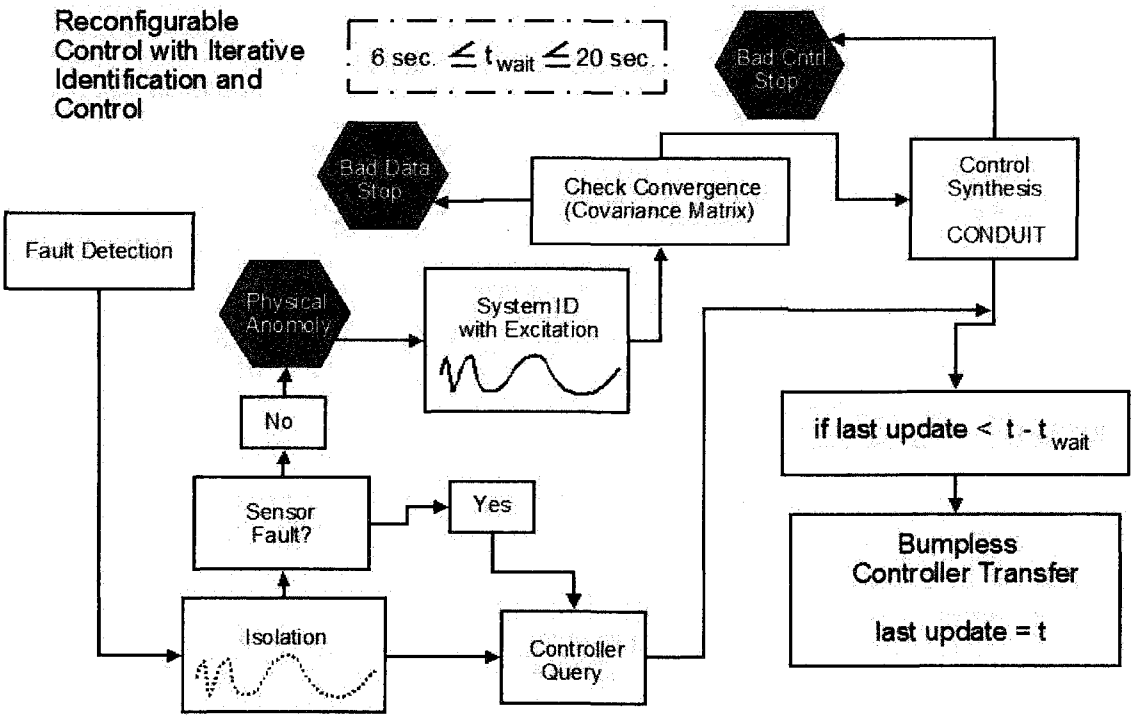


Fig. 8.12. A hybrid implementation of iterative identification and control with reconfigurable control. Note that if there is a sensor fault, the system does not perform system identification and continue with control synthesis.

Recommendations for Identification and Control:

1. Verify that system identification has converged before designing a flight control law. Least squares methods generate a covariance matrix. If this is small, then the data is probably good. Avoid using real-time system identification routines such as recursive least squares. Instead, use system identification routines with batches of flight data lasting 100 seconds for capturing fast and slow frequency dynamics and 3 to 20 seconds for fast dynamics like the short period mode.

2. Some clairvoyant knowledge is needed. One should have a good guess as to what the next flight condition will be, because the current controller you are designing will be used at the next flight condition. This is a difficulty with iterative identification and control methods which one may wish to be left unchallenged. Applying iterative identification only to precision landing and approach bypasses this issue.

3. Swap controllers using a bumpless controller transfer discussed in section 2.28.

4. A switching period of at least 12 seconds is advised. This means that one should only switch between controllers every 6 seconds. Faster switching periods can be achieved through testing.

Direct Adaptive Control with Gain Limiting

Direct adaptive control without some kind of gain limiting is unlikely to be successful, since most direct adaptive control algorithms integrate to infinite gain [1.23]. This method is impractical, although it can be made less impractical.

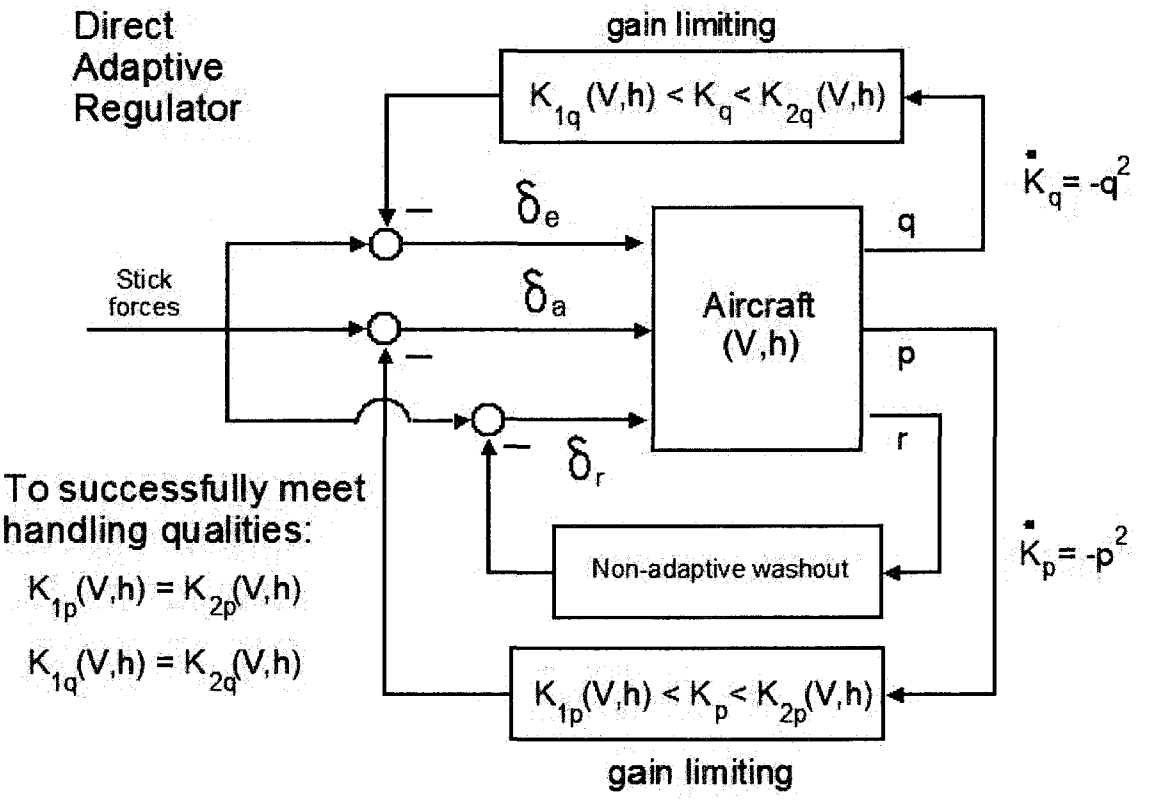


Fig. 8.13. Direct adaptive control with gain limiting can work.

Recommendations for Direct Adaptive Control:

1. Use gain limiting to prevent integrating to infinite gain.

2. Verify that gain limitation satisfies the circle criterion at every velocity, V, and altitude, h.

3. The circle criterion cannot be met for the entire lateral control system. However, one can meet the circle criterion for SISO feedback of roll rate to the aileron

4. The circle criterion only dictates stability. Stable and non-adaptive control systems can cause Pilot Induced Oscillations (PIO) and subsequently cause accidents as seen with the recent F-22 incident [3.13]. Rapid time varying gains can cause excessive overshoot and strange behavior. It is therefore best not allow large deviations in gain.

Pilot Based Adaptive Control with Gain Limiting

Allowing the pilot to change the gain has helped a pilot deal with faulty servomechanisms [5.2]. Note that this scenario involved a test pilot, Bruce Peterson, who also had training as an engineer.

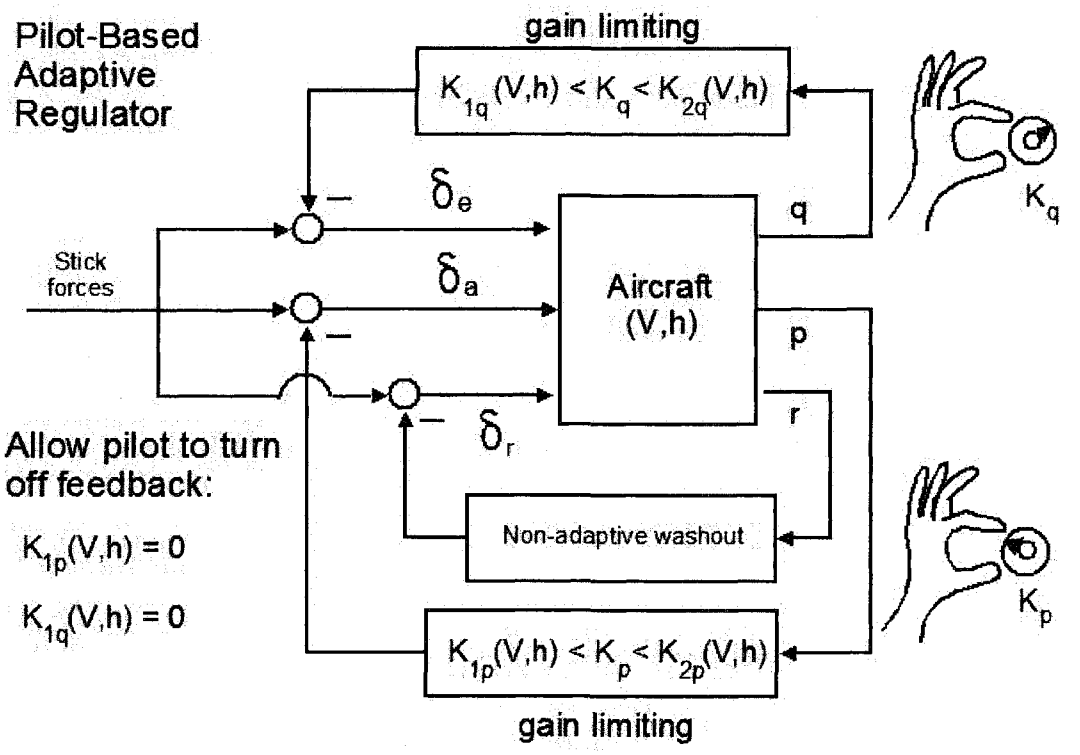


Fig. 8.14. A diagram of a pilot based adaptive control system.

Recommendations:

1. Use gain limiting to prevent integrating to infinite gain.

2. Verify that gain limitation satisfies the circle criterion at every velocity, V, and altitude, h.

3. The circle criterion cannot be met for the entire lateral control system. However, one can meet the circle criterion for SISO feedback of roll rate to the aileron

8.7 Flight Test Experience with Adaptive Control

Three versions of the X-15 hypersonic research vehicle built from 1959 through 1967: the X-15A-1, the X-15A-2, and the X-15A-3. Only the later X-15A-3 vehicle used a kind of computer driven adaptive control which was called the MH-96 adaptive control [8.2]. The earlier X-15A-1 and X-15A-2 vehicles had a regulator where the pilot could select the gains [2.21, 8.3]. No test flight with the X-15A-1 and X-15A-2 experienced a catastrophic failure. However, the X-15A-3 had a catastrophic failure in 1967 which was due to a limit cycle in the MH-96 adaptive control system; the pilot did not survive [8.4].

Until the unfortunate mishap with the MH-96 system, favorable pilot opinions were reported [8.2]. More recent attempts at direct adaptive control with an F-15 have created PIO and poor pilot opinion [8.5].

Even with stability provided by gain limiting, it is difficult to gauge performance with the rapid time varying gains used by direct adaptive control. This problem might never be solved. Perhaps this could be due to the complexity of the problem, but there is also a lack of necessity. Handling qualities metrics and control methods have evolved through nearly a century of research, development and practice based upon the assumption of linearity. Linear control methods and even some non-linear techniques like feedback linearization are easy to understand, implement and successfully improve performance. A stable adaptive system with gain limiting is mostly a complicated nuisance for performance analysis when simple and accurate linear handling qualities metrics are already available.

8.8 A Simple Fault Detection Scheme for Wing Damage

Direct adaptive control does not really make sense even if it is stable with gain limiting. Iterative identification and control needs clairvoyant awareness except for landing and approach. Reconfigurable control has limitations, but at least it makes sense and can be used at any altitude and airspeed. Reconfigurable control requires that a fault detection and isolation system be designed. This section shows a simple Fault Detection and Isolation (FDI) method for wing damage.

In section 1.2, it was stated that perfect FDI was impossible. However, nearly perfect fault detection is possible. This FDI scheme for wing damage is not nearly perfect, but it can be implemented without any new sensors.

After running several wing damage simulations, the author noticed that a non-zero aileron deflection would accompany a zero bank angle in steady flight. The tendency to reach this strange trim position can also be seen in Fig. 1.5.

However, a steady flight condition of non-zero aileron and zero bank angle is not unique to wing damage. Asymmetric fuel loading in undamaged aircraft has been known to cause such circumstances [8.6]. Additionally, it is possible to trim a normal airplane in a steady sideslip condition, where the aircraft has a nearly zero bank angle, non-zero aileron and rudder deflections and a non-zero sideslip angle. This condition can be reflected in the frequency domain by the following expression in eq. 8.25.

$$\frac{\phi}{\delta_a}(s)\delta_a(s)+\frac{\phi}{\delta_r}(s)\delta_r(s)=0 \tag{8.25}$$

Bode plots of the $\phi/\delta_a(s)$ and $\phi/\delta_r(s)$ transfer functions are shown in Fig. 8.15. For the undamaged P-17 at low frequency, $\phi/\delta_a(s) \approx c_o \, \phi/\delta_r(s)$, where $c_o = 1$. This knowledge can help reduce false positives and misdiagnosis of wing damage when the aircraft is fine.

The green stripe region in Fig. 8.16 shows steady flight conditions that would be indicative of wing damage. The red region in Fig. 8.16 is indicative of a normal sideslip flight condition.

One must be aware that the entire reasoning for this scheme is based on an assumption of a steady flight condition. If the aircraft only momentarily has its wings level with a non-zero aileron deflection, then no indication of wing damage will be given. The wings must remain level with non-zero aileron deflection for some period of time in the green stripe region before any indication of wing damage is given.

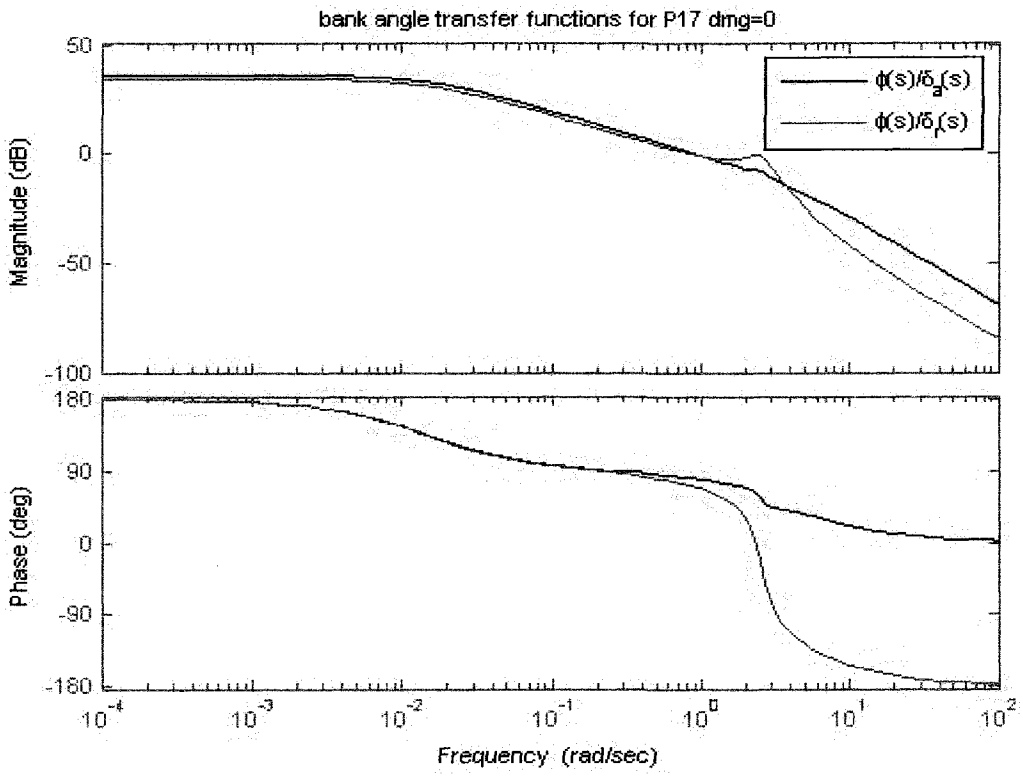


Fig. 8.15. Bode plot with bank angle as the output for the undamaged P17

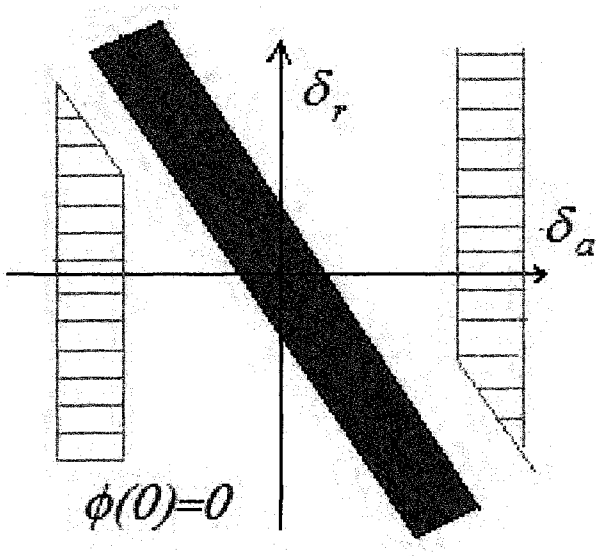


Fig. 8.16. Fault detection logic with a small bank angle will check to see if inputs are in the green region for a certain amount of time. If this is true, then an indication of damage will be given. Otherwise, no indication of damage will be given. The red region indicates a region to avoid which could lead to false positives.

A SIMULINK implementation of this detection logic is shown in Fig. 8.17. Relays are used as logic elements and a resetting integrator is reset if there is a deviation outside of the green stripe region.

In order for there to be any detection of damage, the aircraft must first be restored to a wings level flight condition. Then, a small bank angle must be maintained with a non-zero aileron deflection. During this time, the resetting integrator will integrate a constant rate as seen in Fig. 8.18. Once the output from the integrator exceeds a certain threshold, an

indication of damage can be given. This inherently implies a time delay in the detection of wing damage. Selecting a smaller threshold would give quicker fault detection. However, it is likely that more false positives would arise. Using a larger threshold would give fewer false positives and slower detection.

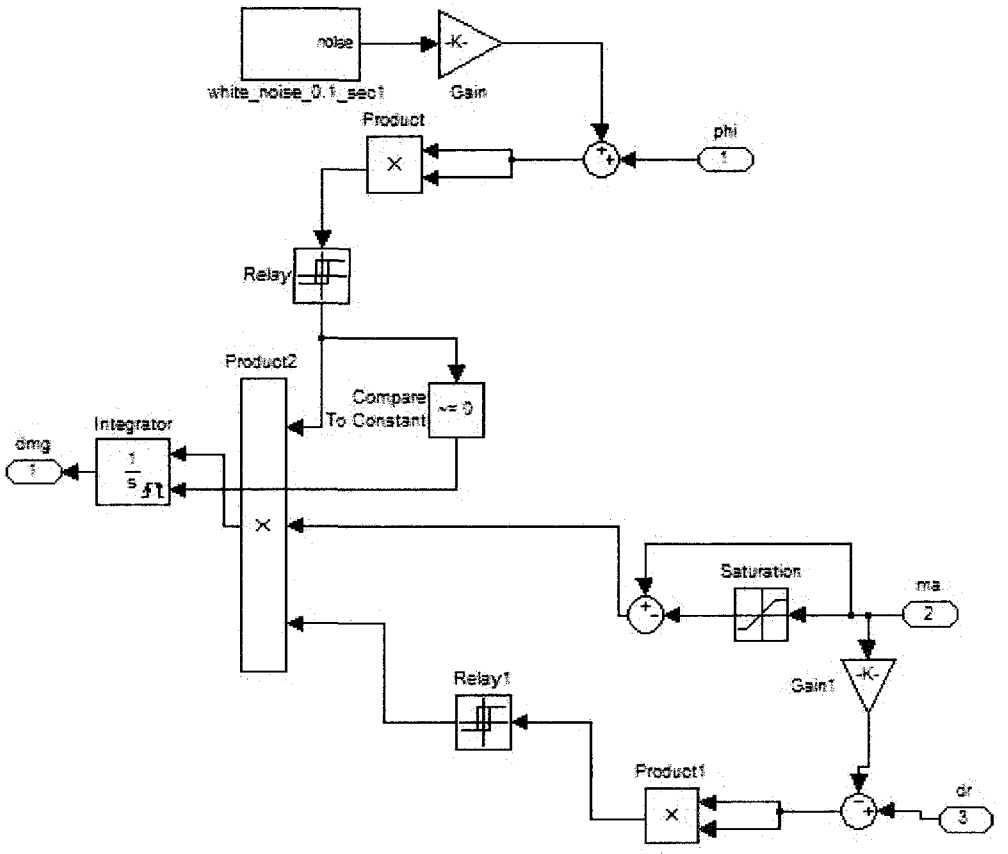


Fig. 8.17. A SIMULINK implementation of the detection logic

One might possibly improve the detection delay with an observer design [1.32]. However, observer approaches will always have drift problems which can give false positives without a washout filter. Other filtering designs merely filter inputs and outputs [1.33]. The designer must still design logic that considers what filtered error is normal and what is indicative of wing damage. Unnecessary filtering and observer design may only serve to obscure things so that they become too difficult to design logic elements which separate normal filter error from abnormal behavior.

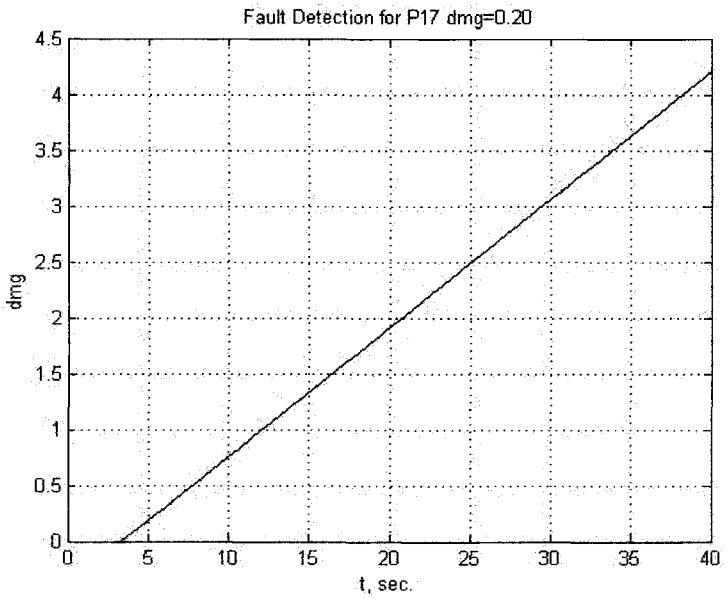


Fig. 8.18. Time response of the damage indicator, dmg, with wing damage.

A universal problem with FDI is that there can also be faults with the FDI system. FDI systems should be more reliable than the faults in question. With this scheme, the bank angle output would come from an Inertial Navigation System (INS). Modern INS instruments have Mean Time Between Failures (MTBF) on the order of tens of thousands of hours. Although INS failures are rare, they might be more frequent than wing damage. With this scheme alone, one could misdiagnosis an INS failure as wing damage.

8.9 A Reconfigurable Controller for the Wing Damaged P17

In section 4.6, it was suggested that one could reduce longitudinal to lateral cross coupling by simply increasing the roll gain. Sections 5.7 and 6.12 explored decoupling methods which use cross-feed, which could pose a threat to the undamaged aircraft in the event of a false positive for wing damage. The proposal in section 4.6 is superior to those suggested in sections 5.7 and 6.12. This is because there is a well known risk to the undamaged aircraft associated with using too much roll gain [8.7]. Misapplying longitudinal to lateral cross-feed to the undamaged aircraft produces an unknown risk.

We will combine the FDI technique from section 8.8, the bumpless controller transfer in section 2.28 and the suggestions in section 4.6 to form this complete reconfigurable controller.

The controller for the undamaged P17 is presented in eq. 8.26.

$$C_{\delta na\phi}(s)=\frac{-7.67(s+0.6)(s+3.75)}{s(s+15)} \tag{8.26}$$

Once there is an indication of wing damage, the controller in eq. 8.27 with higher roll gain will be used. The bumpless controller transfer will be used, as discussed in section 2.28.

$$C_{\delta ma\phi}(s) = \frac{-11(s+0.6)(s+3.75)}{s(s+15)} \tag{8.27}$$

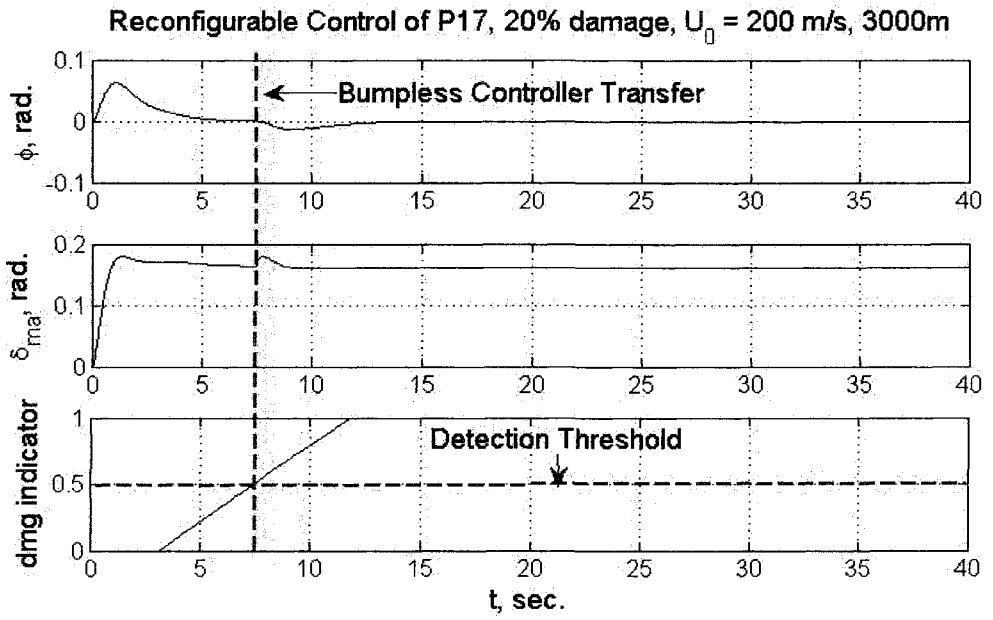


Fig. 8.19. Simulation of reconfigurable control

Results from a simulation are shown in Fig. 8.19. After roughly 7 seconds, damage has been detected and the controllers are immediately switched using the bumpless controller transfer. If a second fault was indicated at 8 or 10 seconds, it would be wise to wait another 3 or 4 seconds before switching again to avoid instability from time varying, as seen in section 8.1.

Chapter 9 – Conclusions

Small amounts of wing damage will create small problems for aircraft handling and control. Recall from section 1.3 that the International Civil Aviation Organization (ICAO) did not consider damage to wingtips or puncture holes in the wing skin to be substantial damage [1.9]. Data collected in this dissertation shows that 10-20% wing damage will not cause a significant change in aircraft handling. Large trim forces and static instability at low speed could prevent transport aircraft with small ailerons from surviving wing damage in the range of 20-30% wing loss.

Larger amounts of wing damage will create larger problems for aircraft handling and control. Awkward handling may onset after 30% wing loss and get progressively worse with increasing wing loss. Aircraft with low aspect ratio wings and powerful ailerons are expected to sustain more wing loss than aircraft with high aspect ratios and small ailerons. Given a percentage of wing loss, rolling moment imbalances will be smaller with smaller aspect ratio wings than with high aspect ratio wings. Dynamic instability may onset beyond 60% wing loss.

Although it is not likely to greatly effect survival, one may want to modify the autopilot in flight. Reconfigurable control methods with bumpless controller transfers are recommended. Direct adaptive control does not really make sense even if it is stable with gain limiting. Iterative identification and control needs clairvoyant awareness except for landing and approach. Reconfigurable control has limitations, but at least it makes sense and can be used at any altitude and airspeed.

Appendix – Selected Results of Linear Algebra

A full understanding of all the mathematics for control theory requires a great deal of knowledge. Currently, there are many ways of mathematically expressing the same idea in control theory. For example, one could describe stability of a linear system using eigenvalues, a Bode plot, or a Linear Matrix Inequality (LMI). However, this does not mean that all representations are equally useful in all cases. For instance, unstable hidden modes with close pole-zero pairs can be very hard to see on a Bode plot. By contrast, one can always check eigenvalues of the A matrix of the state space and notice instability even with unstable hidden modes.

Different mathematical representations in control theory arise because they are particularly good at expressing one idea where other representations fail. Consider state space representations and stabilizability. State space representations affirm that as long as all of the unstable modes are observable and controllable, then a stabilizing feedback controller exists. Unfortunately, this does not communicate what is *strongly* stabilizable, nor does it communicate issues with sensitivity arising from close unstable pole-zero pairs. Fundamental limitations of control [1.4-1.8] are best expressed with sensitivity integrals. Full understanding of limitations of control, sensitivity integrals and Nyquist's stability criterion would require knowledge of a mathematical topic known as complex analysis [A.1-A.3]. Most control texts only

briefly summarize results from complex analysis such as Nyquist's stability criterion without thorough discussion. A complete mathematical development of all the useful ideas and representations in control theory would require an entire textbook at least. This appendix very briefly describes some specific results of linear algebra which are used in this dissertation. More complete treatments of linear algebra can be found in Refs. [A.4-A.7].

A.1 Matrix Inversion Lemma

The matrix inversion lemma is sometimes useful in control theory and design.

$$(A+USV)^{-1} = A^{-1} - A^{-1}U\left(S^{-1}+VA^{-1}U\right)^{-1}VA^{-1} \tag{A.1}$$

Some commonly used results from the matrix inversion lemma are given below.

$$(I+AB)^{-1}A = \left(A^{-1}+B\right)^{-1} = A(I+BA)^{-1} \tag{A.2}$$

A.2 Schur Compliments

Schur compliments are useful for inverting partitioned matrices. X would be the Schur compliment of A_{11} in A. $A_{11}\in \mathbf{C}^{p\times p}$, $A_{12}\in \mathbf{C}^{p\times(n-p)}$, $A_{22}\in \mathbf{C}^{(n-p)\times(n-p)}$ and $A_{21}\in \mathbf{C}^{(n-p)\times p}$.

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}^{-1} = \begin{bmatrix} A_{11}^{-1}+A_{11}^{-1}A_{12}X^{-1}A_{21}A_{11}^{-1} & -A_{11}^{-1}A_{12}X^{-1} \\ -X^{-1}A_{21}A_{11}^{-1} & X^{-1} \end{bmatrix}, \quad X \equiv A_{22}-A_{21}A_{11}^{-1}A_{12} \tag{A.3}$$

One can also take the Schur compliment of A_{22} in A, which is labeled Y.

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}^{-1} = \begin{bmatrix} Y^{-1} & -Y^{-1}A_{12}A_{22}^{-1} \\ -A_{22}^{-1}A_{21}Y^{-1} & A_{22}^{-1}+A_{22}^{-1}A_{21}Y^{-1}A_{12}A_{22}^{-1} \end{bmatrix}, \quad Y \equiv A_{11}-A_{12}A_{22}^{-1}A_{21} \tag{A.4}$$

Schur's determinant formula can also be useful as below.

$$\boldsymbol{det}\left(\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}\right) = \boldsymbol{det}(A_{11})\boldsymbol{det}\left(A_{22}-A_{21}A_{11}^{-1}A_{12}\right) = \boldsymbol{det}(A_{22})\boldsymbol{det}\left(A_{11}-A_{12}A_{22}^{-1}A_{21}\right) \tag{A.5}$$

A.3 Sylvester's Inequality

Sylvester's inequality establishes bounds on the rank of products of matrices. We suppose that $A\in \mathbf{C}^{m\times n}$ and $B\in \mathbf{C}^{n\times p}$

$$rank[A]+rank[B]-n \le rank[AB] \le \boldsymbol{min}\{rank[A], rank[B]\} \tag{A.6}$$

This means that the rank of the product, AB, cannot be greater than the minimum rank of A or B. A rank deficient matrix multiplied by any matrix will still be a rank deficient matrix.

Consider the case where both A and B are square, $\boldsymbol{det}(AB) = \boldsymbol{det}(A)\boldsymbol{det}(B)$. Now suppose that A is rank deficient, then $\boldsymbol{det}(A) = 0$. It is therefore true that $\boldsymbol{det}(AB) = 0\boldsymbol{det}(B) = 0$ and the product AB is rank deficient.

References

[1.1] *NTSB – Most Wanted Safety Improvements* http://www.ntsb.gov/recs/mostwanted/air_ice.htm

[1.2] *NTSB Issues Icing Recommendations for Saab 340's* http://www.aviationtoday.com/regions/weur/5124.html

[1.3] *NO WING F15 – CREW STORIES – USS BENNINGTON*; http://www.uss-bennington.org/phz-nowing-f15.html, 2001.

[1.4] Seron, M., Braslavsky, J., and Goodwin, G. **Fundamental Limitations in Filtering and Control,** New York: Springer-Verlag, 1997.

[1.5] Freudenberg, J.S., Hollot, C.V., Middleton, R.H., Toochinda, V. "Fundamental Design Limitations of the General Control Configuration", *IEEE Transaction on Automatic Control,* vol. 48, no. 8, August 2003, p. 1355-1370

[1.6] Freudenberg, J.S. and Looze, D., "Right Half Plane Poles and Zeros and Design Tradeoffs in Feedback Systems", *IEEE Transaction on Automatic Control,* vol. AC-30, no. 6, June 1985, p. 555-565

[1.7] Goodwin, G.C., Salgado, M.E., and Yuz "Performance Limitations for Linear Feedback Systems in the Presence of Plant Uncertainty", *IEEE Transaction on Automatic Control,* vol. 48, no. 8, August 2003, p. 1312-1320

[1.8] Lau, K., Middleton, R.H., and Braslavsky, J.H. "Undershoot and Settling Time Tradeoffs for Non-minimum Phase Systems", *IEEE Transaction on Automatic Control,* vol. 48, no. 8, August 2003, p. 1389-1393

[1.9] *Statistical summary of Commercial Jet Transport Airplane Accidents,* http://www.boeing.com/news/techissues/pdf/statsum.pdf

[1.10] Ross, S.M., **Introduction to Probability Models,** 7th Edition, Harcourt Academic Press, 2000.

[1.11] *NTSB table*, http://www.ntsb.gov/aviation/Table5.htm

[1.12] Wong, D., Poll, S., and Krishnakumar, K., "Aircraft Fault Detection and Classification Using Multi-Level Immune Learning Detection", *AIAA Infotech@Aerospace Conference*, 26-29 September 2005, Arlington, Virginia

[1.13] *Boeing F-18 Super Hornet Homepage*,

http://www.boeing.com/defense-space/military/fa18ef/index.htm

[1.14] Kowal, B.W., Scherz, C.J., and Quinlivan, R., "C17 Flight Control System Overview", *Aerospace and Electronics Conference, 1992. NAECON 1992, Proceedings of the 1992 IEEE Natrional Congress,* Dayton, OH, May 18-22, 1992, vol. 2, p. 829-835

[1.15] *Jane's All the World's Aircraft,* http://jawa.janes.com

[1.16] Gero, D., **Aviation Disasters**, 4th Edition, Patrick-Stevens Limited, 2006

[1.17] *Boeing Investigates as Saudi 777 lands with Wing Damage* http://www.flightglobal.com/articles/2008/03/17/222268/video.html

[1.18] *2003 Baghdad DHL Attempted Shootdown Incident* http://en.wikipedia.org/wiki/Eric_Genotte

[1.19] Sarigul-Klijn, N., Nespeca, P., Marchelli, T., and Sarigul-Klijn, M., "An Approach to Predict Flight Dynamics and Stability Derivatives of Distressed Aircraft", *AIAA Atmospheric Flight Mechanics Conference and Exhibit Conference*, 18-21 August, 2008, Honolulu, Hawaii, AIAA-2008-6877

[1.20] Anderson, B.D.O., Dehgani, A., "Challenges of Adaptive Control – Past, Permanent, and Future," *Annual Reviews in Control*, vol. 32, 2008, pp. 123-135.

[1.21] Nguyen, N., Krishnakumar, K., Kaneshige, K., and Nespeca, P., "Flight Dynamics and Hybrid Adaptive Control of Damaged Aircraft", *AIAA Journal of Guidance Control and Dynamics*, 2008, vol. 31, no. 3, p. 751-764, AIAA-0731-5090

[1.22] H. P. Whitaker. "An adaptive system for control of the dynamics performances of aircraft and spacecraft" *Inst. Aeronautical Services*,1959. Paper 59-100

[1.23] Rhors, C.E., Valvani, L., Athans, M., Stein, G., "Robustness of adaptive control algorithms in the presence of unmodelled dynamics," *21st IEEE Conference on Decision and Control,* 1982, Vol. 21, p. 3-11

[1.24] Athans, M.; Valvani, L., "Adaptive control: Myths and Realities" Langley Research Center NASA Aircraft Controls Research, 1983, p 343-361 (SEE N84-20567 11-08)

[1.25] Ioannou, P.A., Sun, J., **Robust Adaptive Control,** Prentice Hall, Englewood, Cliffs, NJ, 1996. pp. 313-355.

[1.26] Narendra, K.S., and Annaswamy, A.M., "A new adaptive law for robust adaptive control without persistent excitation," *IEEE Transactions on Automatic Control;* 1987, Vol. 31, pp. 134-145

[1.27] Bobal, V., Bohm, J., Fessl, J. and Machacek, J., **Digital Self-Tuning Controllers**, Springer 2005.

[1.28] Dewey, A., Jury, E., "A note on Aizerman's conjecture," vol. AC-19, 1963, pp. 482-483.

[1.29] Albertos, P., Sala, A., Eds. **Iterative Identification and Control,** London: Springer-Verlag, 2002.

[1.30] *CONDUIT,* http://uarc.ucsc.edu/flight-control/conduit/

[1.31] Tischler, M.S., and Remple, R.K., **Aircraft and Rotorcraft System Identification**, AIAA press, 2006

[1.32] Isermann, R., **Fault-Diagnosis Systems: an Introduction from Fault Detection to Fault Tolerance,** Springer-Verlag: Berlin Heidelberg, 2006

[1.33] Campos-Delgado, D. U., and Zhou, K., "Reconfigurable fault-tolerant control using GIMC structure," *IEEE Transactions on Automatic Control*, vol. 48, no. 5, pp. 832–838, 2003.

[1.34] Vetter, T. K.; Wells, S. R.; Hess, Ronald A., "Designing for Damage – robust flight control using sliding mode techniques," NASA Contract-Grant-Task-No. NCC1-01010, Document ID: 200207785

[1.35] Spaulding, C., "Non-linear inversion control for a ducted fan UAV," M.S. Thesis, UC Davis, 2005

[1.36] Gundes, A.N., "Controllers for Systems with Sensor or Actuator Failures" *IEEE Proceedings of the 30th Conference on Decision and Control,* Brighton, England, December 1991, pp. 81-82

[2.1] Nise, N.S., **Control Systems Engineering,** John Wiley and Sons, 4th Edition, 2003.

[2.2] Dorf, R.C., Bishop, R.H., **Modern Control Systems,** Pearson Prentice Hall, 11th Edition, 2008

[2.3] Chen, C.T., **Linear System Theory and Design,** Oxford University Press, 3rd Edition, 1998

[2.4] Antsaklis, P.J., Michel, A.N., **A Linear Systems Primer,** Birkhauser: Boston, 2007

[2.5] Callier, F.M., Desoer, C.A., **Linear System Theory,** Springer-Verlag: 1991

[2.6] Chen, B.M., Lin, Z., Shamash, Y. **Linear Systems Theory: A Structural Decomposition Approach,** Birkhauser: Boston, 2004

[2.7] Williams, J.H., **Fundamentals of Applied Dynamics,** John Wiley and Sons, 1996

[2.8] Karnopp, D.C., Margolis, D.L., and Rosenburg, R.C., **System Dynamics,** John Wiley and Sons, 3rd Edition, 2000

[2.9] Kouvaritakis, B., Macfarlane, A.G.J., "Geometric Approach to Analysis and Synthesis of System Zeros – Part 1: Square Systems," *International Journal of Control*, Vol. 23, No. 2, 1976, pp. 149-166

[2.10] Skogestad, S., Postlethwaite, I., **Multivariable Feedback Control: Analysis and Design,** John Wiley and Sons, 2nd Edtion, 2005

[2.11] Youla, D.C., Jabr, H.A., Bongiorno, J.J., "Modern Wiener-Hopf Design of Optimal Controllers – Part II: The Multivariable Case," *IEEE Transactions on Automatic Control*, Vol. 21, no. 3, 1976, pp. 319-338

[2.12] Andry, et. al., "Eigenstructure Assignment for Linear Systems," *IEEE Transactions on Aerospace and Electronic Systems*, Sept. 1983, Vol. AES-19, no. 5, pp. 711-729

[2.13] Kautsky, J., Nichols, N.K., Van Dooren, P., "Robust pole assignment in linear state feedback," *International Journal of Control,* 1985, vol. 41, no. 5, pp.1129-1155

[2.14] Doyle, J.C., Stein, G., "Robustness with observers," *IEEE Transactions on Automatic Control,* vol. 24, no. 4, pp. 607-611

[2.15] Stengel, R.F., **Optimal Control and Estimation,** Dover, 1994

[2.16] Goodwin, G.C., Graebe, S.F., Salgado,M.E., **Control System Design,** Prentice Hall, 2000

[2.17] McFarlane, D.C., Glover, K., **Robust Controller Design Using Normalized Co-prime Factor Plant Descriptions,** Springer-Verlag, 1989

[2.18] Desoer, C.A., Vidyasagar, M., **Feedback Systems: Input-Output Properties,** SIAM Classics in

Applied Mathematics, 2009

[2.19] Teper, G.L., Jewell, W.T., "Aircraft Stability and Control Data," NASA CR-96008, 1969

[2.20] Tischler, M.B., **Advances in Flight Control,** Taylor and Francis, London, 1996

[2.21] Heffley, R.K., Jewell, W.T., "Aircraft Handling Qualities Data," NASA CR-2144, 1972

[2.22] Hanus, R., Kinnaert, M., and Henrotte, J., "Conditioning Technique, a General Anti-WindUp and Bumpless Transfer Method," *Automatica*, Vol. 23, No. 6, pp. 729-739

[2.23] Stein, G., "Respect the Unstable," *IEEE Control Systems Magazine*, Bode Lecture, Vol. 23, no. 4, 2003, pp. 12-25

[2.24] Subrahmanyam, M. B., "H-infinity Design of F/A-18A Automatic Carrier Landing System," *Journal of Guidance, Control, and Dynamics*, Vol. 17, no. 1, January-February 1994, pp. 187-191

[3.1] Cook, M.V., **Flight Dynamics Principles**, Butterworth-Heinemann, 2nd Edition, 2007

[3.2] McRuer, D.T., Ashkenas I.L., Graham, D., **Aircraft Dynamics and Automatic Control**, Princeton University Press, 1972

[3.3] Phillips, W.F., **Mechanics of Flight**, John Wiley and Sons, 2004

[3.4] Roskam, J., **Airplane Flight Dynamics and Automatic Flight Controls Parts 1 and 2**, DARcorporation, 2001

[3.5] Iliff, K.W., Maine, R.E., Steers, S.T., "Flight-Determined Stability and Control Coefficients of the F-111A Airplane," NASA TM-72851, 1978

[3.6] Iliff, K.W., Maine, R.E., Shafer, M.F., "Subsonic Stability and Control Derivatives for an Unpowered Remotely Piloted 3/8-scale F-15 Airplane Model Obtained from Flight Test," NASA TN D-8136, 1976

[3.7] Klein, V., Morelli, E.A., **Aircraft System Identification: Theory and Practice**, AIAA Press, 2006

[3.8] Burris, P.M., Bender, M.A., "Aircraft Load Alleviation and Mode Stabilization (LAMS): B-52 System Analysis, Synthesis and Design," AFFDL-TR-68-161, 1969

[3.9] Marchelli, T., "Damaged Aircraft Flight Dynamics, Stability, and Control: Wing Damaged C-17 Transport Aircraft Model", M.S. Thesis, UC Davis, 2004

[3.10] Stevens, B.L., Lewis, F.L., **Aircraft Control and Simulation, 2ed.,** Wiley-Interscience, 2nd Edition, 2003

[3.11] *JSBSim, An Open Source Flight Dynamics Model,* http://www.jsbsim.com/

[3.12] Wykes, J.H., Nardi, L.U., Mori, A.S., "XB-70 Structural Mode Control System Design and Performance Analysis," NASA CR-1557, 1970

[3.13] Moorhouse, D.J., et. al., "Flight Control Design–Best Practices," NATO AGARD RTO Technical Report 29

[3.14] Hess, R.A., "Unified Theory for Aircraft Handling Qualities and Adverse Aircraft-Pilot Coupling," *Journal of Guidance, Control, and Dynamics*, vol. 20, no. 6, Sept.-Oct. 1997, pp. 1141-1148

[3.15] Sura, N.K., Patel, V.V., Deodhare, G., Chetty, S., "Structural Stability Margin Criteria for Accelerated Clearance of Developmental Flights," *Proceedings of the International Conference on*

Aerospace Science and Technology, 26-28 June 2008, Bangalore, India

[3.16] Newman, B., Schmidt, D.K., "Numerical and Literal Aeroelastic-Vehicle-Model Reduction for Feedback Control Synthesis," *Journal of Guidance, Control, and Dynamics*, vol. 14, no. 5, Sept.-Oct. 1991, pp. 943-953

[3.17] Wright, J.R., Cooper, J.E., **Introduction to Aircraft Aeroelasticity and Loads**, John Wiley and Sons, 2007

[3.18] Weisshar, T.A., et. al., "Structrual Aspects of Flexible Aircraft Control," NATO AGARD RTO Meeting Proceedings 36

[3.19] Chan, S.Y., Cheng, P.Y., Myers, T.T., Klyde, D.H., Magdaleno, R.E., McRuer, D.T., "Advanced Aeroservoelastic Stabilization Techniques for Hypersonic Flight Vehicles," NASA CR-189702, 1992

[3.20] Hughes, T.J.R., **The Finite Element Method: Linear Static and Dynamic Finte Element Analysis**, Dover, 2000

[3.21] McGuire, W., Gallagher, R.H., Ziemian, R.D., **Matrix Structural Analysis**, John Wiley and Sons, 2nd Edition, 1999

[3.22] Zienkiewicz, O.C., Taylor, R.L., **The Finite Element Method for Solid and Structural Mechanics**, 6th Edition, Butterworth-Heinemann, 2005

[3.23] Gupta, K.K., Meek, J.L., **Finite Element Multidisciplinary Analysis**, AIAA Press, 2000

[3.24] Lange, R.H., Cahill, J.F., Bradley, E.S., Eudaily, R.R., Jenness, C.M., MacWilkinson, D.G., "Feasibility Study of the Transonic Biplane Concept for Transport Aircraft Application," NASA CR-132462, 1974

[3.25] Wasczak, M. R. and Schmidt, D. K., "Flight Dynamics of Aeroelastic Vehicles," *Journal of Aircraft*, Vol. 25, No. 6, 1988, pp. 563-571.

[3.26] Garrick, I.E., "Reciprocal Relations in the Theory of Nonstationary Flows," *NACA Report No. 629*, 1938

[3.27] McLean, D., **Automatic Flight Control Systems**, Prentice Hall International, 1990

[3.28] Nguyen, N., "Integrated Flight Dynamic Modeling of Flexible Aircraft with Inertial Force-Propulsion-Aeroelastic Coupling," *46th AIAA Aerospace Sciences Meeting and Exhibit*, 7-10 January 2008, Reno, Nevada, AIAA 2008-194

[3.29] Katz, J., Plotkin, A., **Low Speed Aerodynamics**, Cambridge University Press, 2001

[4.1] Gundes, A.N., Desoer, C.A., **Algebraic Theory of Linear Feedback Systems with Full and Decentralized Compensators**, Springer-Verlag, 1990

[4.2] Bristol, E., "On a new measure of interaction for multivariable process control," *IEEE Trans. Automat. Control*, AC-11, 1966, p. 133–134

[4.3] Grosdider, P., Morari, M., and Holt, B. R., "Closed-loop Properties from Steady State Gain Information," *Industrial and Engineering Chemistry Fundamentals*, vol. 24, no. 2, 1985, pp. 221-235

[4.4] Liang, Q.,"Is the relative gain array a sensitivity measure?," *IFAC Workshop on Interactions Between Process Design and Process Control*, London, UK, 1992, pp. 133-138

[4.5] Nett, C.N., and Manousiothakis, V. (1987) "Euclidean condition and block relative gain: Connections, conjectures and clarifications," *IEEE Transactions on Automatic Control*, AC-32, no. 5, pp. 405-407

[4.6] Hess, R.A., "Coupling Numerators and Input-Output Pairing in Square Control Systems," *AIAA Journal of Guidance, Navigation and Control*, vol. 26, no. 2, 2003, p. 367-369

[5.1] Linnemann, A., Maier, R., "Decoupling by Precompensation While Maintaining Stabilizability," *IEEE Transactions on Automatic Control,* vol. 38, no. 4, April 1993 pp. 629-632

[5.2] Painter, W.D., Sitterle, G.J., "HL-10 Lifting Body Flight Control System Characteristics and Operational Experience," NASA TM X-2956, 1974

[5.3] Zak, S.H., **Systems and Control**, Oxford University Press, 2002.

[6.1] Peng, Y., Kinnaert, M., "Explicit Solution to the Singular LQ Regulation Problem," *IEEE Transactions on Automatic Control,* vol. 37, 1992, pp. 633-636

[6.2] Snell, A., "Decoupling Control Design with Applications to Flight," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 4, July-August 1998, pp. 647-655

[6.3] Wang, Q.G., **Decoupling Control**, Springer-Verlag, 2003

[6.4] Wolovich, W.A., Falb, P.L., "Invariants and Canonical Forms Under Dynamic Compensation," *SIAM Journal of Control and Optimization*, vol. 14, no. 6, 1976, pp. 996-1008

[6.5] Kase, W., Mutoh, Y.,Teranishi, M., "A Simple Derivation of Interactor Matrix and its Applications," *Proceedings of the 38th Conference on Decision and Control*, Phoenix, Arizona, Dec. 1999, pp. 493-498

[6.6] Kase, W., Watanabe, T.,Yasuhiko, M., "A Simple Derivation of Interactor Matrix for Non-Square Transfer Matrices," *Society of Instrument and Control Engineers*, Osaka, Aug. 5-7, 2002, pp. 3041-3045

[6.7] Skogestad, S., Havre, K., "The Use of the RGA and Condition Number as Robustness Measures," *Computers & Chemical Engineering*, Vol. 20, 1996, Suppl. pp. S1005-S1010

[6.8] Lin, C.A., "Necessary and Sufficient Conditions for Existence of Decoupling Controllers," *IEEE Transactions on Automatic Control,* vol. 42, no. 8, Aug. 1997, pp. 1157-1161

[6.9] Blakelock, J.H., **Automatic Control of Aircraft and Missiles**, Wiley-Interscience, 2nd ed., 1991

[6.10] Courtheyn, T.L., "Multivariable Control Law Design for the X-29," M.S. Thesis, Air Force Institute of Technology, AFIT/GE/ENG/84D-21, 1985

[7.1] Meirovitch, L. and Tuzcu, I., "Integrated Approach to the Dynamics and Control of Maneuvering Flexible Aircraft," *NASA CR-2003-211748,*

[7.2] Meirovitch, L. and Tuzcu, I., "Unified Theory for the Dynamics and Control of Maneuvering Flexible Aircraft," *AIAA Journal 2004-0001-1452,* Vol. 42, No. 4, 2004, pp. 714-727.

[7.3] Dowell, E.H., **Aeroelasticity of Plates and Shells,** Noordhoof International Publishing, 1975, pp.

11-34.

[7.4]Al-Shehabi, A. G., Newman, Brett, "Aeroelastic vehicle sensor placement for feedback control applications using phase stability" , AIAA-2000-4148 AIAA Guidance, Navigation, and Control Conference and Exhibit, Denver, CO, Aug. 14-17, 2000

[7.5] Al-Shehabi, A. G., Newman, Brett, "Optimal Blending Filter Parameters and Sensor Placement for Flight Control" , AIAA-2002-4750 AIAA Guidance, Navigation, and Control Conference and Exhibit, Monterey, CA, Aug. 5-8, 2002

[7.6] Maghami, P. G., and Joshi, S.M., "Sensor/Actuator Placement for Flexible Space Structures" , *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 29, No. 2 April 1993

[7.7] Franklin, G.F., Powell, J.D., Workman, M.L., **Digital Control of Dynamic Systems**, Prentice Hall, 3rd Edition, 1997, p. 452-453

[7.8] Phillips, C.L., Nagle, H.T., **Digital Control System Analysis and Design**, Prentice Hall, 3rd Edition, 1994

[7.9] Waszak M. R., Davidson, J.B., and Schmidt D.K., "Simulation Study of the Flight Dynamics of Elastic Aircraft" , NASA CR-4102, 1987

[8.1] Haddad, W.M., Bernstein, D.S., "Explicit Construction of Quadratic Lyapunov Functions for the Small Gain, Positivity, Circle, and Popov Theorems and Their Application to Robust Stability," *IEEE Proc. of the 30th Conference on Decision and Control,* Brighton, England, Dec. 1991, pp. 2618-2623

[8.2] Staff of the Fight Research Center, "Experience with the X-15 Adaptive Flight Control System," NASA TN D-6208, 1971

[8.3] Tremant, R.A., "Operational Experiences and Characteristics of the X-15 Flight Control System," NASA TN D-1402, 1962

[8.4] Jenkins, D.R., "Hypersonics before the shuttle: A concise history of the X-15 research airplane," NASA Special Publication SP-2000-4518

[8.5] Burken, J.J., Hanson, C.E., Lee, J.A., "Flight Test Comparison of Different Adaptive Augmentations of Fault Tolerant Control Laws for a Modified F-15 Aircraft," *AIAA Infotech@Aerospace Conference, AIAA Unmanned...Unlimited Conference*, 6-9 April 2009, Seattle, AIAA 2009-2056

[8.6] Staff of the FAA "Aircraft Weight and Balance Handbook", FAA-H-8083-1A, 2007

[8.7] Chalk, C.R. "Excessive Roll Damping Can Cause Roll Ratchet," *AIAA Journal of Guidance, Navigation and Control*, Vol. 6, No. 3, 1983, pp. 218, 219.

[A.1] Needham, T., **Visual Complex Analysis**, Oxford University Press, 1997

[A.2] Silverman, R.A., **Introductory Complex Analysis**, Dover, 1984

[A.3] Freitag, E., Busam, R., **Complex Analysis**, Springer, 2nd edition, 2009

[A.4] Strang, G., **Linear Algebra and Its Applications**, Brooks Cole, 4th edition, 2005

[A.5] Kolman, B., Hill, D.R., **Introductory Linear Algebra with Applications**, Prentice Hall, 7th

Edition, 2001

[A.6] Meyer, C.D., **Matrix Analysis and Applied Linear Algebra**, SIAM press, 2001

[A.7] Bernstein, D.S., **Matrix Mathematics: Theory, Facts, and Formulas**, Princeton University Press, 2nd Edition, 2009