IOWA STATE UNIVERSITY
Digital Repository

Retrospective Theses and Dissertations

Iowa State University Capstones, Theses and Dissertations

2002

A general stochastic model for product development processes

Yi-Chiuan Lai
Iowa State University

Follow this and additional works at: https://lib.dr.iastate.edu/rtd

 Part of the Industrial Engineering Commons

Recommended Citation
Lai, Yi-Chiuan, "A general stochastic model for product development processes " (2002). *Retrospective Theses and Dissertations*. 393.
https://lib.dr.iastate.edu/rtd/393

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.

www.manaraa.com

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

A general stochastic model for product development processes

by

Yi-Chiuan Lai

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Industrial Engineering

Program of Study Committee:
John K. Jackman, Co-major Professor
Douglas D. Gemmill, Co-major Professor
Sarah M. Ryan
Stephen B. Vardeman
Ananda Weerasinghe

Iowa State University

Ames, Iowa

2002

Copyright © Yi-Chiuan Lai, 2002. All rights reserved.

UMI Number: 3051484

UMI Microform 3051484

Copyright 2002 by ProQuest Information and Learning Company.
All rights reserved. This microform edition is protected against
unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

Graduate College
Iowa State University

This is to certify that the doctoral dissertation of

Yi-Chiuan Lai

has met the dissertation requirements of Iowa State University

Signature was redacted for privacy.

Co-major Professor

Signature was redacted for privacy.

Co-major Professor

Signature was redacted for privacy.

For the Major Program

To Chin-Wen and Kuei-Mei, my parents

TABLE OF CONTENTS

LIST OF TABLES vii

LIST OF FIGURES viii

ACKNOWLEDGMENTS ix

ABSTRACT x

CHAPTER 1. INTRODUCTION 1

1.1 Motivation for the Research 3

1.2 Objectives of the Research 3

1.3 Benefits of the Research 3

1.4 Organization of This Dissertation 4

CHAPTER 2. LITERATURE REVIEW 5

2.1 Overlapping Processes 7

2.2 Concurrent Engineering 9

2.3 Axiomatic Design 11

2.4 Predictive Models for Product Development Processes 12

2.5 Other Related Papers 14

CHAPTER 3. AXIOMATIC DESIGN 15

3.1 Introduction 15

3.2 Design Axioms 16

3.3 Design Equations 17

3.4 Types of Design Matrices 18

3.5 Assumptions of Axiomatic Designs in this Research 19

CHAPTER 4. DESIGN STRUCTURE TREE GENERATING PROCESSES FOR PRODUCT DEVELOPMENT PROCESSES 20

4.1 Introduction 20

4.2 Design Structure Tree Generating Processes 21

4.3 Branching Process Resemblance 22

4.4 Generating Design Structure Trees 25

CHAPTER 5. HIERARCHICAL PROBABILISTIC MODELS FOR PRODUCT DEVELOPMENT PROCESSES 26

5.1 Introduction 26

5.2 Evolution Within a Design Activity 27

5.3 Evolution Within a Design Structure Tree 31

5.3.1 Evolution between a parent node and its child nodes 32

5.3.2 Evolution in a DST 35

CHAPTER 6. A THREE-LEVEL DESIGN PROJECT 42

6.1 Introduction 42

6.2 The Third Level of the Design Structure Tree 45

6.3 The Second Level of the Design Structure Tree 47

6.4 The First Level of the Design Structure Tree 49

6.5 Summary 50

CHAPTER 7. CONCLUSIONS 54

7.1 Summary 54

7.2 Research Contributions 55

7.3 Strengths and Limitations 56

7.4 Future Research 57

7.5 Conclusions 57

APPENDIX A. PROOF OF THEOREM 5.1 59

APPENDIX B. PROOF OF THEOREM 5.2 64

APPENDIX C. PROOF OF LEMMA 5.2.1 66

APPENDIX D. PROOF OF LEMMA 5.2.2 67

APPENDIX E. PROOF OF THEOREM 5.3 68

APPENDIX F. PROOF OF LEMMA 5.3.1 69

APPENDIX G. PROOF OF LEMMA 5.3.2 70

APPENDIX H. MATLAB PROGRAM A DESIGN TASK 71

APPENDIX I. MATLAB PROGRAM A TWO-LEVEL DST 72

APPENDIX J. MATLAB PROGRAM A THREE-LEVEL DST 73

APPENDIX K. ARENA MODEL FOR A ONE-LEVEL DST 75

APPENDIX L. ARENA MODEL FOR A TWO-LEVEL DST 76

APPENDIX M. ARENA MODEL FOR A THREE-LEVEL DST 78

REFERENCES 80

LIST OF TABLES

Table 6.1. Model parameters for the DST 43

Table 6.2. Model parameters for the SM (I) 44

Table 6.3. Model parameters for the SM (II) 45

Table 6.4. Expected times and variances for a design task from the PM 46

Table 6.5. Expected times for a design task obtained from the SM 46

Table 6.6. 95% confidence intervals for time duration of a design task from the SM 46

Table 6.7. Expected times and variances for a two-level DST from the PM 48

Table 6.8. Expected times for a two-level DST from the SM 48

Table 6.9. 95% confidence intervals for time duration of a two-level DST from the SM 48

Table 6.10. Expected times and variances for a three-level DST from the PM 49

Table 6.11. Expected times for a three-level DST from the SM 50

Table 6.12. 95% confidence intervals for time duration of a three-level DST from the SM 50

Table 6.13. Summary of error percentages 51

Table 6.14. Summary of variances 52

Table 6.15. Summary of estimate of expected times 52

Table 6.16. Comparison of expected times from the PMs 53

Table 6.17. Comparison of expected times of T and T' 53

Table 6.18. Comparison of variances of T and T' 53

LIST OF FIGURES

Figure 2.1.	Stages of a product development process	5
Figure 2.2.	Sequential, parallel, and overlapped processes	6
Figure 3.1.	Four domains of the design world	15
Figure 4.1.	The mapping of the product development process to the domain based design world	20
Figure 4.2.	An example of a zigzag procedure of FR and DP hierarchical structures	21
Figure 4.3.	An example of a branching process	23
Figure 4.4.	An example of a three-generation FR structure	24
Figure 5.1.	An example of an DP structure	26
Figure 5.2.	Plot of the $f(t)$	30
Figure 5.3.	An example of a parent-child sub-tree structure	32
Figure 5.4.	The flowchart of the HPMG algorithm	41
Figure 6.1.	A design structure tree for a design project	42
Figure 6.2.	A two-level sub-DST	47

ACKNOWLEDGMENTS

I would like to thank my major professors, Dr. John K. Jackman and Dr. Douglas D. Gemmill, for their time and efforts in assisting me with this dissertation. Their support and advice were very important for the completion of this dissertation. I would also like to thank the members of my committee, Dr. Sarah M. Ryan, Dr. Stephen Vardeman and Dr. Ananda Weerasinghe, for their valuable suggestions to this dissertation.

I would like to thank my friend, Dr. Chiang-Sheng Lee, for his helpful suggestions with proofs of theorems in this dissertation.

Most importantly, I would like to express my gratitude to my parents Chin-Wen and Kuei-Mei Lai, and my younger sisters Shu-Jiun and Shu-Rung, for their love and support during my entire study at Iowa State University.

ABSTRACT

The pressure to reduce product development time has increased due to rapidly changing technology and customer's needs. Therefore, a shortened product development time has become a critical issue for many enterprises. Being first to market is a major strategy for establishing product identity and capturing market share. Axiomatic design can be used to structure the design processes and reduce the complexity of product development. Using a hierarchical design structure tree (DST) created using design axioms, several theorems have been derived for modeling the product development process. These theorems capture successive iterations of design activities and provide a basis for a hierarchical probabilistic model-generating (HPMG) algorithm. This algorithm can be used to generate probabilistic models for every task in a DST. These models provide a mechanism to forecast the expected time window of a design project or design tasks in a project and assess the impact of design decisions. In addition, the derived theorems and the developed algorithms can also be employed to predict a general process with a hierarchical tree structure in which certain assumptions are made.

CHAPTER 1. INTRODUCTION

The pressure to reduce product development time has increased due to rapidly changing technology and customer's needs. Therefore, a shortened product development time has become a critical issue for many enterprises. The economic success of an enterprise is determined by its ability to satisfy perceived needs of customers. A successful enterprise can deliver a product that can meet all perceived needs of customers and be available in the market quickly at an acceptable price. Being first to market is a major strategy in establishing product identity and capturing market share. Product quality, product cost, development time, development cost, and development capability are commonly used as measures of success for a product (Ulrich and Eppinger 2000). Among them, the product development time is critical for enterprises to remain competitive.

Design is an information processing activity used to create an object (Kusiak 1999). The purpose of design is to convert customers' perceived needs into a solution (hardware or software) that can satisfy those needs. This solution is often called a product. Engineers or designers utilize their knowledge and creativity to find solutions to satisfy those needs. A product is eventually produced and available in the market. The evolution of the process of finding solutions is called a product development process (also known as a design process). To be more specific, the product development process is the set of steps taken to produce a final design specification from a set of perceived needs (Jackman 1998). Some researchers (Suh 1990; Simon 1996; Braha and Maimon 1997) viewed design processes as a stepwise, iterative, and evolutionary transformation processes. There are numerous creative activities

in the product development process. Therefore, the entire process is dynamic and can appear to be random.

The dynamics of a product development process is due not only to the many creative activities, but also the lack of a structure to lead an engineer or a designer to a solution. Engineers or designers work on their design tasks based on their knowledge and experience. Every engineer or designer has their own methods to find solutions and develop products since there are no general guidelines or principles to follow. Due to this reason, Suh (1990) derived two fundamental axioms to help engineers or designers generate a design structure tree (DST) to guide them through the design process. Suh (1990) developed two design axioms in an effort to define scientific principles of design. The purpose of design axioms is to reduce the complexity of design activities, evaluate conceptual designs, and insure that all requirements are satisfied. The axioms can reduce the interdependency between design activities, and establish a well-organized design project. This structure tree not only decomposes a design problem into many sub-problems, but also reduces the complexity of the problem. Such a tree is a hierarchical structure. A product development process evolves according to this hierarchical structure tree and will be more organized and predictable. A detailed explanation of axiomatic design will be provided in Chapter 3.

There are two types of tasks in a design process, namely, a design task and an integration task. Design tasks are located at the bottom level of a DST (i.e., leaf nodes). Integration tasks are parent tasks of design tasks. The objective of a design task is to determine design parameters for components of a solution while an integration task is to synthesize components created at the leaf nodes. A detailed derivation of evolutions of a design task and an integration task will be presented in Chapter 5.

1.1 Motivation for the Research

Many enterprises struggle with meeting deadlines for product development projects. Some of this struggle is due to a poor understanding of the underlying process and how their decisions affect these processes. In addition, time to market (TTM) is still a mystery to many enterprises. Therefore, a new product development model which can lead engineers to an organized design structure is needed. Furthermore, tools which possess the ability to evaluate the product development process and forecast the expected development time horizon are needed.

1.2 Objectives of the Research

The objectives of this research are to create a new product development model in the context of axiomatic design and to develop probabilistic models for a design task and an integration task. Furthermore, an algorithm will be developed to generate a hierarchical probabilistic model for an equivalent design structure tree. This stochastic model can provide a mechanism to forecast the expected duration for a design project or individual tasks in a design project, assess the effect of design decisions, and estimate associated costs for a design project.

1.3 Benefits of this Research

The expected benefits of this research include the following.

1. A new product development model
2. Mathematical foundation for evaluating product development projects
3. Understanding of the evolution of design tasks, integration tasks, and design projects

4. Method for forecasting the expected time window for a design project or individual tasks

1.4 Organization of this Dissertation

The remainder of this dissertation is organized into six chapters. In Chapter 2, previous work related to this research is reviewed. Chapter 3 introduces the fundamentals of axiomatic design. It includes descriptions of design axioms and design equations and definitions of types of design. Chapter 4 describes a design structure generating process for product development processes. A branching process to generate a design structure tree for a product development process is described. Chapter 5 presents a hierarchical probabilistic model for a product development process. This includes the derivation of nine theorems and the development of an algorithm. This is followed by an application of the theorems and the algorithm to solve a hypothetical numerical example in Chapter 6. Finally, conclusions for this research are presented in Chapter 7.

The proof of the theorems, the source codes of computer programs, and the simulation models are provided in the Appendices of this dissertation.

CHAPTER 2. LITERATURE REVIEW

A traditional product development process is a set of sequential tasks. Results of up-stream tasks will not be passed to downstream tasks until they are completed. The process is well structured but time-consuming due to multiple iterations. Since it is time-consuming, the idea of starting downstream tasks earlier (overlapping processes) and in parallel (concurrent engineering) have been popular approaches for reduce lead times. Both concurrent engineering and overlapping processes employ preliminary (i.e., incomplete) information during the development process. Figure 2.1 shows the typical stages of a product development process. The degree of overlapping between stages determines the nature of the product development process.

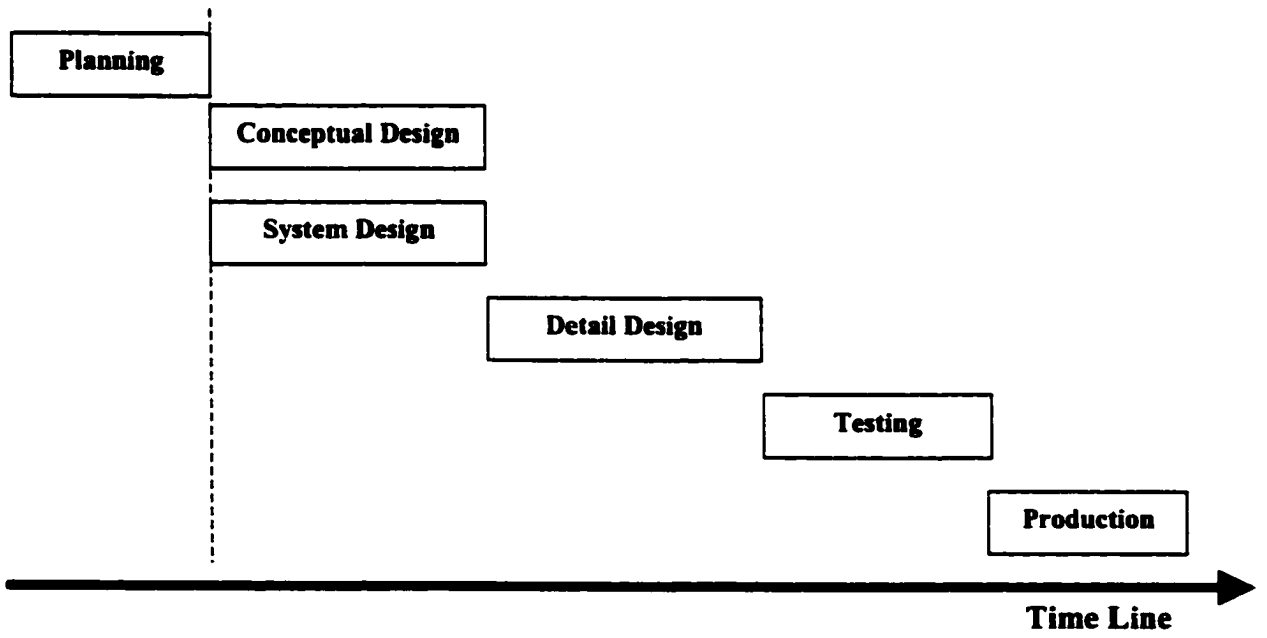


Figure 2.1. Stages of a product development process

Figure 2.2 shows three different types of product development processes (Krishnan *et al.* 1993). The downstream activity begins only when finalized information is released by the upstream activity (Figure 2.2 a) in a sequential process. Frequent information exchanges are made between the upstream and downstream activities in an overlapped process (Figure 2.2 b). No information exchange occurs between the upstream and downstream activities in a parallel process (Figure 2.2 c).

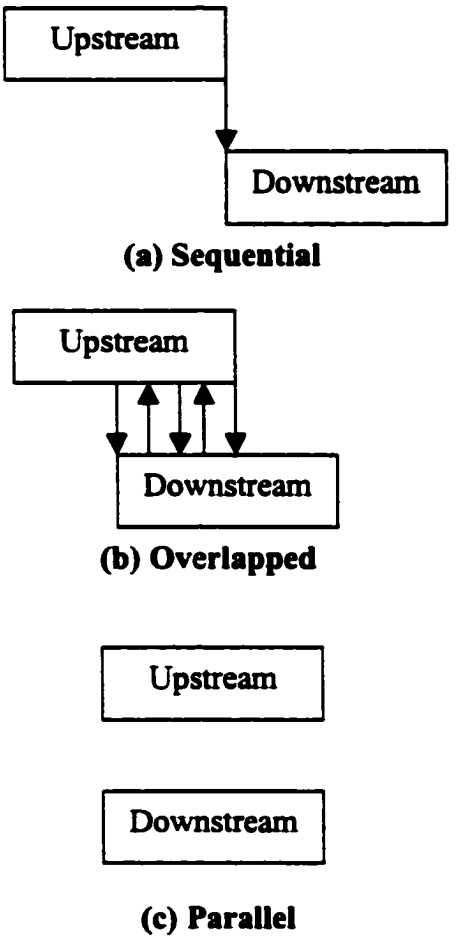


Figure 2.2. Sequential, parallel, and overlapped processes

Related research in this area can be categorized as overlapping processes, concurrent engineering, axiomatic design, predictive models for product development processes, or other related papers.

2.1 Overlapping Processes

Many researchers (Krishnan *et al.* 1995; Zirger and Hartley 1996; Krishnan *et al.* 1997) have devoted themselves to methods for accelerating the product development process. Most of them focused on starting downstream tasks as early as possible. Krishnan *et al.* (1995) proposed an iterative overlapping approach to start downstream design activities earlier by using uncertain upstream design information and adjusting design changes in subsequent iterations. The nature of the iterative overlapping model is complex due to its coupled relationship between upstream and downstream design tasks. Krishnan *et al.* (1997a) continued exploring their iterative overlapping model. They presented a model-based framework to manage the iterative overlapping model in order to identify inappropriate overlapping coupled design activities. Meanwhile, Zirger and Hartley (1996) investigated the effectiveness of techniques for accelerating product development time. Their findings showed that few of such techniques work. They also found that cross-functional design teams, which had overlapped development activities, had faster product development process. Krishnan *et al.* (1997b) considered the situation of cross-functional decision-making processes. They found inefficiencies in sequential design decision-making process, and then proposed a procedure to simplify complex design problems. Their hypothesis is that the smaller size of design project would be more receptive to simultaneous decision-making.

Without careful control of preliminary information usage, an overlapping process will cause excessive rework and increase development time (Krishnan *et al.* 1993). Krishnan *et al.* (1993) developed models of iterations and design change to help determine the timing of releasing preliminary information from the upstream design activity in order to reduce development time. Results indicated that the development time for an automobile door panel had been reduced by 27%. The disadvantage of starting downstream tasks earlier is that only preliminary information is available initially. Resolving uncertainties earlier is beneficial to overlapping processes (Terwiesch and Loch 1999). The overlapping product development process is typically tightly coupled due to the dependencies of upstream and downstream tasks. In addition, the lack of accuracy of information also causes iterations of the design process. Ahmadi *et al.* (2001) noticed this increasing-iteration phenomenon. They developed a procedure to minimize iterations between activities in order to reduce development time. In addition, an overlapping process is often costly due to requirements of additional resources (Roemer *et al.* 2000). Roemer *et al.* (2000) investigated the trade-off between product development time and costs in overlapped product development. They developed an algorithm to determine how stages should be overlapped and such overlapping strategies can provide a better method to overlap stages so that product development time can be reduced with an acceptable budget.

Several researchers (Gebala and Eppinger 1991; Christian and Seering 1995; Smith and Eppinger 1997a) have focused on analyzing the product development process in order to have a better understanding of product development processes. A design structure matrix (DSM) was first proposed by Steward (1981). The matrix representation of design shows the relationship between design tasks. Gebala and Eppinger (1991) introduced several methods

for analyzing the product development process design structure matrix. They recommended that the DSM is the best tool that allows complicated analysis to be performed. Smith and Eppinger (1997a) extended the DSM method and explored a model that utilized the DSM method to identify slow convergence of iteration within a design. Furthermore, the model can identify the design activity that requires significant iteration to obtain a solution. However, they didn't provide a mechanism to estimate the effect of communication and personnel assignments. Christian and Seering (1995) proposed a model to describe the product development process emphasizing the communication and personnel assignments. The model addresses the interdependent relationships between design activities and the requirements of communication between each individual in the design team. They also conducted a simulation to simulate design information flows between each individual based on the previous model. A prediction about design team performance would be provided by the simulation.

2.2 Concurrent Engineering

Lead time is the most important factor for a firm to be competitive (Blackburn 1991; Clark and Fujimoto 1991). The idea of concurrent engineering was developed in the 19th century (Black 1990; Black 1994; Smith 1997). The cooperation between artisans in the 19th century is similar to the modern DFM (Design For Manufacturing) practice. Concurrent engineering was highly applied and faded away during and after World War II (Ziemke and Spann 1993). It regained the popularity during the late 1970's and early 1980's (Ziemke and Spann 1993).

Concurrent engineering is a systematic approach to design products, processes, and systems simultaneously (Kusiak and Wang 1993). The idea is similar to the overlapping

process. (i.e., starting downstream tasks earlier). The only difference is that the downstream task is started at the same time as the upstream task in a concurrent engineering approach. The challenge is that only preliminary information is available at the time of starting the downstream task. Research results from Eversheim *et al.* (1997) indicated that incomplete and uncertain (preliminary) information reduced lead time more than of certain information in a sequential process. However, without careful management, product development can be degraded (Krishnan 1996). Krishnan (1996) developed a model-based framework to manage coupled phases in concurrent product development. He also offered several methods to overlap coupled phases for concurrent development. Eppinger (1991) focused on the management issues for concurrent engineering. He discussed the complexity of a design task in a concurrent development environment in his paper. Due to the coupled relationship between tasks, he suggested having a framework to evaluate if a design task should begin early in order to save time by applying concurrent engineering.

Since design of complex projects or large-scale systems often involves numerous tasks, it is not easy to control. Therefore, several researchers proposed to decompose design projects into subsystems (Kusiak and Park 1990; Kusiak and Wang 1993a). Kusiak and Park (1990) presented a methodology to decompose the design task into activities and modules. A knowledge-based system was employed for managing design activities. Kusiak and Wang (1993a) provide another systematic way to decompose a design project into subsystems. They presented a branch-and-bound algorithm to decompose design tasks. Furthermore, Kusiak and Wang (1993b) developed another algorithm in order to organize those decomposed design activities effectively.

Another characteristic of concurrent engineering is a cross-functional design team. The effort of the cross-functional design team has been emphasized by Frankenberger and Badke-Schaub (1998). They developed a model of group design processes for a design team. Teamwork effort can reduce the development time, while it also increases the complexity of communications. The communication between people and involvement of people in a design team complicate the product development process. According to the investigation of Morelli *et al.* (1995), they found that 81% of all coordination-type communications could be predicted in advance, and the prediction of frequent communications is more accurate than the prediction of infrequent communications. They suggested that an organizational design project is needed. The interdependencies between design activities require communications between team members. Decisions are reached in a meeting of the cross-functional design team by information exchanges. To reduce communication between team members, Loch and Terwiesch (1998) developed a model to determine the optimal meeting schedule based on the frequency of engineering changes.

2.3 Axiomatic Design

Whether employing sequential processes, overlapping processes, or concurrent engineering for a product development process, trial-and-error and experience is the most common method to perform design activities. There is no general framework for the design. However, there is a hierarchical nature in design (Suh 1990). That is, engineers define a problem based on customers' needs and decompose it into many sub-problems. Thus, many creative alternatives can be generated by engineers or designers. Therefore, engineers or designers will have to make many decisions during the design process. A good decision can be

made only when all other related activities provide precise information to the current activity. This is often not realistic. This is why most problems in design are due to "bad design" decisions (Jackman 1998). Therefore, it is reasonable to assume that the less interdependence between design activities then the greater likelihood of reaching a good decision. The absence of a general framework and guideline for the product development process causes many design issues. Suh (1990) defined two fundamental axioms (independence and information axioms) to guide engineers toward a good design. These two axioms help engineers structure their "thinking" and provide a method to generate a framework for a product development process.

Design axioms have been widely used in engineering design. For example, Albano and Suh (1994) presented a framework for concurrent engineering based on the concept of axiomatic design. Kim *et al.* (1991) provided a guideline for designing software based on design axioms, Black (1991) used axiomatic approach to design manufacturing systems in order to strengthen a company's ability to compete in the global world of manufacturing, Babic (1999) applied design axioms to design a modern flexible manufacturing system, and Suh (1997; 1998) applied design axioms to system design.

2.4 Predictive Models for Product Development Processes

The estimation of the duration of a design project is not an easy task due to the dynamic and uncertain nature of design. Based upon a design structure matrix (DSM) representation, which was first developed by Steward (1981), Smith and Eppinger (1997b) and Carrascosa *et al.* (1998) have developed models for estimating product development time. Smith and Eppinger (1997b) assumed that the duration for each individual design activity is deterministic,

and the design activity could be repeated with probabilities of failure in a sequential iteration. However, the absence of variability of duration could cause inaccurate predictions. Carrascosa *et al.* (1998) also used a DSM to represent the information-dependent relationships between design tasks in their prediction model. They assumed that coupled design tasks could be completed in either parallel or serial iteration. There are two quantities in the model to guide the evolution of a design task, namely, the probability of changes in design parameters and the impact of change. The impact measures possible rework due to design parameter changes. Although Carrascosa *et al.* (1998) employed a stochastic element to represent the possibility of changes in design parameters; they still assumed that the task duration is fixed unless affected by other jobs. The deterministic duration is not representative of the product development process. Another predictive model can be seen in (Ahmadi *et al.* 2001). They developed two Markov models with consideration of two different types of transition probabilities to compute the development time. The transition probabilities for the first Markov model are stationary and independent of number of iterations and the transition probabilities for the second Markov model are changing over time. In addition, they assumed the duration of an activity decreases with the number of iterations in the first Markov model while the second model addressed that the probability of additional iterations decreases with number of iterations.

In addition to mathematical models, simulation was employed by Lai and Jackman (2001). They used IDEF0 to represent a product development process. They showed how a generic simulation model could be developed using additional information that is not available in IDEF0. Their mapping process between IDEF0 and the model can be easily extended to a variety of simulation languages. Their simulation model can be used strategically to

forecast the expected time windows for a design project, identify problem areas, assess the effect of design decisions, and estimate associated costs for a design project.

2.5 Other Related Papers

Maimon and Braha (1999) and Braha and Maimon (1999) presented a mathematical theory for design in their papers. They developed a formal general design theory (FGDT), a mathematical theory of design. The purpose of FGDT is to present a domain independent modeling of design artifacts and the design process (Maimon and Braha 1999). Furthermore, FGDT provides a perspective of design practice and rules for developing a CAD system. Maimon and Braha were not alone in developing mathematics for design. Hazelrigg (1999) also attempted to develop mathematics for design. He presented eight axioms and three theorems in order to provide an axiomatic framework for engineering design. Much research has been done recently in the area of product development. Several excellent review articles have been published (Shocker and Srinivasan 1979; Finger and Dixon 1989a; Finger and Dixon 1989b; Whitney 1990; Cusumano and Nobeoka 1992; Montoya-Weiss and Calantone 1994; Brown and Eisenhardt 1995; Griffin and Hauser 1996; Balachandra and Friar 1997; Krishnan and Ulrich 2001). They provide a comprehensive review in the area of product development.

CHAPTER 3. AXIOMATIC DESIGN

3.1 Introduction

Suh *et al.* (1978) proposed a set of axioms to guide the design process. They divided design into four domains. The domain-based product development process can be seen in Figure 3.1 (Suh 1998). The customer domain is characterized by customers' perceived needs. The perceived needs are represented in terms of specified functional requirements (FRs) in the functional domain. In order to satisfy the specified FRs, corresponding design parameters (DPs) are conceived in the physical domain. Once DPs have been created, a process characterized by process variables (PVs) will be developed in the process domain.

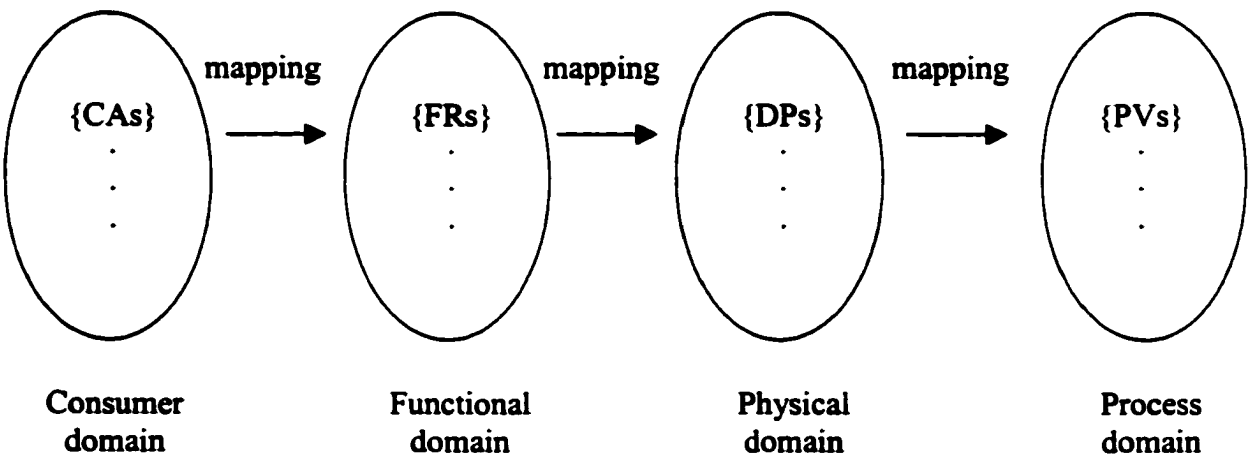


Figure 3.1. Four domains of the design world

The input to a domain characterizes "what to achieve" (what) while the output of a domain proposes "how to achieve" (how). The "how" output becomes the "what" input as we move from left to right. A design is the information content that maps "what" content to "how" content. The goal of the mapping process is to find a solution to satisfy requirements from the "what" domain. For example, {FRs} represents a characteristic vector of one level of functional requirements in the functional domain. The vector states objectives for a designer or an engineer to create a corresponding characteristic vector of solutions (design parameters), {DPs}, to satisfy the {FRs} set. A similar mapping process occurs between other domains.

3.2 Design Axioms

Suh (1990) defined two fundamental axioms to evaluate a design. These design axioms govern good designs. The two axioms are as follows:

Axiom 1 The Independence Axiom

Axiom 2 The Information Axiom

Axiom 1 states that functional requirements should be satisfied by design parameters so that FRs are not coupled. For example, if the same DPs are necessary to satisfy two FRs, then the design is problematic because changing a DP to satisfy one FR will affect another FR.

Axiom 2 states that the best design is the one that maximizes the probability of success. A design is a structured set of information content. It is possible that there can be many equally acceptable designs. One of these designs may be superior to others in terms of the probability of success in satisfying FRs (Suh 1998).

DEFINITION 3.1 Let $\{I_1, I_2, \cdots, I_m\}$ be the information content for $\{FR_1, FR_2, \cdots, FR_m\}$. According to Axiom 2, we want to minimize the information content of the design based on the Axiom 2. That is

$$Min\ I = \sum_{i=1}^{m} I_i \tag{3.1}$$

where $I_i = \log_2(1/p_i)$ and p_i is the probability of satisfying the FR_i for a given set of DPs. The problem with Axiom 2 is the difficulty in estimating p_i.

3.3 Design Equations

The mapping between FRs in the functional domain and DPs in the physical domain can be represented as a design equation. The design equation is defined as

$$\{FR\} = [A] \cdot \{DP\}, \tag{3.2}$$

where $\{FR\}$ is an $m \times 1$ functicnal requirement vector, $\{DP\}$ is an $n \times 1$ design parameter vector, and $[A]$ is an $m \times n$ design matrix. The design matrix, $[A]$, is given by

$$[A] = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{21} & \cdots & A_{2n} \\ \vdots & \vdots & & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{bmatrix}. \tag{3.3}$$

It can be seen from (3.2) that for a given FR_i,

$$FR_i = \sum_j A_{ij} DP_j\ . \tag{3.4}$$

An individual element A_{ij} is defined as

$$A_{ij} = \frac{\partial FR_i}{\partial DP_j} \tag{3.5}$$

where FR_i is the *ith* row element of $\{FR\}$, and DP_j is the *jth* row element of $\{DP\}$. Typically, the elements are defined as 0 or 1 because the exact nature of the relationship is difficult to determine.

3.4 Types of Design Matrices

Design matrices can be further classified into three different categories based on the structure of the design matrix $[A]$, namely, uncoupled designs, decoupled designs, and coupled designs. Uncoupled designs satisfy Axiom 1 completely. Coupled designs violate Axiom 1. Decoupled designs obey Axiom 1 only when design parameters are defined in a special order.

DEFINITION 3.2 Uncoupled Design

An uncoupled design matrix $[A]$ is a square matrix such that

$$A_{ij} = \begin{cases} \neq 0 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \tag{3.6}$$

(i.e., a one to one mapping between design parameters and functional requirements).

DEFINITION 3.3 Coupled Design

A coupled design matrix $[A]$ is a matrix such that at least two rows are identical.

DEFINITION 3.4 Decoupled Design

A decoupled design matrix $[A]$ is a matrix such that either upper or lower diagonal elements are all zero elements. That is,

$$A_{ij} = \begin{cases} x & \text{if } i \geq j, x = 0,1 \\ 0 & \textit{otherwise} \end{cases} \tag{3.7}$$

$$\text{or } A_{ij} = \begin{cases} x & \text{if } i \leq j, x = 0,1 \\ 0 & \textit{otherwise} \end{cases} \tag{3.8}$$

In practice, a coupled design can be decoupled by adding additional design parameters (Suh 1990).

3.5 Assumptions of Axiomatic Designs in this Research

To simplify the model in this research, it is assumed that all designs are uncoupled designs.

CHAPTER 4. DESIGN STRUCTURE TREE GENERATING PROCESSES FOR PRODUCT DEVELOPMENT PROCESSES

4.1 Introduction

Figure 4.1 shows the mapping of the product development process to the domain based design approach of axiomatic design. The major activities of product development occur between functional and physical domains. These domains encompass four phases in the product development process, namely, conceptual design, system design, detail design, and testing. The mapping process between these two domains generates tree structures that represent information content produced by an engineer or a designer in both domains. Nodes in the tree correspond to design activities in the product development process.

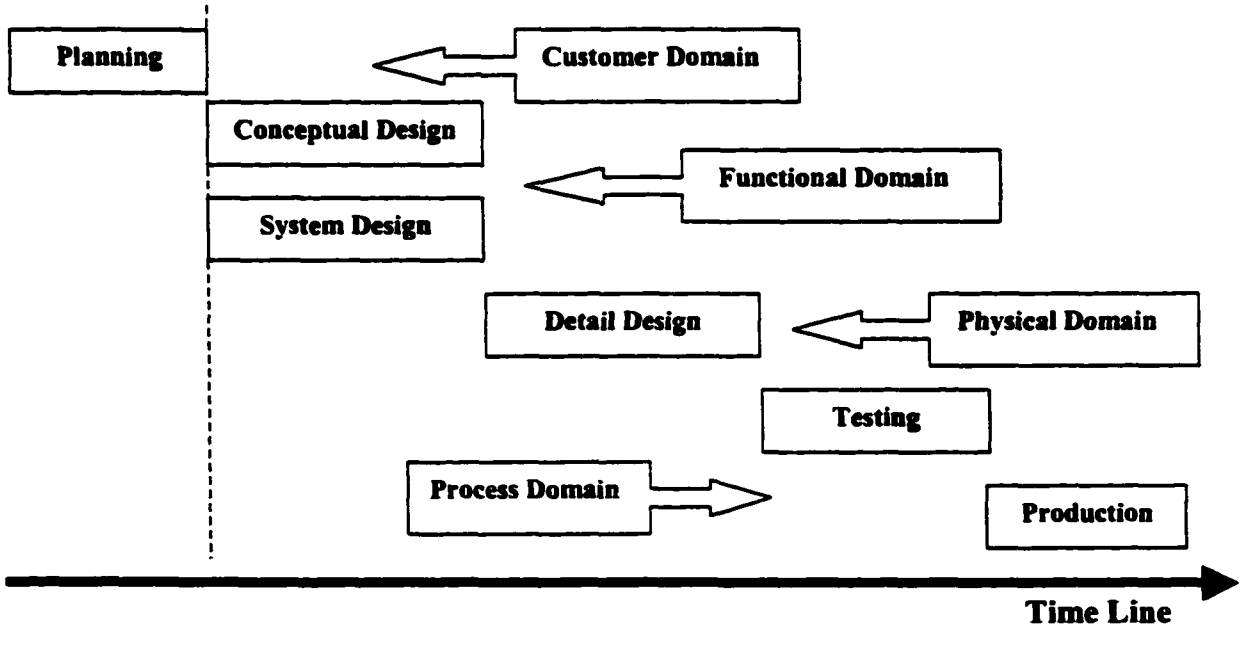


Figure 4.1. The mapping of the product development process to the domain based design world

A design structure tree (DST) is the definition of all DPs necessary to satisfy FRs. A DST is not unique as there could be multiple solutions.

4.2 Design Structure Tree Generating Processes

According to the axiomatic approach (Suh, 1990), the product development process begins with defining a set of {FRs} to satisfy a given set of needs, and ends with creating a set of {DPs} to fulfill the {FRs} set, respectively. Another set (the second level) of {FRs} can be further defined based on the {DPs} set if necessary. Then, another set of {DPs} (the second level) will be generated in order to satisfy the second level {FRs} set. The next level {FRs} set cannot be defined until the same level {DPs} set has been determined. The entire procedure is repeated until {FRs} and {DPs} have been completely defined. Therefore, {FRs} and {DPs} are inherently hierarchical in nature. Hierarchical structures of {FRs} and {DPs} can appear to be random due to multiple possible solutions for any given {FRs} and {DPs}. Figure 4.2 illustrates schematically an example of the zigzag procedure of FR and DP hierarchical structures.

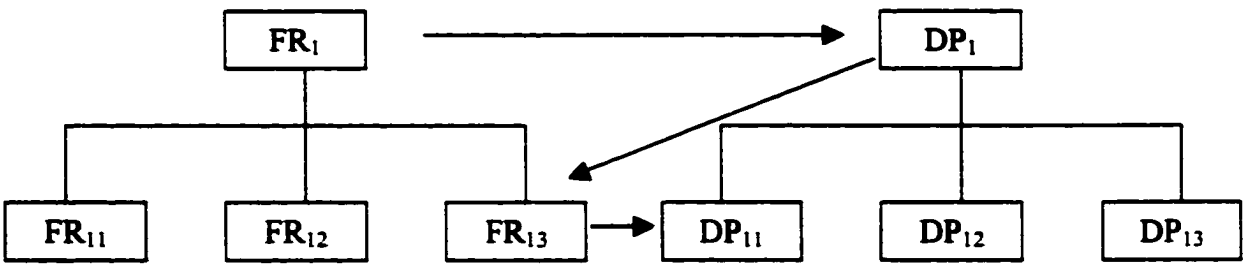


Figure 4.2. An example of a zigzag procedure of FR and DP hierarchical structures

4.3 Branching Process Resemblance

Each element of {FRs} can potentially have another set of {FRs} at the next level of the DST. Thus, the DST from axiomatic design can be viewed as a stochastic branching process. A branching process starts from an initial population (the zeroth generation) (Ross 1996). The size of the zeroth generation is denoted by X_0. Each individual in the zeroth generation will produce its own offspring independently. The offspring of the zeroth generation become the first generation of size X_1. In general, offspring of the *(n-1)th* generation become the *nth* generation and the size of the *nth* generation is denoted by X_n. The propagating process will continue until each branch of offspring eventually dies out. An example of a branching process is shown in Figure 4.3.

In a product development process, the first level of FRs is converted from customer's needs (usually $X_0 = 1$). This represents an ultimate goal of a design project. It resembles the zeroth generation in a branching process. A DP is defined in order to specify the "offspring" FRs. The FRs at the next level are generated based on the previous level of DPs. Due to the one to one mapping assumption, the generating process can be treated as a single structure rather than two identical structures. When the FRs and DPs are completely defined, conceptual design and the system design have been completed.

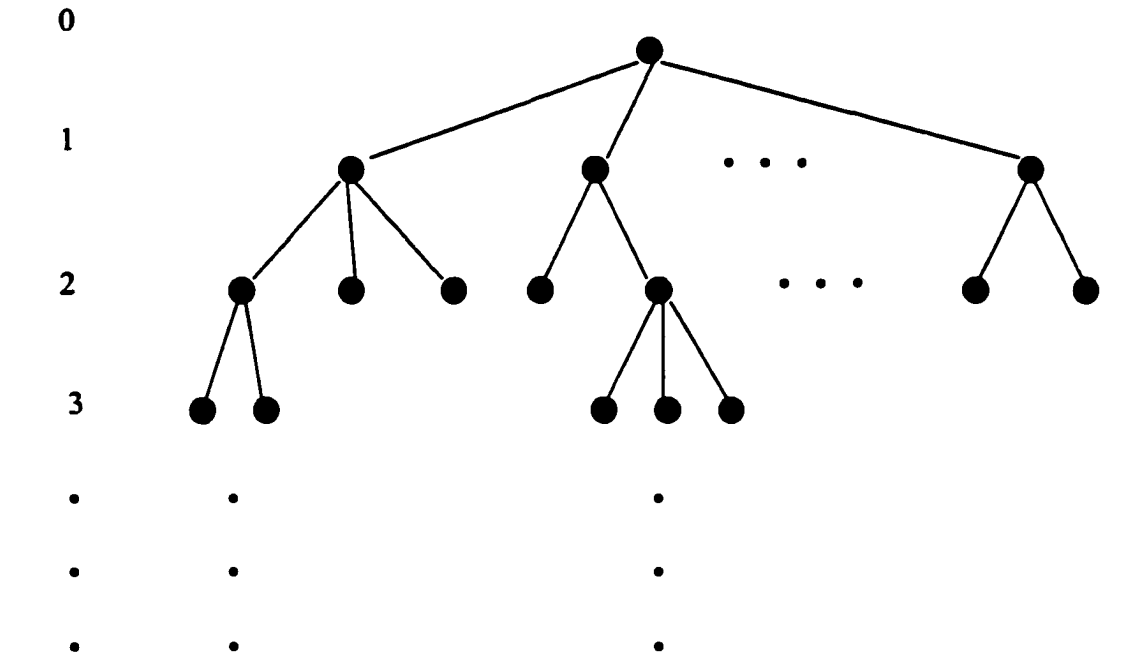


Figure 4.3. An example of a branching process

The similarities between a branching process and the FR and DP structures give rise to the use of the branching process to represent the processes. The dynamic and complex nature of the product development process causes uncertainty in the structures. Each individual in a level (generation) of the FR structure will produce j FRs for the next level (generation) with probability $P_j, j \geq 0$, independently. Assume that each individual (FR) has the same probability distribution, $P_j, j \geq 0$, for generating offspring. The number of FRs in the *nth* level of the FR structure can be calculated by

$$X_n = \sum_{i=1}^{X_{n-1}} Z_i \tag{4.1}$$

where Z_i is a random variable representing the number of FRs produced by the *ith* FR of the *(n-1)th* generation. Furthermore, the expected number of FRs in the *nth* level of the FR structure can be calculated by

$$E[X_n] = \mu^n, \tag{4.2}$$

where μ is the expected number of offspring per FR and

$$\mu = \sum_{j=0}^{\infty} jP_j \tag{4.3}$$

Eq. 4.2 is obtained by conditional expectation. By conditioning on X_{n-1},

$$E[X_n] = E[E[X_n | X_{n-1}]] = \mu E[X_{n-1}] = \cdots = \mu^n \tag{4.4}$$

The FR structure will eventually reach the bottom of the structure if FRs are completely defined at each branch (i.e., $P_0 = 1$). This means that the FR structure has been completed. Figure 4.4 shows an example of a three-generation FR structure tree.

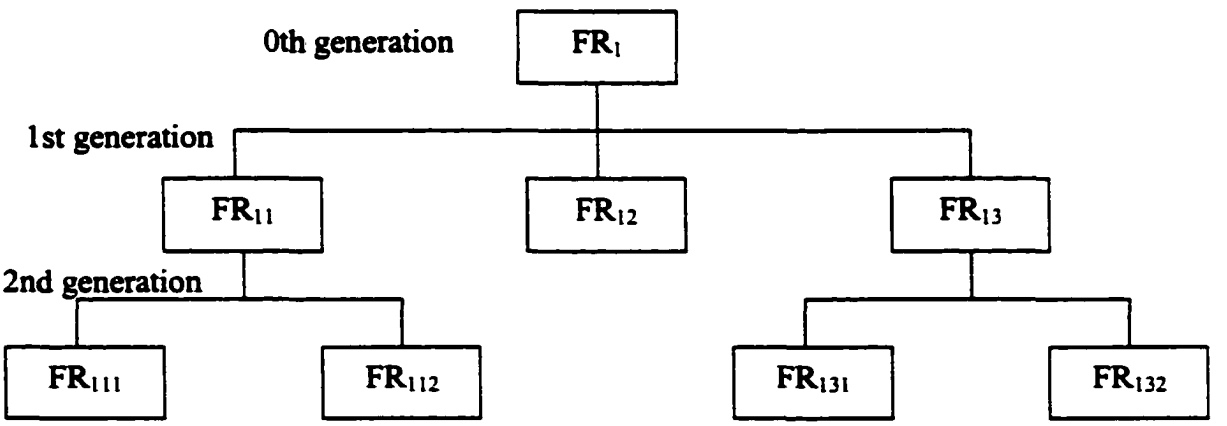


Figure 4.4. An example of a three-generation FR structure

4.4 Generating Design Structure Trees

For a given branching process, we need to know the probability distribution of P_j for each individual (FR) in order to generate a DST for a design project. Therefore, it is important to have historical data of similar design projects to estimate probabilities. With the help of this data, a DST for a design project can be generated. Using this theoretical model, a probabilistic model is derived for the DST. Using the probabilistic model, one can forecast the expected time window for a design project or individual design tasks in the DST. The method to generate a probabilistic model for a DST will be described in the next chapter.

CHAPTER 5. HIERARCHICAL PROBABILISTIC MODELS FOR PRODUCT DEVELOPMENT PROCESSES

5.1 Introduction

Design activities can be structured based on a design parameter structure tree such as the one in Figure 5.1. Each node (DP) represents a design activity, either an individual design task or an integration task. Design activities at the lowest level of a DST represent an individual design task. The remaining nodes of a DST represent integration tasks. In term of a design project timeline, design activities start from the lowest level of a DST and end at the top level of a DST. In this model, resource availability (i.e., capacitated resources) will not be considered. It is assumed that resources are allocated and available at the beginning of design activities.

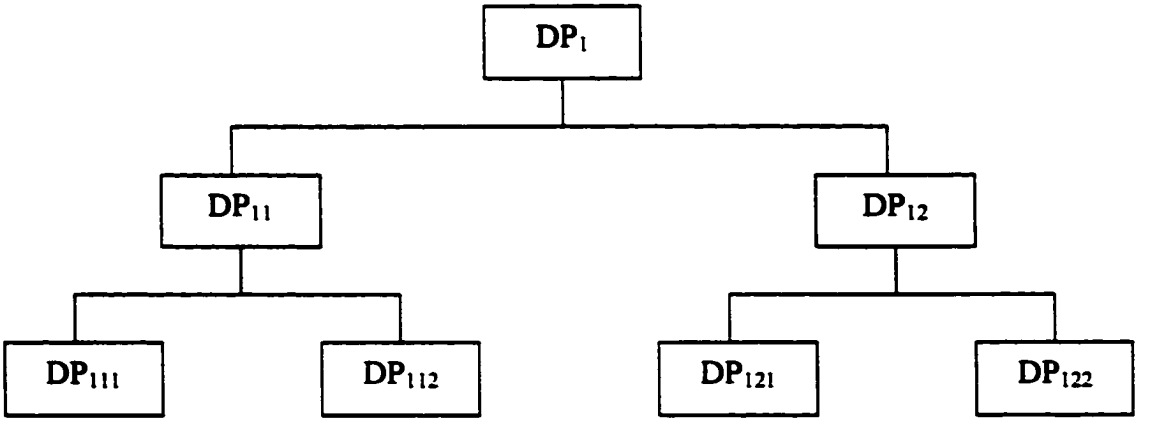


Figure 5.1. An example of an DP structure

Evolution of a product development process is a function of time. Essentially, there are two types of evolution, namely, local evolution and global evolution. The local evolution is an evolutionary process of a design task or an integration task while the global evolution is an evolutionary process between design activities. In the next section, a probabilistic model is derived for an evolutionary process within a design activity. This is followed by deriving a hierarchical probabilistic model for an evolutionary process within a DST.

5.2 Evolution Within a Design Activity

Local evolution is an evolutionary process of a design task or an integration task. The scope of a local evolution is within the design activity itself. A design activity is considered to be a series of Bernoulli trials with a different task time distribution in each trial. A design activity continues until the first success is reached. Thus, the number of trials before the first success is reached is a geometric distribution. Furthermore, it is assumed that the duration for a trial is exponentially distributed. A design activity is essentially a trial and error effort (Bernoulli trials). The next trial often depends on the outcomes of the previous trial. This can result in correlation between trials in a design activity. In order to simplify the model, the potential correlation between trials is not considered. Therefore, the duration for a design activity is a sum of a random number of exponential distributions with different means. That is $T = T_1 + T_2 + \cdots + T_N$ where T is the duration of a design activity, T_i's are the duration of the *ith* trial and success is achieved in trial N.

In order to derive the cumulative distribution function of the sum of a random number of exponential distributions with different parameters respectively, we will have to derive the

cumulative distribution function of the sum of n exponential distributions with different parameters.

Distribution for a Fixed Number of Trials

THEOREM 5.1 Independent random variables $T_1, T_2, \cdots, T_n$ have an exponential distribution with different parameters $\mu_1, \mu_2, \cdots, \mu_n$, respectively ($\mu_i \neq \mu_j$ *for all* i, j). The cumulative distribution function and the probability density function of T, the sum of all T_i's, are given by

$$F(t) = 1 - (-1)^{n+1} \sum_{i=1}^{n} \left(\prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{\mu_j}{\mu_i - \mu_j} \right) e^{-\mu_i t} \tag{5.1}$$

and

$$f(t) = (-1)^{n+1} \sum_{i=1}^{n} \mu_i \left(\prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{\mu_j}{\mu_i - \mu_j} \right) e^{-\mu_i t}, \tag{5.2}$$

for $t > 0$ and $T = T_1 + T_2 + \cdots + T_n$

Proof: See Appendix A

Furthermore, since the T_i's are mutually independent, the expected value and variance of T can be obtained by

$$E[T] = \sum_{i=1}^{n} \frac{1}{\mu_i}, \tag{5.3}$$

and $Var(T) = \sum_{i=1}^{n} \frac{1}{\mu_i^2}$. (5.4)

Distribution for a Random Number of Trials

A design activity is considered to be a series of Bernoulli trials with a different task time distribution in each trial and the number of trials is geometrically distributed. Furthermore, the number of trials and the duration of each trial are independent. Thus, the duration of a design activity is a sum of a random number of random variables. The cumulative distribution function and the probability density function of the sum of a random number of random variables can be found from the following theorem.

THEOREM 5.2 The cumulative distribution function and the probability density function of $T = T_1 + T_2 + \cdots + T_N$ is given by

$$F(t) = \sum_{n=1}^{\infty} \left[p(1-p)^{n-1} \right] \left(1 - (-1)^{n+1} \sum_{i=1}^{n} \left(\prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{\mu_j}{\mu_i - \mu_j} \right) e^{-\mu_i t} \right) \tag{5.5}$$

and

$$f(t) = \sum_{n=1}^{\infty} \left[p(1-p)^{n-1} \right] \left((-1)^{n+1} \sum_{i=1}^{n} \mu_i \left(\prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{\mu_j}{\mu_i - \mu_j} \right) e^{-\mu_i t} \right) \tag{5.6}$$

where T_i is exponentially distributed with a parameter μ_i and N is geometrically distributed with parameter p. And, T_i and N are independent.

Proof: See Appendix B

Figure 5.2 shows an example of the plot of the $f(t)$ in Theorem 5.2 with different probabilities of success.

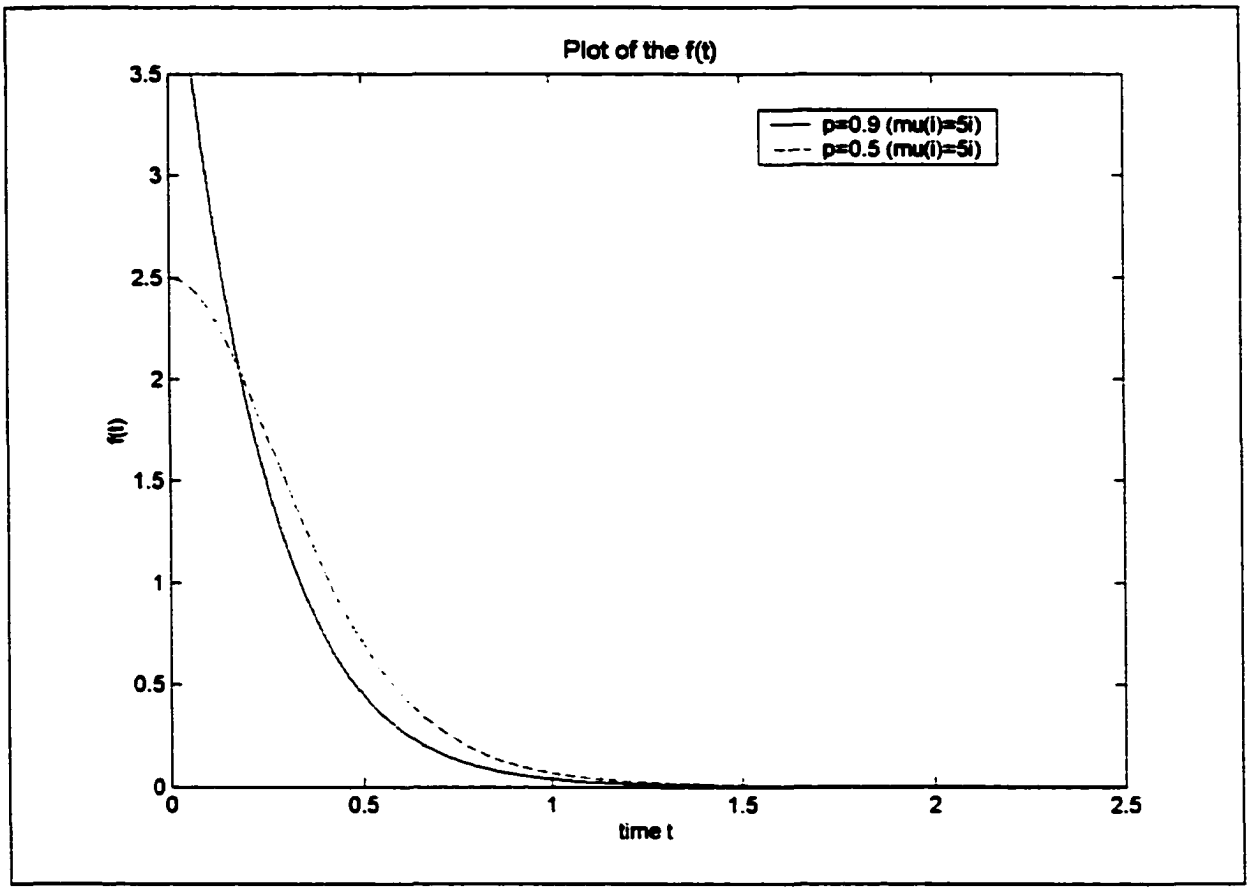


Figure 5.2. Plot of the $f(t)$

The mean and variance of the sum of a random number of exponential random variables are found by the following lemmas.

LEMMA 5.2.1 The mean of $T = T_1 + T_2 + \cdots + T_N$ is given by

$$E(t) = \sum_{n=1}^{\infty} p(1-p)^{n-1} \sum_{i=1}^{n} \frac{1}{\mu_i} \tag{5.7}$$

where T_i is exponentially distributed with a parameter μ_i and N is geometrically distributed with parameter p.

Proof: See Appendix C

LEMMA 5.2.2 The variance of $T = T_1 + T_2 + \cdots + T_N$ is given by

$$Var(t) = \sum_{n=1}^{\infty} p(1-p)^{n-1} \sum_{i=1}^{n} \frac{2}{\mu_i^2} - \sum_{n=1}^{\infty} p^2 (1-p)^{2n-2} \sum_{i=1}^{n} \frac{1}{\mu_i^2} \tag{5.8}$$

where T_i is exponentially distributed with a parameter μ_i and N is geometrically distributed with parameter p.

Proof: See Appendix D

Based on Theorems 5.1 and 5.2, an expected time window for a design task can be found for given values of p and μ_i. Furthermore, the impact of changes of design parameters (μ_i's) or resource skill levels (reflected in p and μ_i) on the duration of a design task can also be estimated by those theorems.

5.3 Evolution Within a Design Structure Tree

The global evolution is a stepwise evolutionary process. That is the process moves up one level at a time. An upper level design activity starts when all lower level design activities are completed. The entire process stops when design activities at the top level are

completed. It is also assumed that the task at the upper level (i.e., parent node) is also a Bernoulli trial without consideration of correlation between trials. Even though causes of failures at the upper level might come from lower level tasks, it is assumed that the problem will be resolved at the upper level.

In this section, a hierarchical probabilistic model is derived for an evolutionary process within a DST.

5.3.1 Evolution between a parent node and its child nodes

Evolution of a product development process resembles an assembly process. An assembly process can consist of multiple subassembly processes. Similarly, a product development process also has multiple sub-design processes. Essentially, design is a synthesis process. Engineers or designers "assemble" design parameters according to the DST. Therefore, a DST can be partitioned into multiple subassembly processes. A partition is a parent-child sub-tree as shown in Figure 5.3. According to the first design axiom and the assumption of uncoupled designs, design tasks between nodes at the same level are mutually independent. Thus, each node in the DST is an independent activity.

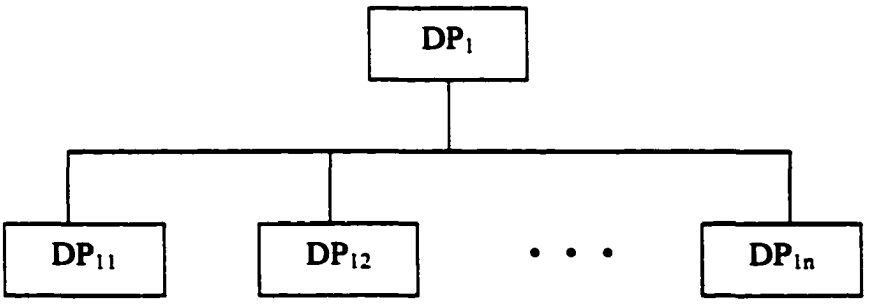


Figure 5.3. An example of a parent-child sub-tree structure

A design task at the parent-node level is essentially an integration task. An integration task is also a trial and error process. The integrating process will not stop until the first success is reached. There are numerous possibilities for the failure of an integration task. It might involve going back to the child-node level for modification. In order to reduce the complexity of our model, it is assumed that modifications only occur at the parent level in order to integrate child nodes successfully. The correlation between trials is not considered.

The distribution function of a child node is given by Theorem 5.2. An integration task at the parent node will only be initiated when all design tasks of its child nodes are completed. Thus, the duration T of a parent-child sub-tree 1 is given by

$$T = Max\{T_{11}, T_{12}, \cdots, T_{1n}\} + T_1 \tag{5.9}$$

where the distribution functions of $T_{1j}, j = 1,2,\cdots,n$, are defined by Theorem 5.2 and T_1 also follows the same distribution defined in Theorem 5.2. The distribution function of (5.9) is defined by the following theorem.

THEOREM 5.3 Given n child nodes with time durations represented as independent random variables $T_{11}, T_{12}, \cdots, T_{1n}$, and a parent node, T_1, having a distribution as in THEOREM 5.2, the cumulative distribution function of T is given by

$$F(t) = \int_0^t F_{11}(t-t_1)F_{12}(t-t_1)\cdots F_{1n}(t-t_1)f_1(t_1)dt_1 \tag{5.10}$$

where $T = Max\{T_{11}, T_{12}, \cdots, T_{1n}\} + T_1$, and $f_1(t)$ and $F_{1i}(s)$ are the probability density function and the cumulative distribution function of T_1 and T_{1i}, respectively.

Proof: See Appendix E

The expected value and variance of $T = Max\{T_{11}, T_{12}, \cdots, T_{1n}\} + T_1$ are determined from Lemmas 5.3.1 and 5.3.2 (also see (Feldman and Valdez-Flores 1996)).

LEMMA 5.3.1 The mean of $T = Max\{T_{11}, T_{12}, \cdots, T_{1n}\} + T_1$ is given by

$$E[T] = \int_0^\infty (1 - F(t))dt \tag{5.11}$$

where $F(t)$ is the cumulative distribution function of $T = Max\{T_{11}, T_{12}, \cdots, T_{1n}\} + T_1$.

Proof: See Appendix F

LEMMA 5.3.2 The variance of $T = Max\{T_{11}, T_{12}, \cdots, T_{1n}\} + T_1$ is given by

$$Var[T] = 2\int_0^\infty \int_t^\infty (1 - F(x))dxdt - \left(\int_0^\infty (1 - F(t))dt\right)^2 \tag{5.12}$$

where $F(t)$ is the cumulative distribution function of $T = Max\{T_{11}, T_{12}, \cdots, T_{1n}\} + T_1$.

Proof: See Appendix G

Based on Theorem 5.3, an expected time window for an integration task can be found for given values of p and μ_i. Furthermore, the impact of changes of design parameters (μ_i) or resource skill levels (p and μ_i) on the duration of a design task can also be estimated by the theorem.

5.3.2 Evolution in a DST

In the previous section, a probabilistic model for the evolution within a parent-child sub-tree was developed. For a DST, a parent node could also be a child node of its upper level node. Thus, the evolution of a parent-child sub-tree begins again. As mentioned previously, it is assumed that failures do not result in revisiting child nodes again. Therefore, the procedure is a non-descending process. In this section, a hierarchical probabilistic model-generating (HPMG) algorithm will be presented.

HPMG algorithm

The evolution of a product development process moves toward the top level of a DST. The probabilistic models for design activities at each level are obtained from Theorems 5.2 and 5.3. Lemmas 5.3.1 and 5.3.2 are used to find their means and variances, respectively. The notations for the algorithm and the algorithm itself can be summarized as follows:

Notation

D_{ij}^{l}: Expected duration of child j of parent i at the lowest level (l) of a DST

V_{ij}^{l}: Variance of child j of parent i at the lowest level (l) of a DST

F_{ij}^{l}: Cumulative distribution function (c.d.f.) of child j of parent i at the lowest level (l) of a DST

f_{ij}^{l}: Probability density function (p.d.f.) of child j of parent i at the lowest level (l) of a DST

$D_{ij}^{(k)}$: Expected duration of child j of parent i at the level k of a DST

$V_{ij}^{(k)}$: Variance of child j of parent i at the level k of a DST

$F_{ij}^{(k)}$: Cumulative distribution function (c.d.f.) of child j of parent i at the level k of a DST

$f_{ij}^{(k)}$: Probability density function (p.d.f.) of child j of parent i at the level k of a DST

$f_i^{(k)}$: Probability density function (p.d.f.) of parent i at the level k of a DST

N: Number of levels of a DST ($k = 1$: the top level of a DST; $k = N$: the lowest level of a DST)

$N^{(k)}$: Number of nodes at level k of a DST

$N_i^{(k)}$: Number of child nodes of node i at level k of a DST ($i = 1,2,\cdots,N_k$)

p_{ij}^{l}: Probability of success of child j of parent i at the lowest level (l) of a DST

$p_{ij}^{(k)}$: Probability of success of child j of parent i at the level k of a DST

μ: Mean of an exponential distribution (There is no special number system for μ in order to reduce the complexity of notations. Each node has its own set of μ_j, and μ_j's can be different.)

The purpose of the HPMG algorithm is to utilize theorems developed previously to generate probabilistic models for every task in a DST. The procedure of the algorithm is as follows:

1. Probabilistic models for design tasks at the lowest level of a DST

The design process starts with design tasks at the lowest level of a DST. Since Theorem 5.2 and Lemmas 5.2.1 and 5.2.2 are developed for a design task, they are employed to generate probabilistic models for design tasks at the lowest level of a DST.

2. Probabilistic models for integration tasks at the second lowest level of a DST

It becomes an integration task once the design activity moves up to the parent level. The probabilistic model for design tasks at the second lowest level of a DST can be obtained based on Theorems 5.2 and 5.3 and their lemmas.

3. Probabilistic models for integration tasks at other levels of a DST

Probabilistic models for integration tasks at other levels of a DST can be obtained based on Theorem 5.3 and its lemmas. The cumulative distribution function of the task at the current level of a DST can only be obtained when the cumulative distribution functions of tasks at previous level are known. The cumulative distribution function from Theorem 5.3 is used in the parent node. Repeating this procedure, probabilistic models for integration tasks at other levels of a DST can be obtained.

A detailed HPMG algorithm is presented as follows:

Algorithm

Step 1: Generate probabilistic models and calculate means and variances for the lowest level of a DST

Set $k = N$

For $i = 1$ *to* $N^{(k-1)}$

For $j = 1$ *to* $N_i^{(k-1)}$

$$F_{ij}^{l}(t) = \sum_{n=1}^{\infty}\left[p_{ij}^{l}\left(1-p_{ij}^{l}\right)^{n-1}\right]\cdot\left(1-(-1)^{n+1}\sum_{l=1}^{n}\left(\prod_{\substack{m=1\\ m\neq l}}^{n}\frac{\mu_m}{\mu_l-\mu_m}\right)e^{-\mu_l t}\right)$$

$$f_{ij}^{l}(t) = \sum_{n=1}^{\infty}\left[p_{ij}^{l}\left(1-p_{ij}^{l}\right)^{n-1}\right]\cdot\left((-1)^{n+1}\sum_{l=1}^{n}\mu_l\left(\prod_{\substack{m=1\\ m\neq l}}^{n}\frac{\mu_m}{\mu_l-\mu_m}\right)e^{-\mu_l t}\right)$$

$$D_{ij}^{l} = \sum_{n=1}^{\infty} p_{ij}^{l}(1-p_{ij}^{l})^{n-1} \sum_{m=1}^{n} \frac{1}{\mu_m}$$

$$V_{ij}^{l} = \sum_{n=1}^{\infty} p_{ij}^{l}(1-p_{ij}^{l})^{n-1} \sum_{m=1}^{n} \frac{2}{\mu_m^2} - \sum_{n=1}^{\infty} (p_{ij}^{l})^2 (1-p_{ij}^{l})^{2n-2} \sum_{m=1}^{n} \frac{1}{\mu_m^2}$$

Next j

Next i

Step 2: Generate probabilistic models and calculate means and variances for the (N-1)th level of a DST

Set $k = N-1$

For $i = 1$ *to* $N^{(N-2)}$

For $j = 1$ *to* $N_i^{(N-2)}$

$$f_{ij}^{(N-1)}(t) = \sum_{n=1}^{\infty} \left[p_{ij}^{(N-1)} \left(1-p_{ij}^{(N-1)}\right)^{n-1} \right] \cdot \left((-1)^{n+1} \sum_{l=1}^{n} \mu_l \left(\prod_{\substack{m=1 \\ m \neq l}}^{n} \frac{\mu_m}{\mu_l - \mu_m} \right) e^{-\mu_l t} \right)$$

$$F_{ij}^{(N-1)}(t) = \int F_{ij}^{l}(t-t_1) F_{ij}^{l}(t-t_1) \cdots F_{ij}^{l}(t-t_1) f_{ij}^{(N-1)}(t_1) dt_1$$

$$D_{ij}^{(N-1)} = \int_0^{\infty} \left(1 - F_{ij}^{(N-1)}(t)\right) dt$$

$$V_{ij}^{(N-1)} = 2 \int_0^{\infty} \int_0^{\infty} \left(1 - F_{ij}^{(N-1)}(x)\right) dx dt - \left(\int_0^{\infty} \left(1 - F_{ij}^{(N-1)}(t)\right) dt \right)^2$$

Next j

Next i

Step 3:

Set $k = N-2$

Step 4: Generate probabilistic models and calculate means and variances for the *kth* level of a DST

For $i = 1$ *to* $N^{(k-1)}$

For $j = 1$ *to* $N_i^{(k-1)}$

$$f_{ij}^{(k)}(t) = \sum_{n=1}^{\infty}\left[p_{ij}^{(k)}\left(1-p_{ij}^{(k)}\right)^{n-1}\right]*\left((-1)^{n+1}\sum_{l=1}^{n}\mu_l\left(\prod_{\substack{m=1\\m\neq l}}^{n}\frac{\mu_m}{\mu_l-\mu_m}\right)e^{-\mu_l t}\right)$$

$$F_{ij}^{(k)}(t) = \int_0 F_{ij}^{(k+1)}(t-t_1)F_{ij}^{(k+1)}(t-t_1)\cdots F_{ij}^{(k+1)}(t-t_1)f_{ij}^{(k)}(t_1)dt_1$$

$$D_{ij}^{(k)} = \int_0^{\infty}\left(1-F_{ij}^{(k)}(t)\right)dt$$

$$V_{ij}^{(k)} = 2\int_0^{\infty}\int^{\infty}\left(1-F_{ij}^{(k)}(x)\right)dxdt-\left(\int_0^{\infty}\left(1-F_{ij}^{(k)}(t)\right)dt\right)^2$$

Next j

Next i

Step 5:

Decrease k *by 1*

If $k = 0$, *then STOP;*

otherwise, go to Step 4

Figure 5.4 shows the flowchart of the HPMG algorithm. The algorithm generates probabilistic models for every node at each level of a DST. Some parameters must be known

before using this algorithm, namely, probabilities of success for each node in a DST, and means of exponential distributions for each trial in a design activity.

The procedure of the HPMG algorithm is exactly like a product development process. The process moves up one level at a time. The probabilistic model of an upper level task can be found only if the probabilistic models of lower level tasks are known.

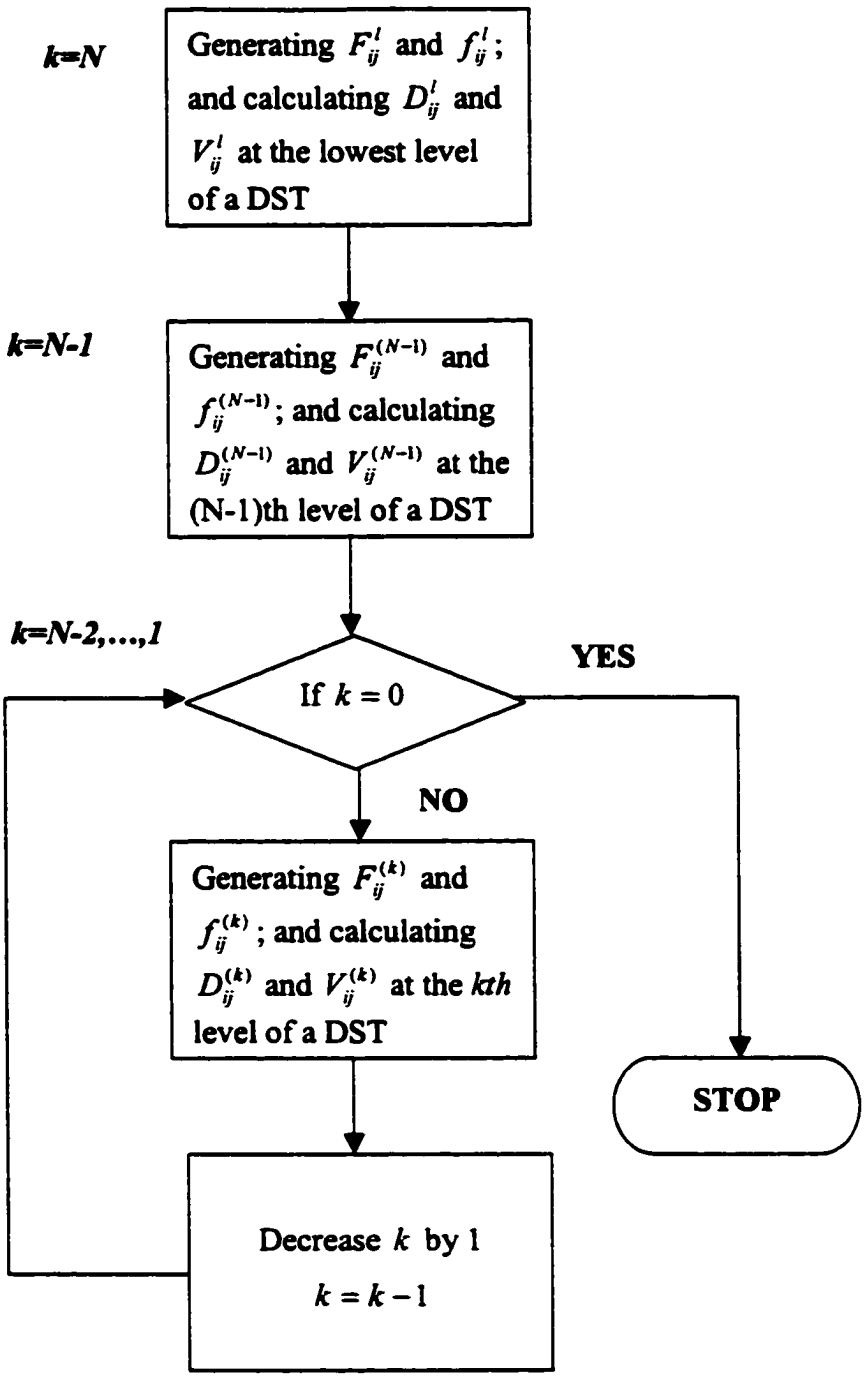


Figure 5.4. The flowchart of the HPMG algorithm

CHAPTER 6. A THREE-LEVEL DESIGN PROJECT

6.1 Introduction

With a design structure tree (DST) and proper model parameters (probability of success of a design task and mean process time of a design task), we can apply the theorems and the hierarchical probabilistic model-generating (HPMG) algorithm described in Chapter 5 to obtain expected time windows and probabilistic models for design tasks. In this section, a hypothetical numerical example is used to apply those theorems and the HPMG algorithm to calculate expected time windows and generate probabilistic models for design tasks.

For this example, a three-level DST is used for a design project as shown in Figure 6.1. As mentioned previously, a design project starts with design tasks at the lowest level of a DST. Therefore, the project begins with simultaneously designing DP_{111}, DP_{112}, DP_{121}, and DP_{122} at the lowest level of the DST since they are mutually independent. DP_{11} can't be started until both DP_{111} and DP_{112} are completed. The same situation applies to DP_{12}. Once DP_{11} and DP_{12} are completed, we can start the DP_1 task.

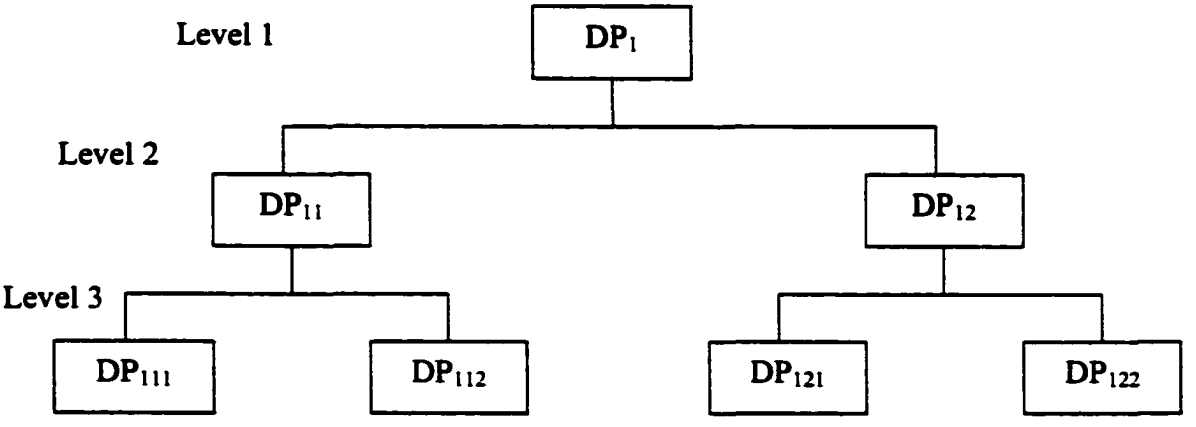


Figure 6.1. A design structure tree for a design project

Model parameters for the DST are shown in Table 6.1. Three different probabilities at the lowest level of the DST were used to represent three different risk levels (high, medium, and low) of a design project at the early stage. Two different risk levels (medium and low) were used for the intermediate stage of a design project. One risk level (low) for the closing stage of a design project was used. The computations of high risk at the second level of the DST and high and medium risks at the first level of the DST could not be completed when using the Vincent Farm, a high performance computing system at Iowa State University. In addition, the duration of each trial in a design task is exponentially distributed with different parameters shown in Table 6.1. Parameters of exponential distributions for trials in a design task are increasing because of the consideration of learning effect.

Table 6.1. Model parameters for the DST

Level of the DST	Probability of success of a design task	Parameters of exponential distributions for trials in a design task
3	0.2, 0.5, 0.9	5(1)[a], 10(2), 15(3), 20(4), 25(5), ..., etc.
2	0.5, 0.9	5(1), 10(2), 15(3), 20(4), 25(5), ..., etc.
1	0.9	5(1), 10(2), 15(3), 20(4), 25(5), ..., etc.

a: mean of an exponential distribution, number in the bracket stands for the *ith* trial

In addition to applying our theorems and algorithm to the numerical example, a simulation model was used to simulate design tasks in the example DST and compare the results between our probabilistic model (PM) and the simulation model (SM). Furthermore, sensitivity analysis was also performed by using a simulation model to simulate the same DST with different time distributions (deterministic duration and the beta distribution) for trials in a design task to determine how changes in a task time distribution affect expected time win-

dows for design tasks. Due to the complexity of computation, MATLAB (Hanselman *et al.* 2000) was utilized for the computation of the probabilistic model. In addition, Arena (Kelton *et al.* 2001) was employed for the simulation. Tables 6.2 and 6.3 show model parameters for the simulation model. In order to make the means of the beta distributions equivalent to the means of exponential distributions, we used $\frac{2}{\mu_i} f(x)$ as the task time distribution for the *ith* trial where $f(x)$ is a beta distribution with parameters α and β (α=2, and β=2) and μ_i is the mean of the exponential distribution for the *ith* trial. Then, the mean of $\frac{2}{\mu_i} f(x)$ will be $\frac{1}{\mu_i}$, which is the same as the mean of the exponential distribution. However, the variance of $\frac{2}{\mu_i} f(x)$ is different from the variance of the exponential distribution ($\frac{1}{\mu_i^2}$), which is $\frac{1}{5\mu_i^2}$[1].

Table 6.2. Model parameters for the SM (I)

Level of the DST	Probability of success of a design task	Parameters of beta distributions for trials in a design task
3	0.2, 0.5, 0.9	5(1)[b], 10(2), 15(3), 20(4), 25(5), ..., etc.
2	0.5, 0.9	5(1), 10(2), 15(3), 20(4), 25(5), ..., etc.
1	0.9	5(1), 10(2), 15(3), 20(4), 25(5), ..., etc.

b: number in the bracket stands for the value of μ_i for the *ith* trial

[1] $Var\left[\left(\frac{2}{\mu_i}\right) f(x)\right] = \left(\frac{2}{\mu_i}\right)^2 Var[f(x)] = \frac{4}{\mu_i^2} \cdot \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)} = \frac{1}{5\mu_i^2}$

where $Var[f(x)]$ is the variance of the beta distribution with parameters $\alpha = 2$ and $\beta = 2$.

Table 6.3. Model parameters for the SM (II)

Level of the DST	Probability of success of a design task	Deterministic time for trials in a design task
3	0.2, 0.5, 0.9	5(1)[c], 10(2), 15(3), 20(4), 25(5), ..., etc.
2	0.5, 0.9	5(1), 10(2), 15(3), 20(4), 25(5), ..., etc.
1	0.9	5(1), 10(2), 15(3), 20(4), 25(5), ..., etc.

c: deterministic duration, number in the bracket stands for the *ith* trial

In the following sections, the results for each level are shown separately. Since the same time duration distribution and probability of success were used for every design task in the DST, one design task was calculated at each level of the DST only, namely, DP_{111}, DP_{11}, and DP_1. Thus, the probabilistic models of DP_{112}, DP_{121}, and DP_{122} are the same as DP_{111} while the probabilistic model of DP_{12} is the same as DP_{11}.

6.2 The Third Level of the Design Structure Tree

In the first step of the HPMG algorithm, we generate probabilistic models and calculate means and variances for design tasks at the lowest level of a DST. The probabilistic model of the design task at the lowest level of the DST can be obtained based on Theorem 5.2. Also, Lemmas 5.2.1 and 5.2.2 were used to obtain the mean and variance of the model.

Results shown in Table 6.4 were obtained from MATLAB. A MATLAB program for one-level DST can be found in Appendix H.

Table 6.4. Expected times and variances for a design task from the PM

Probability of success	Expected duration	Variance
0.2	0.3946	0.0989
0.5	0.2773	0.0584
0.9	0.2107	0.0431

A series of simulations (10000 replications) were implemented. The simulation model for a design task can be found in Appendix K. Results of the simulation for different time distributions are shown in Tables 6.5 and 6.6.

Table 6.5. Expected times for a design task obtained from the SM

Prob. of success	Exponential	Beta	Deterministic
0.2	0.3970 (0.61%)*	0.3989 (1.09%)	0.3998 (1.32%)
0.5	0.2746 (-1.11%)	0.2759 (-0.51%)	0.2761 (-0.43%)
0.9	0.2090 (-0.82%)	0.2096 (-0.53%)	0.2111 (0.18%)

*: Numbers in the parentheses stand for percent deviations from expected times obtained from PM

Table 6.6. 95% confidence intervals for time duration of a design task from the SM

Prob. of success	Exponential	Beta	Deterministic
0.2	(0.3916, 0.4023)	**(0.3954, 0.4025)***	**(0.3969, 0.4027)**
0.5	(0.2697, 0.2788)	(0.2733, 0.2785)	(0.2743, 0.2779)
0.9	(0.2049, 0.2130)	(0.2076, 0.2115)	(0.2104, 0.2118)

*: Intervals in bold indicate that expected times do not fall into the intervals

As would be expected, the results obtained from the simulation models indicate reasonable agreement with the probabilistic model for the mean, because the mean will not change for different task distributions except for the high-risk task. However, one would expect difference in the variance. Table 6.6 shows 95% confidence intervals obtained from the

simulation model for the exponential, beta, and deterministic time distributions. It shows the overlap of the intervals, indicating no significant differences in the results.

6.3 The Second Level of the Design Structure Tree

When all design tasks under the same parent node are completed (DP_{111} and DP_{112} in our example), the design activity will move to the parent level (DP_{11} in our example). A sub-DST (see Figure 6.2) in our example is partitioned from the DST. According to the HPMG algorithm, the probabilistic model of DP_{11} can be obtained based on Theorem 5.3. Lemmas 5.3.1 and 5.3.2 were employed to calculate the mean and variance of the design task. The probabilistic model of DP_{12} is the same as the probabilistic model of DP_{11} due to the same sub-DST and parameters.

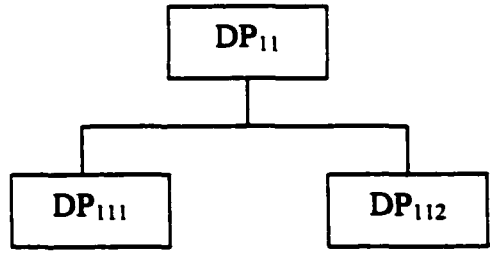


Figure 6.2. A two-level sub-DST

The results shown in Table 6.7 were obtained from MATLAB. A MATLAB program for a two-level DST can be found in Appendix I.

Table 6.7. Expected times and variances for a two-level DST from the PM

Probability of success	Expected duration	Variance
0.5	0.6773	0.1277
0.9	0.5250	0.0970

A series of simulations (10000 replications) were implemented for the two-level DST. The simulation model for a two-level DST can be found in Appendix L. Results of the simulation for different time distributions are shown in Tables 6.8 and 6.9.

Table 6.8. Expected times for a two-level DST from the SM

Prob. of success	Exponential	Beta	Deterministic
0.5	0.6711 (-0.91%)*	0.6292 (-7.11%)	0.6025 (-11.04%)
0.9	0.5217 (-0.05%)	0.4779 (-8.46%)	0.4306 (-17.52%)

*: Numbers in the parentheses stand for percent deviations from expected times obtained from PM

Table 6.9. 95% confidence intervals for time duration of a two-level DST from the SM

Prob. of success	Exponential	Beta	Deterministic
0.5	(0.6645, 0.6777)	**(0.6256, 0.6326)***	**(0.6000, 0.6051)**
0.9	(0.5158, 0.5277)	**(0.4753, 0.4804)**	**(0.4295, 0.4316)**

*: Intervals in bold indicate that expected times do not fall into the intervals

The results indicate disagreement with the probabilistic model when the assumptions are violated by using beta and deterministic values. However, the expected times obtained from the simulation model are within ten percent deviations from the expected times obtained from the probabilistic model. On the other hand, the expected times deviate away from the expected times obtained from the probabilistic model when the assumption is vio-

lated by using deterministic values. The sensitivity to other distributions remains to be investigated.

Table 6.9 shows 95% confidence intervals obtained for the simulation model of the exponential, beta, and deterministic time distribution. The expected time obtained from the probabilistic model falls into the confidence interval predicted by the simulation model only when the exponential time distribution assumption is not violated.

6.4 The First Level of the Design Structure Tree

The final stage of the design project in our example is DP_1 which integrates DP_{11} and DP_{12}. This represents the overall project lead time. Based on the HPMG algorithm, Theorem 5.3, and Lemmas 5.3.1 and 5.3.2 were used to generate the probabilistic model for DP_1 and calculate the expected time and the variance of DP_1, respectively.

Results shown in Table 6.10 were obtained from MATLAB. A MATLAB program for a three-level DST can be found in Appendix J.

Table 6.10. Expected times and variances for a three-level DST from the PM

Probability of success	Expected duration	Variance
0.9	0.9009	0.1463

A series of simulations (10000 replications) were implemented for the three-level DST. The simulation model for a three-level DST can be found in Appendix M. Results of the simulation for different time distributions are shown in Tables 6.11 and 6.12.

Table 6.11. Expected times for a three-level DST from the SM

Prob. of success	Exponential	Beta	Deterministic
0.9	0.8940 (-0.76%)*	0.7593 (-15.72%)	0.6647 (-26.21%)

*: Numbers in the parentheses stand for percent deviations from expected times obtained from PM

Table 6.12. 95% confidence intervals for time duration of a three-level DST from the SM

Prob. of success	Exponential	Beta	Deterministic
0.9	(0.8868, 0.9012)	**(0.7564, 0.7622)***	**(0.6633, 0.6661)**

*: Intervals in bold indicate that expected times do not fall into the intervals

The results indicate disagreement with the probabilistic model when the assumptions are violated by using beta and deterministic values. In addition, the expected times deviate away from the expected times obtained from the probabilistic model when the assumptions are violated by using a beta distribution and deterministic values. The sensitivity to other distributions remains to be investigated.

Table 6.12 shows 95% confidence intervals obtained for the simulation model of the exponential, beta, and deterministic time distribution. The expected time obtained from the probabilistic model falls within the confidence interval predicted by the simulation model only when the exponential time distribution assumption is not violated.

6.5 Summary

Summaries of error percentages, estimate of variances, estimate of expected times for design tasks at three levels in the DST are shown in Tables 6.13, 6.14, and 6.15. As discussed in previous sections, results of the simulation models for exponential time distributions indicate agreement with the probabilistic models. In addition, our probabilistic model could also predict expected time windows for design activities within about 10% error when

the duration of a trial in a design task is a beta distribution. There is a clear trend that the error percentage increases as the number of levels increases. However, when the assumption of the time distribution is violated and the time duration is a deterministic value, the results of the simulation models indicate agreement with the probabilistic models only for design tasks at the lowest level of the DST with medium and low risk levels. For a design task at the lowest level of the tree-level DST, the difference between the simulation model with deterministic time durations and the probabilistic model is not significant.

Table 6.13. Summary of error percentages

	POS*								
	0.2			0.5			0.9		
Level	EXP*	BET*	DET*	EXP	BET	DET	EXP	BET	DET
3	0.61#	1.09	1.32	1.11	0.51	0.43	0.82	0.53	0.18
2	N/A	N/A	N/A	0.91	7.11	11.0	0.05	8.5	17.5
1	N/A	N/A	N/A	N/A	N/A	N/A	0.76	15.7	26.2

*: POS: Probability of success; EXP: exponential; BET: beta; DET: deterministic.
#: Percent deviation from PM results.

Table 6.14 shows that estimates of variance for the duration of design activities with beta and deterministic trial time distributions are smaller than for exponential time distributions. Consider two similar distributions with the same means and different variances ($\sigma_1 < \sigma_2$). The probability of getting a large number, M, from distribution 1 is smaller than for distribution 2. The distribution with the large variance tends to get a larger maximum or smaller minimum task time in terms of probabilities. In our case, the distribution of design activity duration with exponential task times represents distribution 2, and the distribution of design activity duration with beta or deterministic task times represents distribution 1. In ad-

dition, the variances of the distributions of design activities at the parent level (Equation 5.9) increase as the number of levels increases. The parent level distribution is based on the maximum value from the child nodes. As seen in Table 5.15, the differences in results between the distributions occurs at the parent levels. This is attributed to the larger variance of the exponential distribution having a higher probability of generating a larger maximum.

Table 6.14. Summary of estimate of variances

	POS*								
	0.2			0.5			0.9		
Level	EXP*	BET*	DET*	EXP	BET	DET	EXP	BET	DET
3	0.09	0.03	0.02	0.05	0.02	0.008	0.04	0.01	0.001
2	N/A	N/A	N/A	0.13	0.03	0.01	0.09	0.02	0.003
1	N/A	N/A	N/A	N/A	N/A	N/A	0.14	0.02	0.005

*: POS: Probability of success; EXP: exponential; BET: beta; DET: deterministic.

Table 6.15. Summary of estimate of expected times

	POS*								
	0.2			0.5			0.9		
Level	EXP*	BET*	DET*	EXP	BET	DET	EXP	BET	DET
3	0.3970	0.3989	0.3998	0.2746	0.2759	0.2761	0.2090	0.2096	0.2111
2	N/A	N/A	N/A	0.6711	0.6292	0.6025	0.5217	0.4779	0.4306
1	N/A	N/A	N/A	N/A	N/A	N/A	0.8940	0.7593	0.6647

*: POS: Probability of success; EXP: exponential; BET: beta; DET: deterministic.

Since the parameters are the same for every node in the DST, one might expect that the expected time at the integration level is equal to the summation of the expected times of tasks at the child level. However, the results do not indicate otherwise (see Table 6.16).

Table 6.16. Comparison of expected times from the PMs

Level \ POS*	0.9	0.5
3	0.2107	0.2773
2	0.5250	0.6779
1	0.9009	N/A

*: POS: Probability of success.

Furthermore, Table 6.17 shows the comparison of expected times of an individual design task (T') and the expected time of the maximum finish time of two child tasks (T). It shows that T has a larger expected time than T'. Table 6.18 shows the comparison of variance of T and T'. Applying the same concept described previously to this case, the distribution with a larger variance tends to obtain a larger expected value. The expected value of the distribution of the maximum finish time of two child tasks is greater than of an individual task. This explains why the expected time of an integration task is greater than the summation of expected times of its sub-tasks at the child level.

Table 6.17. Comparison of expected times of T and T'

POS*	0.9	0.9	0.5	0.5
Level	T	T'	T	T'
2	0.3142	0.2773	0.4	0.2773
1	0.6902	0.5250	N/A	N/A

*: POS: Probability of success.

Table 6.18. Comparison of variance estimates for T and T'

POS*	0.9	0.9	0.5	0.5
Level	T	T'	T	T'
2	0.05	0.04	0.07	0.05
1	0.1	0.09	N/A	N/A

*: POS: Probability of success.

CHAPTER 7. CONCLUSIONS

7.1 Summary

The goal of this research is to develop a general stochastic model for product development processes. Based upon axiomatic design, a design structure tree (DST) can be used to structure the product development process. The DST generating process resembles a branching process. Using historical data from previous DSTs, an expected DST can be obtained through a branching process. Once the DST of a design project is created, a probabilistic model for the DST can be developed.

In this research, several theorems and an algorithm were developed to generate probabilistic models for a design project and individual design tasks. Theorem 5.1 serves as a fundamental base for other theorems. It defines the cumulative distribution function and the probability density function of the sum of n potentially different exponential distributions. The mean and the variance of probability functions defined in Theorem 5.1 were also provided. A design task is an iterative series of Bernoulli trials. The number of trials until a success is a random variable with a geometric distribution. Theorem 5.2 defines the cumulative distribution function and the probability density function of the sum of a random number (N) of exponential distributions while Lemmas 5.2.1 and 5.2.2 define the mean and the variance of the probability function in Theorem 5.2.

Theorems 5.1 and 5.2 define a general probabilistic model for a design task. Knowing the general probabilistic model for a design task, we can further develop a general probabilistic model for a design task at the parent level of a DST. A design activity at any level of the DST other than the lowest level of the DST is essentially an integration task. The cu-

mulative distribution function of an integration task is defined in Theorem 5.3. The cumulative distribution function of a design task at parent levels of the DST can be obtained based on its probability density function and the cumulative distribution functions of the design tasks of child nodes. In addition, Lemmas 5.3.1 and 5.3.2 define the mean and the variance of the probability function in Theorem 5.3, respectively.

Theorems 5.3, and Lemmas 5.3.1 and 5.3.2 can be used for the probabilistic model of every parent node in a DST. A hierarchical probabilistic model-generating (HPMG) algorithm was developed. The HPMG algorithm is a recursive procedure for generating a probabilistic model for a design task at every level of a DST. The HPMG algorithm is a stepwise process since the probabilistic model of a design task at the parent level of a DST can only be obtained if probabilistic models for all design tasks at its child level of the DST are known.

In another aspect of this research, a hypothetical example was used to demonstrate the application of theorems and the HPMG algorithm. A simulation model was developed to show the sensitivity of the general probabilistic model to violation of underlying assumptions. Results show that the probabilistic model agreed well with the simulation model.

7.2 Research Contributions

The key contributions of this research can be summarized as follows:

1. The first contribution of this research is the development of the general probabilistic model for the evolution of a design task in the product development process.
2. The second contribution of this research is the development of the general probabilistic model for the evolution of an integration task in the product development process.

3. The third significant contribution of this research is the hierarchical probabilistic model-generating algorithm for generating probabilistic models for every design activity in a design structure tree.

With general probabilistic models, we can

1. Forecast the expected time window for a design project or individual tasks in a design project using the mean and variance.
2. Assess the effect of a design decision such as the introduction of new technology or adding or removing design parameters.
3. Provide a tool for evaluating product development project.

7.3 Strengths and Limitations

The significant strengths of this research model are summarized as follows.

1. The general probabilistic model can predict the expected time window for a design project or individual tasks precisely. This is supported by results obtained from the simulation model.
2. With violation of the exponential time distribution assumption, the model appears to perform well in some circumstances.
3. The HPMG algorithm can generate a general probabilistic model for any tree structure process if there is no dependency between nodes in the tree.

The limitations include the following.

1. The general probabilistic model can only be used for an uncoupled design or an uncoupled tree structure process.
2. Historical data for previous DSTs are needed.
3. The resources have to be unlimited.

4. Modifications can only occur at the parent level once the child nodes are completed.
5. The computation time for a large DST or high-risk design tasks (low probability of success) at parent level may be unacceptable.

7.4 Future Research

The model assumptions can be addressed as candidates for future modifications or extensions. Consideration can be given in the following areas.

1. The assumption of the exponential time distribution for a trial in a design task could be switched to other probability distribution functions. Another commonly used probability distribution is the normal distribution. Theorems 5.1 and 5.2 need to be modified in this case.
2. A decoupled design is also acceptable in the design world. A further development of a general stochastic model for a decoupled design scenario is another possible extension. Theorem 5.3 needs to be revised to reflect dependence between tasks.
3. Other than complicating the current model, simplification could reduce the complexity of the current model. Replacing the exponential distribution in Theorem 5.1 with the Erlang distribution, for example, is a possible modification. Theorem 5.2 needs to be revised in this case.

7.5 Conclusions

Being first to market is a major strategy in establishing product identity and capturing market share. The timing of product introduction is a critical success factor for enterprises. Therefore, a mechanism that has the ability of foreseeing the expected time window for a de-

sign project is important to enterprises. A general stochastic model for product development processes was proposed and developed for this purpose. A hierarchical probabilistic model-generating algorithm was also developed in order to generate probabilistic models for design tasks in a hierarchical design process. The performance of this model has been demonstrated through an example of a design project in this dissertation. In addition, the sensitivity of this model was also shown in this dissertation. Results indicate that the variance of the duration of a design activity may be more important than the probability of success for a design activity in reducing the duration for a design activity. Further investigation needs to be done in this area. In short, the stochastic model can provide a mechanism to forecast the expected time window for a design project or individual tasks in a design project and assess the impact of a design decision. It is believed that the model has the potential to serve as a basis for a useful tool for design project management and control.

APPENDIX A. PROOF OF THEOREM 5.1

THEOREM 5.1 Independent random variables $T_1, T_2, \cdots, T_n$ have an exponential distribution with different parameters $\mu_1, \mu_2, \cdots, \mu_n$, respectively ($\mu_i \neq \mu_j$ *for all i, j*). The cumulative distribution function and the probability density function of T, the sum of all T_i's, are given by

$$F(t) = 1 - (-1)^{n+1} \sum_{i=1}^{n} \left(\prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{\mu_j}{\mu_i - \mu_j} \right) e^{-\mu_i t} \tag{5.1}$$

and

$$f(t) = (-1)^{n+1} \sum_{i=1}^{n} \mu_i \left(\prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{\mu_j}{\mu_i - \mu_j} \right) e^{-\mu_i t}, \tag{5.2}$$

for $t > 0$ and $T = T_1 + T_2 + \cdots + T_n$

Proof:

When $n = 1$

$T = T_1 = \mu_1 e^{-\mu_1 t}$

When $n = 2$

$T = T_2 = \mu_2 e^{-\mu_2 t}$

Since T_1 and T_2 are independent, therefore, the joint distribution function of T_1 and T_2 is

$f(t_1, t_2) = \mu_1 e^{-\mu_1 t_1} \cdot \mu_2 e^{-\mu_2 t_2} = \mu_1 \mu_2 e^{-\mu_1 t_1 - \mu_2 t_2}$

$$
\begin{aligned}
P\{T \le t\} &= P\{T_1 + T_2 \le t\} = F(t) \\
&= \int_0 \int_0^{-t_1} \mu_1 \mu_2 e^{-\mu_1 t_1 - \mu_2 t_2} dt_1 dt_2 \\
&= \int_0 \left[-\mu_2 e^{-\mu_1 t_1 - \mu_2 t_2} \right]_0^{-t_1} dt_2 \\
&= \int_0 \left[-\mu_2 e^{-\mu_1 t + (\mu_1 - \mu_2) t_2} + \mu_2 e^{-\mu_2 t_2} \right] dt_2 \\
&= \left[\frac{-\mu_2}{\mu_1 - \mu_2} e^{-\mu_1 t + (\mu_1 - \mu_2) t_2} - e^{-\mu_2 t_2} \right]_0^t \\
&= \frac{-\mu_2}{\mu_1 - \mu_2} e^{-\mu_1 t + (\mu_1 - \mu_2) t} - e^{-\mu_2 t} + \frac{\mu_2}{\mu_1 - \mu_2} e^{-\mu_1 t} + 1 \\
&= \frac{\mu_2}{\mu_1 - \mu_2} e^{-\mu_1 t} + \frac{\mu_1}{\mu_2 - \mu_1} e^{-\mu_2 t} + 1
\end{aligned}
$$

$$
f(t) = F'(t) = \frac{-\mu_1 \mu_2}{\mu_1 - \mu_2} e^{-\mu_1 t} - \frac{\mu_1 \mu_2}{\mu_2 - \mu_1} e^{-\mu_2 t}
$$

Assume that when $n = k$, (5.2) holds, i.e. $T = T_1 + T_2 + \cdots + T_n$, and

$$
f(t) = (-1)^{n+1} \sum_{i=1}^{n} \mu_i \left(\prod_{\substack{j=1 \\ j \ne i}}^{n} \frac{\mu_j}{\mu_i - \mu_j} \right) e^{-\mu_i t}
$$

When $n = k + 1$,

Let $Y_1 = T = T_1 + T_2 + \cdots + T_k$, $Y_2 = T_{k+1}$, and $Y = Y_1 + Y_2$

$$
f(y_1, y_2) = \mu_1 \mu_2 \cdots \mu_k \mu_{k+1} \sum_{i=1}^{k} (-1)^{k+1} \left(\prod_{\substack{j=1 \\ j \ne i}}^{k} \frac{1}{\mu_i - \mu_j} \right) e^{-\mu_i y_1} e^{-\mu_{k+1} y_2}
$$

Let $\mu = \mu_1\mu_2\cdots\mu_k\mu_{k+1}$

$F(y) = P\{Y \le y\} = P\{Y_1 + Y_2 \le y\}$

$$= \int_0^y \int_0^{y-y_2} \mu \sum_{i=1}^{k} (-1)^{k+1} \left(\prod_{\substack{j=1 \\ j\neq i}}^{k} \frac{1}{\mu_i - \mu_j} \right) e^{-\mu_1 y_1} e^{-\mu_2 y_2} \, dy_1 dy_2$$

$$= \int_0^y \left[\mu \sum_{i=1}^{k} \frac{1}{\mu_i} (-1)^k \left(\prod_{\substack{j=1 \\ j\neq i}}^{k} \frac{1}{\mu_i - \mu_j} \right) e^{-\mu_i y_1} e^{-\mu_{k+1} y_2} \right]_0^{y-y_2} dy_2$$

$$= \int_0^y \left[\mu \sum_{i=1}^{k} \frac{1}{\mu_i} (-1)^k \left(\prod_{\substack{j=1 \\ j\neq i}}^{k} \frac{1}{\mu_i - \mu_j} \right) e^{-\mu_i y} e^{(\mu_i - \mu_{k+1}) y_2} - \mu \sum_{i=1}^{k} \frac{1}{\mu_i} (-1)^k \left(\prod_{\substack{j=1 \\ j\neq i}}^{k} \frac{1}{\mu_i - \mu_j} \right) e^{-\mu_{k+1} y_2} \right] dy_2$$

$$= \left[\mu \sum_{i=1}^{k} \frac{1}{\mu_i} (-1)^k \left(\prod_{\substack{j=1 \\ j\neq i}}^{k+1} \frac{1}{\mu_i - \mu_j} \right) e^{-\mu_i y} e^{(\mu_i - \mu_{k+1}) y_2} + \frac{\mu}{\mu_{k+1}} \sum_{i=1}^{k} \frac{1}{\mu_i} (-1)^k \left(\prod_{\substack{j=1 \\ j\neq i}}^{k} \frac{1}{\mu_i - \mu_j} \right) e^{-\mu_{k+1} y_2} \right]_0^y$$

$$= \mu \sum_{i=1}^{k} \frac{1}{\mu_i} (-1)^k \left(\prod_{\substack{j=1 \\ j\neq i}}^{k+1} \frac{1}{\mu_i - \mu_j} \right) e^{-\mu_{k+1} y} + \frac{\mu}{\mu_{k+1}} \sum_{i=1}^{k} \frac{1}{\mu_i} (-1)^{\cdot} \left(\prod_{\substack{j=1 \\ j\neq i}}^{k} \frac{1}{\mu_i - \mu_j} \right) e^{-\mu_{k+1} y}$$

$$- \mu \sum_{i=1}^{k} \frac{1}{\mu_i} (-1)^k \left(\prod_{\substack{j=1 \\ j\neq i}}^{k+1} \frac{1}{\mu_i - \mu_j} \right) e^{-\mu_i y} - \frac{\mu}{\mu_{k+1}} \sum_{i=1}^{k} \frac{1}{\mu_i} (-1)^k \left(\prod_{\substack{j=1 \\ j\neq i}}^{k} \frac{1}{\mu_i - \mu_j} \right)$$

$$= \mu\sum_{i=1}^{k}\left(\frac{1}{\mu_i-\mu_{k+1}}\right)\left(\frac{1}{\mu_i}\right)(-1)^k\left(\prod_{\substack{j=1\\ j\neq i}}^{k}\frac{1}{\mu_i-\mu_j}\right)e^{-\mu_{k+1}y} + \frac{\mu}{\mu_{k+1}}\sum_{i=1}^{k}\frac{1}{\mu_i}(-1)^k\left(\prod_{\substack{j=1\\ j\neq i}}^{k}\frac{1}{\mu_i-\mu_j}\right)e^{-\mu_{k+1}y}$$

$$-\mu\sum_{i=1}^{k}\frac{1}{\mu_i}(-1)^k\left(\prod_{\substack{j=1\\ j\neq i}}^{k+1}\frac{1}{\mu_i-\mu_j}\right)e^{-\mu_i y} - \frac{\mu}{\mu_{k+1}}\sum_{i=1}^{k}\frac{1}{\mu_i}(-1)^k\left(\prod_{\substack{j=1\\ j\neq i}}^{k}\frac{1}{\mu_i-\mu_j}\right)$$

$$= \frac{\mu}{\mu_{k+1}}\sum_{i=1}^{k}\left(\frac{\mu_{k+1}}{\mu_i-\mu_{k+1}}+1\right)\left(\frac{1}{\mu_i}\right)(-1)^k\left(\prod_{\substack{j=1\\ j\neq i}}^{k}\frac{1}{\mu_i-\mu_j}\right)e^{-\mu_{k+1}y}$$

$$-\mu\sum_{i=1}^{k}\frac{1}{\mu_i}(-1)^k\left(\prod_{\substack{j=1\\ j\neq i}}^{k+1}\frac{1}{\mu_i-\mu_j}\right)e^{-\mu_i y} - \frac{\mu}{\mu_{k+1}}\sum_{i=1}^{k}\frac{1}{\mu_i}(-1)^k\left(\prod_{\substack{j=1\\ j\neq i}}^{k}\frac{1}{\mu_i-\mu_j}\right)$$

$$= \frac{\mu}{\mu_{k+1}}\sum_{i=1}^{k}\left(\frac{1}{\mu_{k+1}-\mu_i}\right)(-1)^{k+1}\left(\prod_{\substack{j=1\\ j\neq i}}^{k}\frac{1}{\mu_i-\mu_j}\right)e^{-\mu_{k+1}y}$$

$$+\mu\sum_{i=1}^{k}\frac{1}{\mu_i}(-1)^{k+1}\left(\prod_{\substack{j=1\\ j\neq i}}^{k+1}\frac{1}{\mu_i-\mu_j}\right)e^{-\mu_i y} + \frac{\mu}{\mu_{k+1}}\sum_{i=1}^{k}\frac{1}{\mu_i}(-1)^{k+1}\left(\prod_{\substack{j=1\\ j\neq i}}^{k}\frac{1}{\mu_i-\mu_j}\right)$$

$$= \frac{\mu}{\mu_{k+1}}\sum_{i=1}^{k}(-1)^{k+1}\left(\prod_{\substack{j=1\\ j\neq i}}^{k+1}\frac{1}{\mu_i-\mu_j}\right)e^{-\mu_{k+1}y} + \mu\sum_{i=1}^{k}(-1)^{k+1}\frac{1}{\mu_i}\left(\prod_{\substack{j=1\\ j\neq i}}^{k+1}\frac{1}{\mu_i-\mu_j}\right)e^{-\mu_i y}$$

$$+\frac{\mu}{\mu_{k+1}}\sum_{i=1}^{k}\frac{1}{\mu_i}(-1)^{k+1}\left(\prod_{\substack{j=1\\ j\neq i}}^{k}\frac{1}{\mu_i-\mu_j}\right)$$

$$= \mu\sum_{i=1}^{k}(-1)^{k+1}\frac{1}{\mu_{k+1}}\left(\prod_{\substack{j=1\\ j\neq i}}^{k+1}\frac{1}{\mu_i-\mu_j}\right)e^{-\mu_{k+1}y} + \mu\sum_{i=1}^{k}(-1)^{k+1}\frac{1}{\mu_i}\left(\prod_{\substack{j=1\\ j\neq i}}^{k+1}\frac{1}{\mu_i-\mu_j}\right)e^{-\mu_i y}$$

$$+\frac{\mu}{\mu_{k+1}}\sum_{i=1}^{k}\frac{1}{\mu_i}(-1)^{k+1}\left(\prod_{\substack{j=1\\ j\neq i}}^{k}\frac{1}{\mu_i-\mu_j}\right)$$

$$= \mu \sum_{i=1}^{k+1} (-1)^{k+1} \frac{1}{\mu_i} \left(\prod_{\substack{j=1 \\ j \neq i}}^{k+1} \frac{1}{\mu_i - \mu_j} \right) e^{-\mu_i y} + \frac{\mu}{\mu_{k+1}} \sum_{i=1}^{k} (-1)^{k+1} \frac{1}{\mu_i} \left(\prod_{\substack{j=1 \\ j \neq i}}^{k} \frac{1}{\mu_i - \mu_j} \right)$$

$$= (-1)^{k+1} \sum_{i=1}^{k+1} \left(\prod_{\substack{j=1 \\ j \neq i}}^{k+1} \frac{\mu_j}{\mu_i - \mu_j} \right) e^{-\mu_i y} + (-1)^{k+1} \sum_{i=1}^{k} \left(\prod_{\substack{j=1 \\ j \neq i}}^{k} \frac{\mu_j}{\mu_i - \mu_j} \right)$$

Since $F(\infty) = 1$, therefore

$$F(y) = 1 - (-1)^{k+2} \sum_{i=1}^{k+1} \left(\prod_{\substack{j=1 \\ j \neq i}}^{k+1} \frac{\mu_j}{\mu_i - \mu_j} \right) e^{-\mu_i y}$$

Furthermore,

$$f(y) = F'(y) = (-1)^{k+2} \sum_{i=1}^{k+1} \mu_i \left(\prod_{\substack{j=1 \\ j \neq i}}^{k+1} \frac{\mu_j}{\mu_i - \mu_j} \right) e^{-\mu_i y}$$

APPENDIX B. PROOF OF THEOREM 5.2

THEOREM 5.2 The cumulative distribution function and the probability density function of $T = T_1 + T_2 + \cdots + T_N$ is given by

$$F(t) = \sum_{n=1}^{\infty} \left[p(1-p)^{n-1} \right] \cdot \left(1 - (-1)^{n+1} \sum_{i=1}^{n} \left(\prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{\mu_j}{\mu_i - \mu_j} \right) e^{-\mu_i t} \right) \tag{5.5}$$

and

$$f(t) = \sum_{n=1}^{\infty} \left[p(1-p)^{n-1} \right] \cdot \left((-1)^{n+1} \sum_{i=1}^{n} \mu_i \left(\prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{\mu_j}{\mu_i - \mu_j} \right) e^{-\mu_i t} \right) \tag{5.6}$$

where T_i is exponentially distributed with a parameter μ_i and N is geometrically distributed with parameter p. And, T_i and N are independent.

Proof:

$$P\{T \le s\} = P\{T_1 + T_2 + \cdots + T_N \le s\}$$

$$= \sum_{n=1}^{\infty} P\{T_1 + T_2 + \cdots + T_n \le s | N = n\} P\{N = n\}$$

$$= \sum_{n=1}^{\infty} P\{T_1 + T_2 + \cdots + T_n \le s\} P\{N = n\} \quad (\because T_i\text{'s and } N \text{ are independent})$$

$$= \sum_{n=1}^{\infty} F_1(s) P\{N = n\}$$

where $F_1(t)$ is the distribution function of $T = T_1 + T_2 + \cdots + T_n$, n is known.

Thus, according to THEOREM 5.1, the cumulative distribution function of $T = T_1 + T_2 + \cdots + T_N$ is given by

$$F(s)=\sum_{n=1}^{\infty}\left[p(1-p)^{n-1}\right]\left(1-(-1)^{n+1}\sum_{i=1}^{n}\left(\prod_{\substack{j=1\\ j\neq i}}^{n}\frac{\mu_j}{\mu_i-\mu_j}\right)e^{-\mu_i s}\right)$$

Therefore, the probability density function can be obtained by differentiating $F(s)$

$$f(s)=\frac{d}{ds}\sum_{n=1}^{\infty}F_1(s)P\{N=n\}$$

$$=\sum_{n=1}^{\infty}\frac{d}{ds}F_1(s)P\{N=n\}$$

$$=\sum_{n=1}^{\infty}f_1(s)P\{N=n\}$$

where $f_1(t)$ is the probability density of $T=T_1+T_2+\cdots+T_n$, n is known.

Thus, according to THEOREM 5.1, the probability density function of $T=T_1+T_2+\cdots+T_N$ is given by

$$f(s)=\sum_{n=1}^{\infty}\left[p(1-p)^{n-1}\right]\left((-1)^{n+1}\sum_{i=1}^{n}\mu_i\left(\prod_{\substack{j=1\\ j\neq i}}^{n}\frac{\mu_j}{\mu_i-\mu_j}\right)e^{-\mu_i s}\right)$$

APPENDIX C. PROOF OF LEMMA 5.2.1

LEMMA 5.2.1 The mean of $T = T_1 + T_2 + \cdots + T_N$ is given by

$$E(t) = \sum_{n=1}^{\infty} p(1-p)^{n-1} \sum_{i=1}^{n} \frac{1}{\mu_i} \tag{5.7}$$

where T_i is exponentially distributed with a parameter μ_i and N is geometrically distributed with parameter p.

Proof:

$$E\left[\sum_{i=1}^{N} T_i\right] = E\left[E\left[\sum_{i=1}^{N} T_i | N = n\right]\right]$$

$$= E\left[E\left[\sum_{i=1}^{N} E[T_i] N = n\right]\right]$$

$$= \sum_{n=1}^{\infty} \sum_{i=1}^{n} E[T_i] p(N = n)$$

$$= \sum_{n=1}^{\infty} p(1-p)^{n-1} \sum_{i=1}^{n} \frac{1}{\mu_i}$$

Therefore,

$$E(t) = \sum_{n=1}^{\infty} p(1-p)^{n-1} \sum_{i=1}^{n} \frac{1}{\mu_i}$$

APPENDIX D. PROOF OF LEMMA 5.2.2

LEMMA 5.2.2 The variance of $T = T_1 + T_2 + \cdots + T_N$ is given by

$$Var(t) = \sum_{n=1}^{\infty} p(1-p)^{n-1} \sum_{i=1}^{n} \frac{2}{\mu_i^2} - \sum_{n=1}^{\infty} p^2 (1-p)^{2n-2} \sum_{i=1}^{n} \frac{1}{\mu_i^2} \tag{5.8}$$

where T_i is exponentially distributed with a parameter μ_i and N is geometrically distributed with parameter p.

Proof:

$$Var\left(\sum_{i=1}^{N} T_i\right) = E\left[\left(\sum_{i=1}^{N} T_i\right)^2\right] - \left(E\left[\sum_{i=1}^{N} T_i\right]\right)^2$$

$$= \sum_{n=1}^{\infty} E\left[\left(\sum_{i=1}^{n} T_i\right)^2\right] p(N=n) - \left(\sum_{n=1}^{\infty} E\left[\sum_{i=1}^{n} T_i\right] p(N=n)\right)^2$$

where $p(N=n) = p(1-p)^{n-1}$ (i.e. The probability mass function of the geometric random variable)

$$= \sum_{n=1}^{\infty} p(N=n)\left(Var\left(\sum_{i=1}^{n} T_i\right) + \left(E\left[\sum_{i=1}^{n} T_i\right]\right)^2\right) - \left(\sum_{n=1}^{\infty} p(N=n)^2 \sum_{i=1}^{n} E[T_i]^2\right)$$

$$= \sum_{n=1}^{\infty} p(N=n) \sum_{i=1}^{n} \left(Var(T_i) + (E[T_i])^2\right) - \left(\sum_{n=1}^{\infty} p(N=n)^2 \sum_{i=1}^{n} E[T_i]^2\right)$$

$$= \sum_{n=1}^{\infty} p(1-p)^{n-1} \sum_{i=1}^{n} \left(\frac{1}{\mu_i^2} + \left(\frac{1}{\mu_i}\right)^2\right) - \sum_{n=1}^{\infty} p^2 (1-p)^{2n-2} \sum_{i=1}^{n} \left(\frac{1}{\mu_i}\right)^2$$

$$= \sum_{n=1}^{\infty} p(1-p)^{n-1} \sum_{i=1}^{n} \frac{2}{\mu_i^2} - \sum_{n=1}^{\infty} p^2 (1-p)^{2n-2} \sum_{i=1}^{n} \frac{1}{\mu_i^2}$$

APPENDIX E. PROOF OF THEOREM 5.3

THEOREM 5.3 Given n child nodes with time durations represented as independent random variables $T_{11}, T_{12}, \cdots, T_{1n}$, and a parent node, T_1, having a distribution as in THEOREM 5.2, the cumulative distribution function of T is given by

$$F(t) = \int_0^t F_{11}(t-t_1)F_{12}(t-t_1)\cdots F_{1n}(t-t_1)f_1(t_1)dt_1 \qquad (5.10)$$

where $T = Max\{T_{11}, T_{12}, \cdots, T_{1n}\} + T_1$, and $f_1(t)$ and $F_{1i}(s)$ are the probability density function and the cumulative distribution function of T_1 and T_{1i}, respectively.

Proof:

$$\begin{aligned}
&F(t) = P\{T \le t\} \\
&= P\{(Max\{T_{11}, T_{12}, \cdots, T_{1n}\} + T_1) \le t\} \\
&= P\{T_{11} + T_1 \le t, T_{12} + T_1 \le t, \cdots, T_{1n} + T_1 \le t\} \\
&= \int P\{T_{11} + T_1 \le t, T_{12} + T_1 \le t, \cdots, T_{1n} + T_1 \le t | T_1 = t_1\} f_1(t_1)dt_1 \\
&= \int P\{T_{11} + t_1 \le t, T_{12} + t_1 \le t, \cdots, T_{1n} + t_1 \le t\} f_1(t_1)dt_1
\end{aligned}$$

Since T_{1i}'s are mutually independent

$$\begin{aligned}
&= \int P\{T_{11} + t_1 \le t\} \cdot P\{T_{12} + t_1 \le t\} \cdot \cdots \cdot P\{T_{1n} + t_1 \le t\} \cdot f_1(t_1)dt_1 \\
&= \int P\{T_{11} \le t - t_1\} \cdot P\{T_{12} \le t - t_1\} \cdot \cdots \cdot P\{T_{1n} \le t - t_1\} \cdot f_1(t_1)dt_1 \\
&= \int F_{11}(t-t_1) \cdot F_{12}(t-t_1) \cdot \cdots \cdot F_{1n}(t-t_1) \cdot f_1(t_1)dt_1
\end{aligned}$$

where $f_1(t)$ and $F_{1i}(s)$ are the probability density function and the cumulative distribution function of T_1 and T_{1i} given in THEOREM 5.2 respectively.

APPENDIX F. PROOF OF LEMMA 5.3.1

LEMMA 5.3.1 The mean of $T = Max\{T_{11}, T_{12}, \cdots, T_{1n}\} + T_1$ is given by

$$E[T] = \int_0^\infty (1 - F(t))dt \tag{5.11}$$

where $F(t)$ is the cumulative distribution function of $T = Max\{T_{11}, T_{12}, \cdots, T_{1n}\} + T_1$.

Proof:

$$E[T] = \int_0^\infty t \cdot f(t)dt$$
$$= \int_0^\infty \left[\int_0^t dy\right] f(t)dt$$
$$= \int_0^\infty \int_y^\infty f(y)dydt$$
$$= \int_0^\infty [F(y)]_y^\infty dt$$
$$= \int_0^\infty (1 - F(t))dt$$

APPENDIX G. PROOF OF LEMMA 5.3.2

LEMMA 5.3.2 The variance of $T = Max\{T_{11}, T_{12}, \cdots, T_{1n}\} + T_1$ is given by

$$Var[T] = 2\int_0^\infty \int_t^\infty (1 - F(x))dxdt - \left(\int_0^\infty (1 - F(t))dt\right)^2 \tag{5.12}$$

where $F(t)$ is the cumulative distribution function of $T = Max\{T_{11}, T_{12}, \cdots, T_{1n}\} + T_1$.

Proof:

$$Var[T] = E[T^2] - (E[T])^2$$

$$E[T^2] = \int_0^\infty t^2 f(t)dt$$

$$= \int_0^\infty \left[\int_0^t 2xdx\right] f(t)dt$$

$$= 2\int_0^\infty \left[\int_0^t xdx\right] f(t)dt$$

$$= 2\int_0^\infty \left[\int_x^\infty xf(x)dx\right] dt$$

$$= 2\int_0^\infty \left[\int_x^\infty \left[\int_0^y dy\right] f(x)dx\right] dt$$

$$= 2\int_0^\infty \left[\int_x^\infty \left[\int_x^y f(y)dy\right] dx\right] dt$$

$$= 2\int_0^\infty \left[\int_x^\infty (1 - F(x))dx\right] dt$$

$$= 2\int_0^\infty \int_x^\infty (1 - F(x))dxdt$$

According to THEOREM 5.8,

$$E[T] = \int_0^\infty (1 - F(t))dt$$

Therefore,

$$Var[T] = 2\int_0^\infty \int_x^\infty (1 - F(x))dxdt - \left(\int_0^\infty (1 - F(t))dt\right)^2$$

APPENDIX H. MATLAB PROGRAM FOR A DESIGN TASK

```
clear all
TIC
syms F f t s F1 M fid V x NewF exp2 exp2t %exp2 = E(x^2)
n = 9;
N = [1:n];
mu= 5*N;
p = 0.9;
C = 0;
F = 0;
f = 0;
for k=1:n % n = 1 to inf
  P = p*((1-p)^(k-1));
  G = 0;
  I = 0;
  for i=1:k % i = 1 to n
    B = 1;
    %starting j
    for j=1:k % j = 1 to n, j ~= i
      if j ~= i
        A = mu(j)/((mu(i)-mu(j)));
        B = B*A;
      end
    end
    %finishing j
    C = (-1)^(k+1)*B*exp(-mu(i)*t);
    H = mu(i)*C;
    G=G+C;
    I=I+H;

 end

 E = 1-G;
 F = F+(P*E);
 f = f+(P*I);
end
M = int(1-F, t, 0, 1000);
V = 2*int(int(1-F,t,x,1000),x,0,1000)-M^2;
subs(M)
subs(V)


TOC
```

APPENDIX I. MATLAB PROGRAM FOR A TWO-LEVEL DST

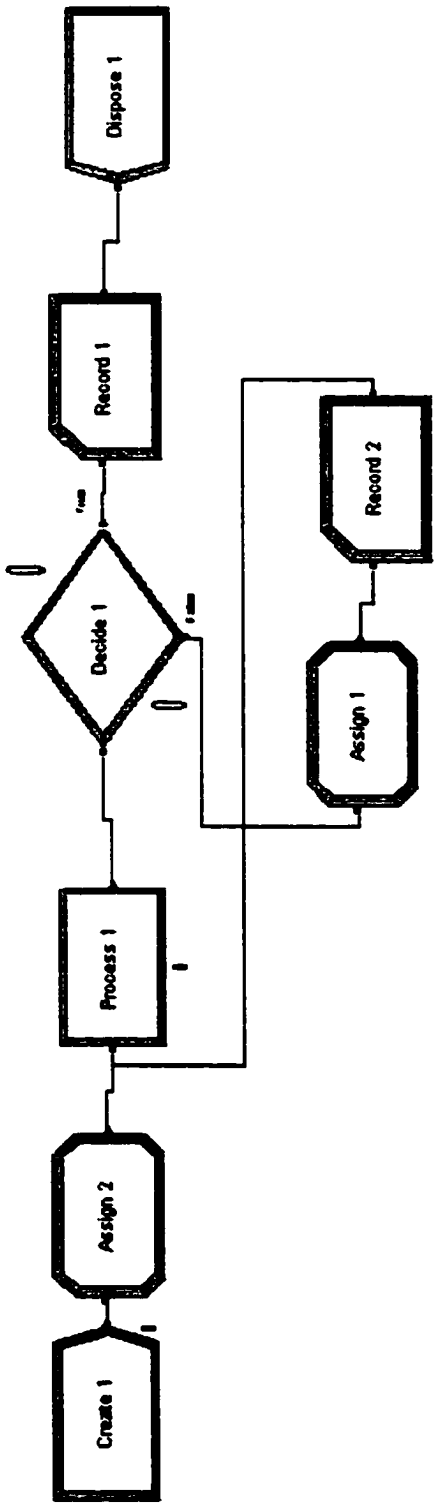
```
clear all
TIC
syms F f t s F1 M x V
n = 9;
N = [1:n];
mu= 5*N;
p = 0.9;
C = 0;
F = 0;
f = 0;
for k=1:n
  P = p*((1-p)^(k-1));
  G = 0;
  I = 0;
  for i=1:k
    B = 1;
    %starting j
   for j=1:k
     if j ~= i
      A = mu(j)/((mu(i)-mu(j)));
      B = B*A;
     end
   end
   %finishing j
   C=(-1)^(k+1)*B*exp(-mu(i)*t);
   H=mu(i)*C;
   G=G+C;
   I=I+H;
end
E = 1-G;
 F = F+(P*E);
 f = f+(P*I);
end
NewF=subs(F,t,s-t);

F1=int(NewF*NewF*f,t,0,s);

M=int(1-F1,s,0,1000);
V = 2*int(int(1-F1,s,x,1000),x,0,1000)-M^2;
subs(M)
subs(V)
TOC
```

APPENDIX J. MATLAB PROGRAM FOR A THREE-LEVEL DST

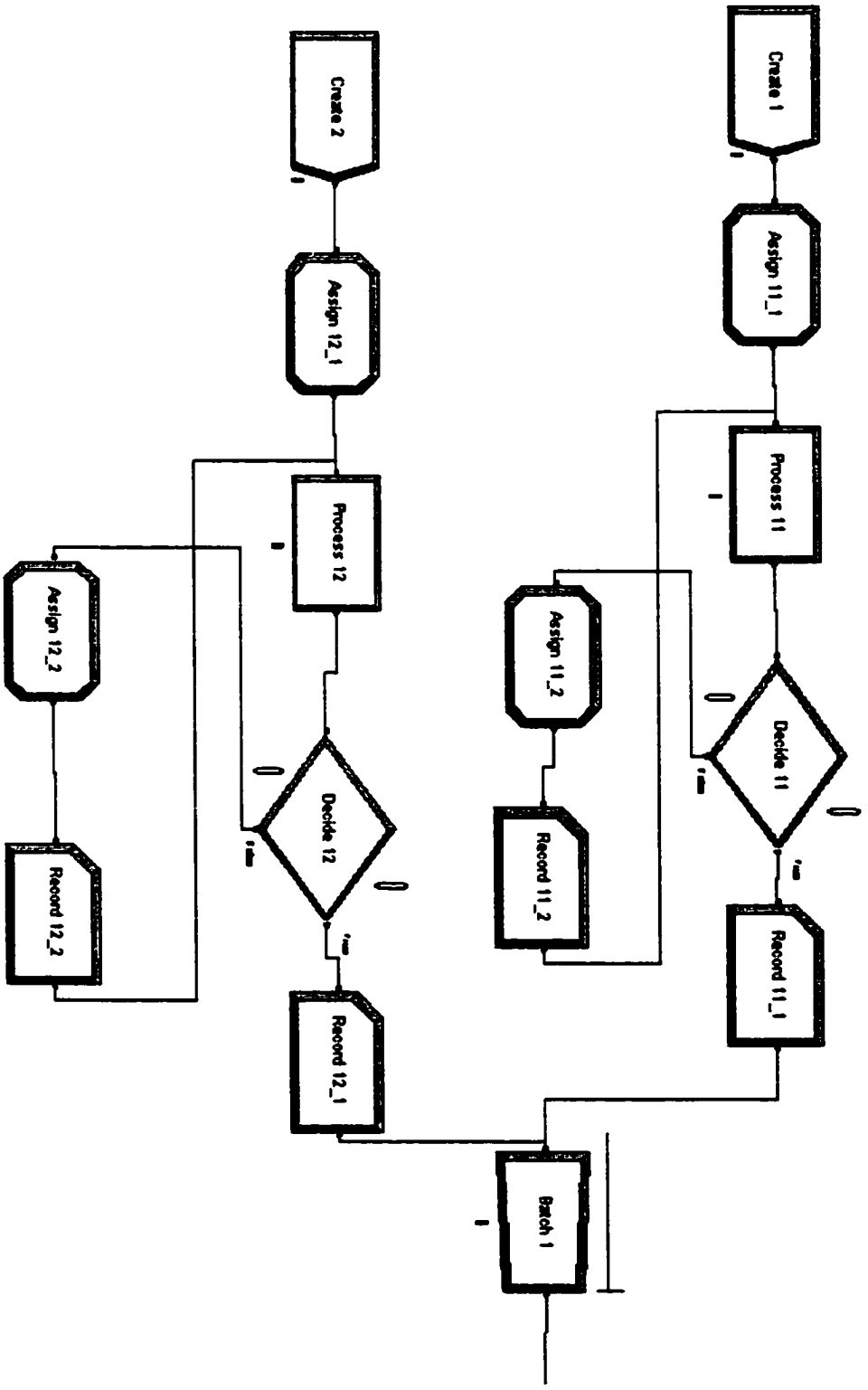
```
clear all
TIC
syms F f t s F1 F2 NewF NewF1 M x y f1 V
n = 9;
N = [1:n];
mu= 5*N;
p = 0.9;
C = 0;
F = 0;
f = 0;
for k=1:n
  P = p*((1-p)^(k-1));
  G = 0;
  I = 0;
  for i=1:k
    B = 1;
    %starting j
    for j=1:k
      if j ~= i
       A = mu(j)/((mu(i)-mu(j)));
       B = B*A;
      end
    end
    %finishing j
    C=(-1)^(k+1)*B*exp(-mu(i)*t);
    H=mu(i)*C;
    G=G+C;
    I=I+H;

 end

 E = 1-G;
 F = F+(P*E);
 f = f+(P*I);
end

NewF=subs(F,t,s-t);

F1=int(NewF*NewF*f,t,0,s); %cdf of the second level

NewF1=subs(F1, s, y-s);
f1 = subs(f,t,s);
```

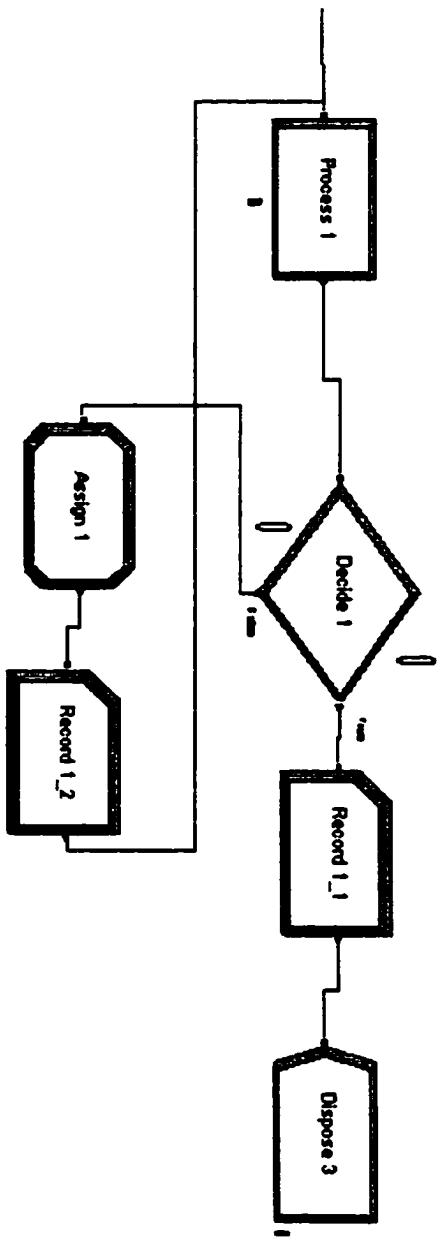
```
F2=int(NewF1*NewF1*f1,s,0,y); %cdf of the first level
M=int(1-F2,y,0,1000);
V = 2*int(int(1-F2,y,x,1000),x,0,1000)-M^2;
subs(M)
subs(V)
TOC
```

APPENDIX K. ARENA MODEL FOR ONE-LEVEL DST

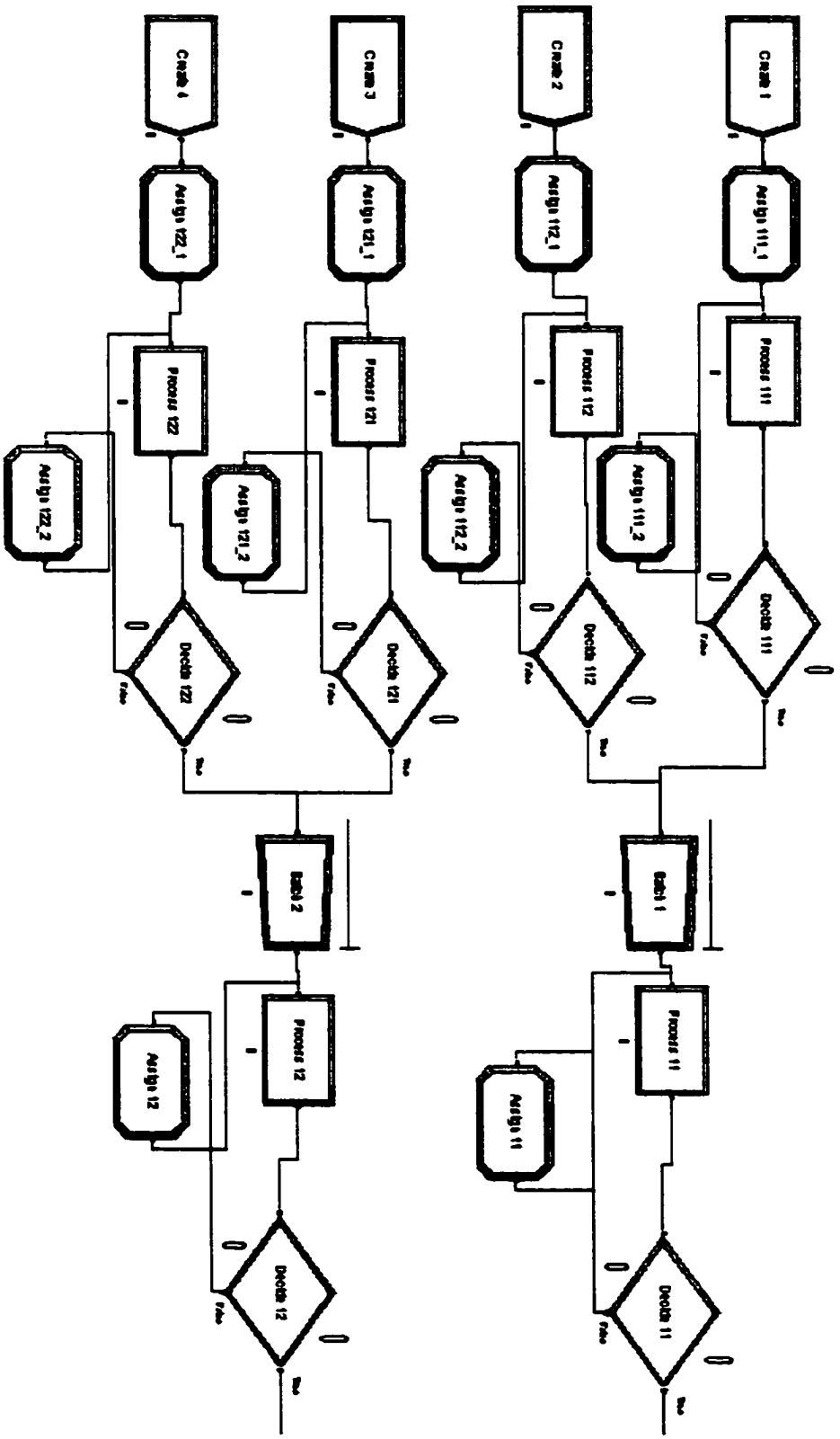


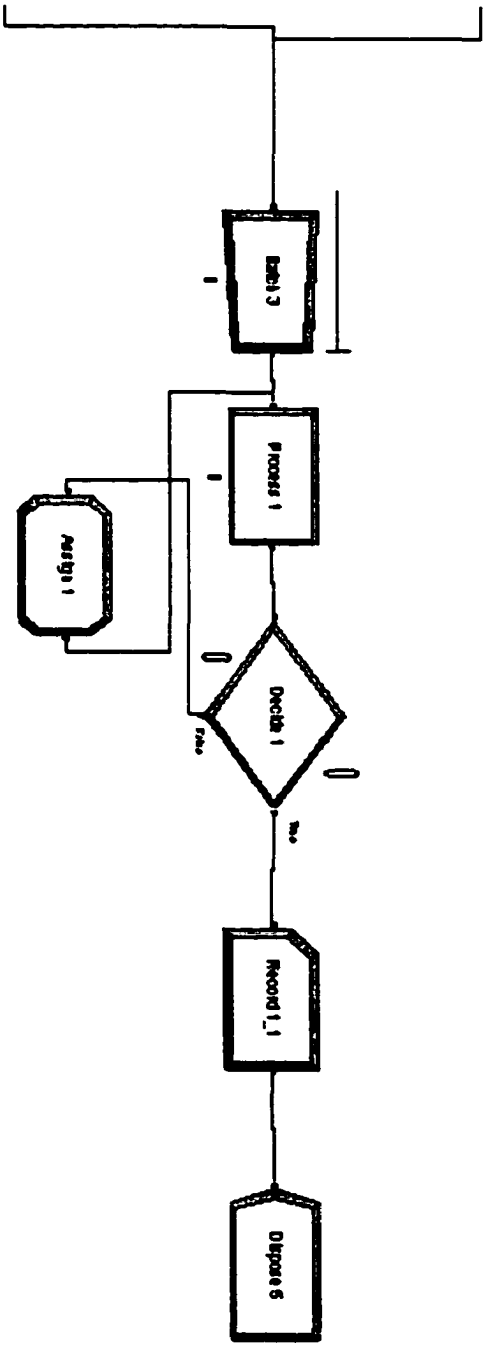
APPENDIX L. ARENA MODEL FOR A TWO-LEVEL DST




Process 1
Decide 1
Assign 1
Record 1_2
Record 1_1
Dispose 3

APPENDIX M. ARENA MODEL FOR A THREE-LEVEL DST




Assign 1
Decide 1
Record1_1
Dispose 6

REFERENCES

Ahmadi, R., *et al.* (2001). "Structuring product development processes." *European Journal of Operational Research* **130**(3): 539-558.

Albano, L. D. and N. P. Suh (1994). "Axiomatic design and concurrent engineering." *Computer Aided Design* **26**(7): 499-504.

Babic, B. (1999). "Axiomatic design of flexible manufacturing systems." *International Journal of Product Research* **37**(5): 1159-1173.

Balachandra, R. and J. H. Friar (1997). "Factors for success in R&D projects and new product innovation: A contextual framework." *IEEE Transactions on Engineering Management* **44**(3): 276-287.

Black, I. (1990). "Back to the future with CAD: Its impact on product design and development." *Design Studies* **11**(4): 207-211.

Black, I. (1994). "Product design in British manufacturing -- New processes from old practices." *Proceedings of the Institution of Mechanical Engineers* **208**(B2): 81-87.

Black, J. T. (1991). "The design of manufacturing systems: Axiomatic approach." *Design, Analysis, and Control of Manufacturing Cells* **53**: 1-13.

Blackburn, J. D. (1991). *Time-Based Competition: The Next Battleground in American Manufacturing*. Homewood, IL, Business One Irwin.

Braha, D. and O. Maimon (1997). "The design process: Properties, paradigms, and structure." *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* **27**(2): 146-166.

Braha, D. and O. Maimon (1999). "A mathematical theory of design: Modeling the design process (Part II)." *International Journal of General Systems* **27**(4-5): 319-347.

Brown, S. L. and K. M. Eisenhardt (1995). "Product development: Past research, present findings, and future directions." *Academic Management Review* **20**: 343-378.

Carrascosa, M., *et al.* (1998). Using the design structure matrix to estimate product development time. *Proceedings of DETC's 98 1998 ASME Design Engineering Technical Conference*, Atlanta, GA.

Christian, A. D. and W. P. Seering (1995). "A model of information exchange in the design process." *Design Engineering Technical Conference* **2**: 323-328.

Clark, K. B. and T. Fujimoto (1991). *Product Development Performance: Strategy, Organization, and Management in the World Auto Industry*. Boston, MA, Harvard Business School.

Cusumano, M. A. and K. Nobeoka (1992). "Strategy, structure and performance in product development -- Observations from the auto industry." *Research Policy* **21**: 265-293.

Eppinger, S. D. (1991). Model-based approaches to managing concurrent engineering. *International Conference on Engineering Design*, Zurich.

Eversheim, W., *et al.* (1997). "Information management for concurrent engineering." *European Journal of Operational Research* **100**(2): 253-265.

Feldman, R. M. and C. Valdez-Flores (1996). *Applied Probability and Stochastic Processes*. Boston, MA, PWS Publishing Company.

Finger, S. and J. R. Dixon (1989a). "A review of research in mechanical engineering design, part I: Descriptive, prescriptive, and computer-based models of design processes." *Research in Engineering Design* **1**(1): 51-68.

Finger, S. and J. R. Dixon (1989b). "A review of research in mechanical engineering design, part II: Representations, analysis, and design for the life cycle." *Research in Engineering Design* **1**(2): 121-137.

Frankenberger, E. and P. Badke-Schaub (1998). "Modeling design processes in industry - empirical investigations of design work in practice." *Automation in Construction* **7**: 139-155.

Gebala, D. A. and S. D. Eppinger (1991). "Methods for analyzing design procedures." *American Society of Mechanical Engineers, Design Engineering Division (Publication)* **31**: 227-233.

Griffin, A. and J. R. Hauser (1996). "Integrating R&D and marketing: A review and analysis of the literature." *Journal of Product Innovation Management* **13**(3): 191-215.

Hanselman, D., *et al.* (2000). *Mastering MATLAB 6: A Comprehensive Tutorial and Reference*. Upper Saddle River, NJ, Prentice Hall.

Hazelrigg, G. A. (1999). "An axiomatic framework for engineering design." *Journal of Mechanical Design* **121**: 342-347.

Jackman, J. (1998). Concurrent engineering lecture notes. Ames, IA.

Kelton, W. D., *et al.* (2001). *Simulation with Arena, second edition*. Boston, MA, McGraw-Hill.

Kim, S.-J., *et al.* (1991). "Design of software system based on axiomatic design." *Annals of the CIRP* **40**(1): 165-170.

Krishnan, V. (1996). "Managing the simultaneous execution of coupled phases in concurrent product development." *IEEE Transactions on Engineering Management* **43**(2): 210-217.

Krishnan, V., *et al.* (1993). "Iterative overlapping: Accelerating product development by preliminary information exchange." *American Society of Mechanical Engineers, Design Engineering Division (Publication)* **53**: 223-231.

Krishnan, V., *et al.* (1995). "Accelerating product development by the exchange of preliminary product design information." *Journal of Mechanical Design* **117**: 491-498.

Krishnan, V., *et al.* (1997a). "A model-based framework to overlap product development activities." *Management Science* **43**(4): 437-451.

Krishnan, V., *et al.* (1997b). "Simplifying iterations in cross-functional design decision making." *Journal of Mechanical Design* **119**: 485-493.

Krishnan, V. and K. T. Ulrich (2001). "Product development decisions: A review of the literature." *Management Science* **47**(1): 1-21.

Kusiak, A. and K. Park (1990). "Concurrent engineering: decomposition and scheduling of design activities." *International Journal of Product Research* **28**(10): 1883-1900.

Kusiak, A. and J. Wang (1993a). "Decomposition in concurrent design, " in *Concurrent Engineering: Automation, Tools, and Techniques*. New York, NY, John Wiley & Sons: 481-507.

Kusiak, A. and J. Wang (1993b). "Efficient organizing of design activities." *International Journal of Product Research* **31**(4): 753-769.

Kusiak, A. (1999). *Engineering Design: Products, Processes, and Systems*. San Diego, Academic Press.

Lai, Y.-C. and J. Jackman (2001). "Modeling product development processes." *Working Papaer, IMSE Department, Iowa State University, Ames, Iowa.*

Loch, C. H. and C. Terwiesch (1998). "Communication and uncertainty in concurrent engineering." *Management Science* **44**(8): 1032-1048.

Maimon, O. and D. Braha (1999). "A mathematical theory of design: Representation of design artifacts (Part I)." *International Journal of General Systems* **27**(4-5): 275-318.

Montoya-Weiss, M. M. and R. Calantone (1994). "Determinants of new product performance: A review and meta-analysis." *Journal of Product Innovation Management* **11**(5): 397-417.

Morelli, M. D., *et al.* (1995). "Predicting technical communication in product development organizations." *IEEE Transactions on Engineering Management* **42**(3): 215-222.

Roemer, T. A., *et al.* (2000). "Time-cost trade-offs in overlapped product development." *Operations Research* **48**(6): 858-865.

Ross, S. M. (1996). *Stochastic Processes, second edition*. New York, NY, Wiley, John & Sons.

Shocker, A. D. and V. Srinivasan (1979). "Multiattribute approaches for product concept evaluation and generation: A critical review." *Journal of Marketing Research* **16**: 159-180.

Simon, H. A. (1996). *The Sciences of the Artificial, third edition*. Cambridge, MA, The MIT Press.

Smith, R. P. (1997). "The historical roots of concurrent engineering fundamentals." *IEEE Transactions on Engineering Management* **44**(1): 67-78.

Smith, R. P. and S. D. Eppinger (1997a). "Identifying controlling features of engineering design iteration." *Management Science* **43**(3): 276-293.

Smith, R. P. and S. D. Eppinger (1997b). "A predictive model of sequential iteration in engineering design." *Management Science* **43**(8): 1104-1120.

Steward, D. V. (1981). "The design structure system: A method for managing the design of complex systems." *IEEE Transactions on Engineering Management* **28**(3): 71-74.

Steward, D. V. (1981). *Systems Analysis and Management: Structure, Strategy, and Design*. New York, NY, Petrocelli Books.

Suh, N. P., *et al.* (1978). "On an axiomatic approach to manufacturing and manufacturing systems." *Journal of Engineering for Industry* **100**(2): 127-130.

Suh, N. P. (1990). *The Principles of Design*, Oxford University Press.

Suh, N. P. (1997). "Design of systems." *Annals of the CIRP* **46**(1): 75-80.

Suh, N. P. (1998). "Axiomatic design theory for systems." *Research in Engineering Design* **10**: 189-209.

Terwiesch, C. and C. H. Loch (1999). "Measuring the effectiveness of overlapping development activities." *Management Science* **45**(4): 455-465.

Ulrich, K. T. and S. D. Eppinger (2000). *Product Design and Development, second edition*, McGraw-Hill Higher Education.

Whitney, D. E. (1990). "Design the design process." *Research in Engineering Design* **2**: 3-13.

Ziemke, M. C. and M. S. Spann (1993). "Concurrent engineering's roots in the World War II era, " in *Concurrent Engineering: Contemporary Issues and Modern Design Tools*. New York, NY, Chapman and Hall: 24-41.

Zirger, B. J. and J. Hartley, L. (1996). "The effect of acceleration techniques on product development time." *IEEE Transactions on Engineering Management* **43**(2): 143-152.