5-2012

Modeling and Managing Engineering Changes in a Complex Product Development Process

Weilin Li
Syracuse University

ABSTRACT

Today's hyper–competitive worldwide market, turbulent environment, demanding customers, and diverse technological advancements force any corporations who develop new products to look into all the possible areas of improvement in the entire product lifecycle management process. One of the areas that both scholars and practitioners have overlooked in the past is Engineering Change Management (ECM).

The vision behind this dissertation is to ultimately bridge this gap by identifying main characteristics of a New Product Development (NPD) process that are potentially associated with the occurrence and magnitude of iterations and Engineering Changes (ECs), developing means to quantify these characteristics as well as the interrelationships between them in a computer simulation model, testing the effects of different parameter settings and various coordination policies on project performance, and finally gaining operational insights considering all relevant EC impacts.

The causes for four major ECM problems (occurrence of ECs, long EC lead time, high EC cost, and occurrence frequency of iterations and ECs), are first discussed diagrammatically and qualitatively. Factors that contribute to particular system behavior patterns and the causal links between them are identified through the exploratory construction of causal/causal–loop diagrams. To further understand the nature of NPD/ECM problems and verify the key assumptions made in the conceptual causal framework, three field survey studies were conducted in the summer of 2010 and 2011. Information and data were collected to assess the current practice in automobile and information technology industries where EC problems are commonly encountered.

Based upon the intuitive understanding gained from these two preparation work, a Discrete Event Simulation (DES) model is proposed. In addition to combining essential project features, such as concurrent engineering, cross functional integration, resource constraints, etc., it is distinct from existing research by introducing the capability of differentiating and characterizing various levels of uncertainties (activity uncertainty, solution uncertainty, and environmental uncertainty) that are dynamically associated with an NPD project and consequently result in stochastic occurrence of NPD iterations and ECs of two different types (emergent ECs and initiated ECs) as the project unfolds. Moreover, “feedback–loop” relationships among model variables are included in the DES model to enable more accurate prediction of dynamic work flow.

Using a numerical example, different project–related model features (e.g., learning curve effects, rework likelihood, and level of dependency of product configuration) and coordination policies (e.g., overlapping strategy, rework review strategy, IEC batching policy, and resource allocation policy) are tested and analyzed in detail concerning three major performance indicators: lead time, cost, and quality, based on which decision–making suggestions regarding EC impacts are drawn from a systems perspective. Simulation results confirm that the nonlinear dynamics of interactions between NPD and ECM plays a vital role in determining the final performance of development efforts.

MODELING AND MANAGING ENGINEERING CHANGES IN A COMPLEX PRODUCT DEVELOPMENT PROCESS

by

Weilin Li

B.S., Shanghai Jiaotong University, 2007

Dissertation
Submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in *Mechanical and Aerospace Engineering*

Syracuse University
May 2012

ACKNOWLEDGEMENT

It is a pleasure to thank those who made this dissertation possible, and my doctoral years exciting and rewarding.

First of all, I would like to express my deepest appreciation to Professor Young Moon who advised me throughout my doctoral work. His wisdom, knowledge, and commitment to the highest standards inspired and motivated me.

My doctoral research was enabled by financial support from Syracuse University, which included two years of a prestigious University Fellowship and two years of Teaching Assistantship from the Department of Mechanical & Aerospace Engineering. I am very grateful to department chair, Dr. Achille Messac, who set me on the right path toward the end goal and generously supported the toughest semester of my doctoral program — allowing me to fully concentrate on writing this dissertation. I am also heartily thankful to former department chair, Dr. Alan Levy, who provided partial support in the last semester, and gave me invaluable opportunities to gain insights and inspiration from the guest lecturer program. My gratitude also extends to the three companies in China and the United States who graciously allowed me to conduct field survey studies. Moreover, I wish to thank my professors with whom I studied and worked throughout my doctoral program, as well as the committee members who attended my qualifying and final examination/defense.

To my parents, Liang and Jianying — I can't be grateful enough for their love and affection. They are the driving force behind my accomplishments. Without them this effort would not have been possible. I would also like to especially thank my friends Jeff & Wendy Chen, Dr. Xiaofei

Hu, Dr. Oksana Shmaliy, Jingjing Sun, Peerawan Wiwattananon, Norma Young, and Xiaochen Zhang, and my research colleagues Dr. Krishna Reddi, Bochao Wang, among others too numerous to mention, who have always been with me throughout the 'ups and downs,' and who encouraged and believed in me.

Table of Contents

List of Tables

List of Figures

CHAPTER 1

INTRODUCTION

1.1 Overview

New Product Development (NPD) is an entire process from idea generation, through product design and manufacturing, and to bringing a new product to the market. On the other hand, ***Engineering Change Management (ECM)*** refers to a collection of procedures, tools, and guidelines for handling modifications and changes to already released product design specifications or locked product scope (Terwiesch and Loch 1999; Huang and Mak 1999; Bhuiyan, Gatard, and Thomson 2006, Bouikni and Desrochers 2006). While these two processes often overlap and influence each other, methodical understanding of the dynamic interactions has been scarce in the research community. From a macro organizational perspective, a comprehensive assessment that aims to quantify the combined impact of key NPD and ECM process characteristics on the performance of development efforts still remains very challenging, particularly in a resource constrained multi–project environment.

The purpose of this dissertation is to systematically investigate the dynamic interactions between NPD and ECM within a single firm. The research generalizes and develops a structured foundation for simulating the iterations and engineering changes occurred stochastically throughout the development process. Thus, it enables the prediction of key project performance indicators within a given context and provides useful managerial insights for decision makers who wish to understand how the complexity and uncertainty that is typically associated with ECM problems may influence the lead time, cost, and quality of their NPD projects.

This first chapter gives an introduction of the entire research work by overviewing the context in which ECM issues are discussed, defining the research problems that this dissertation attempts to address and the purpose of the modeling study, highlighting the research objectives, justifying the methodology adopted to achieve them, and lastly summarizing the organization of this dissertation.

1.2 The Problems

Engineering Change (EC) is a fundamental reality in any new product design and development environment. However, there is no universally accepted definition of EC either in academia or practice. Different process participants, cross–functional stakeholders, and observers describe EC differently in order to reflect their own perspectives of the iterations or modifications taken place during the product design, development and life cycle.

From a *manufacturing and inventory* standpoint (e.g. Hedge, Kekre, and Kekre 1992; Ho 1994; Balakrishnan and Chakravarty 1996; Wright 1997; Tavcar and Duhovnik 2005; Wänström and Jonsson 2005), an EC is defined as modifications to a component or a part after the product has entered production. ECM problems are presumably considered to be root causes of unstable production schedule, inconsistent bill of material planning or maintenance, and obsolete inventories in a shop floor.

Several other researchers (e.g. Huang and Mak 1999; Terwiesch and Loch 1999; Bouikni and Desrochers 2006) specify ECs as “changes and modifications to the form, fit, or function of a product or part after the definition/design is released” from a perspective of *engineering design disciplines and technical functions*. Since the design freeze time of different parts, drawings and software are all different, there is no one certain point in time after which informal design iterations should be regarded as formal ECs when compared to the previous manufacturing and inventory perspective (i.e. beginning of the mass production). However, ECs are considered to appear only in the latter half of the NPD process, most likely in those final stages of the design phase and the entire production phase.

Still others (e.g. Huge 1977; Riviere, DaCunha, and Tollenaere 2002; Eckert, Clarkson, and Zanker 2004; Bhuiyan, Gatard, and Thomson 2006), reflecting their interpretations from a *business* viewpoint, will consider “ECM not to be addressed within a particular phase of the Product Life Cycle (PLC)”. That is to say, an EC may occur at any point during the whole life cycle of a product. This is a far broader way to view any iteration or change from the very beginning of an NPD process to the time when the product is actually in use.

Despite of the above mentioned multiple diverse visions in defining engineering changes, there are several common characteristics of ECs that have been confirmed by previous theoretical and empirical literature.

First, ECs can be classified into two main categories (Loch and Terwiesch 1999; Black and Repenning 2001; Eckert, Clarkson, and Zanker 2004; Bhuiyan, Gatard, and Thomson 2006):

1) ***Emergent EC (EEC)*** originates from the problems or errors detected from activity outcomes (i.e., design data and information) that have already been frozen and formally released

to the downstream phase. In this research, EECs are assumed to occur according to a certain probability determined by the conceptualized *solution uncertainty*, which will be discussed later in more detail; and

2) ***Initiated EC (IEC)*** requested by sources outside the project's control such as changing market conditions, arising customer requirements, new legislation, or emerging technology advances any point along the NPD process in response to the conceptualized *environmental uncertainty*, which will also be discussed in later section.

Under this classification scheme, design iterations within an NPD process and *problem–induced* EECs are very similar, but occur in different situations. Both of them aim at correcting mistakes or solving problems through repetitively achieving unmet goals that have been set initially. EECs are requested rework to prior activities whose outcomes have already been finalized and released to the next phase. However, NPD iterations take place before any design information is formally released to downstream phases, and therefore it generally takes less time to handle iterations due to both a smaller rework scope and a shorter approval processing time. For simplicity, we will use the term "***Rework***" in this dissertation to refer to both iterations and EECs, unless specific distinction is required. From another standpoint, *opportunity–driven* IECs arise from new needs and requirements, which result in the adding of functionality to a product (Clarkson and Eckert 2004), or enlargement of the original design solution scope. A formal assessment and approval process is desirable in handling both types of ECs due to the associated complexity and potential risks (Terwiesch and Loch 1999; Eckert, Clarkson, and Zanker 2004).

Second, typical companies launching new products follow planned schedules. NPD projects are often planned in advance in terms of project specifications (including task schedule, stage gate dates, resource allocation, performance measurement, etc.), financial justification, and preliminary market and technical assessment (Brown 1995). However, ECs occur in far more random patterns compared with regular NPD activities, and the amount of time and effort required for each EC also varies significantly from one case to another. Simple changes to the manufacturing specifications of a product component may need just a few days while other changes to the outcomes of activities in early design lifecycle stages may cause unexpected downstream change propagation, and result in substantial resource consumption, a high EC cost, and a long overall EC processing time.

Third, resources committed to an NPD project are normally pre–determined and stable. That is, a certain amount of resources are dedicated to each NPD project as stated by the proposed resource planning. However, despite the fact that ECM requires an integrated effort from project planning, sales and marketing, research and development, engineering, manufacturing, purchasing and inventory control, quality assurance/control, finance, human resources, and sometimes even suppliers (Huang and Mak 1999; Bhuiyan, Gatard, and Thomson 2006), there are typically no separate cross–functional resources set aside for handling ECs (Huang and Mak 1999). If there are no additional resources available when an EC Request (ECR) is approved, it has to compete for the same resources that have already been assigned to regular NPD activities according to priority levels.

Lastly, besides the above–mentioned primary effects on budget and schedule overruns, the nonlinear cause–and–effect relationships among ECs and regular NPD activities also cause secondary feedback effects on the scope, uncertainty, productivity and quality of an NPD project. Most of them are extensively recorded in literature of Product Development (PD) modeling utilizing a System Dynamics (SD) approach from a macro level with high abstraction. For example, fourteen secondary impacts of changes in construction development projects were identified by Thomas and Napolitan (1994), including decreased worker productivity, learning curve associated with a change, possible out–of–sequence work, increased planning, coordination and rescheduling activities, among others. However, these dynamic secondary feedback effects are not generally incorporated into traditional PD process–oriented discrete event simulation models that are typically constructed under a lower abstraction level as compared with SD.

In sum, ECM is an important aspect to the success of an NPD project. On one hand, it continuously improves products, services, or processes by solving safety and critical functionality problems of a product solution and/or reflecting new customer requirements and technological advances. On the other hand, it also unexpectedly consumes a considerable amount of product development resources, which in turn affects the lead time and productivity of regular NPD activities significantly and thus causes scheduling instability and dramatic project cost increment.

Despite its importance, there are only a few analytical models of ECM exist (e.g., Hegde 1992; Ho 1994; Balakrishinan and Chakravarty 1996; Ho 1997; Barzizza 2001; Bhuiyan et. al 2006; Lin et al. 2008), yielding inadequate ECM strategies for better PD project performance. This research aims to contribute to knowledge on the mutual impacts of ECM and NPD processes by designing and implementing a discrete–event simulation model, and applying it to investigate different NPD and ECM strategies and coordination policies.

1.3 The Context

ECM problems cannot be studied in isolation. But rather, investigation of ECM reveals that problems need to be addressed within a broader context, including the following three principle aspects: i) complex systems, ii) current engineering and uncertainty, and iii) rework and change propagation.

1.3.1 Complex Systems

The stochastic dynamics of ***Complex Systems*** have been studied in various disciplines from natural sciences (physics, biology, chemistry, etc.), social sciences (sociology, psychology, economics, etc.), to interdisciplinary and applied sciences (computer science, engineering, etc.). To capture the universal properties of a complex system, we must understand, not only qualitatively but also quantitatively, the behavior of its interconnected building blocks, and how these parts interact with each other to form a collective macro behavior of the whole (Bar–Yam, 1997). Particularly, the complex systems theory has been gradually accepted as an appropriate context to fit into the product development and management literature (Yassine and Braha 2003; McCarthy et. al 2006; Braha and Bar–Yam 2007; Levardy and Browning 2009).

A new product is designed and developed via an NPD process through the efforts from a group of specialists under dynamic internal and external environment. This dissertation brings together the four main elements of complexity associated with design and product development (Earl, Johnson and Eckert 2005), namely, product, process, team (/designer), and environment (/user), on the decision of how iterations and ECs emerge and thus impact NPD project performance, and how should they be effectively managed by applying different coordination policies. *Figure 1* describes the four main elements of PD project complexity by listing the corresponding contributors under each category. Interdependencies among these factors and how they contribute (i.e., whether positively or negatively) to the occurrence of ECs, long EC lead time, and high EC cost will be discussed in *Chapter 3* with the help of constructing causal/causal–loop diagrams.

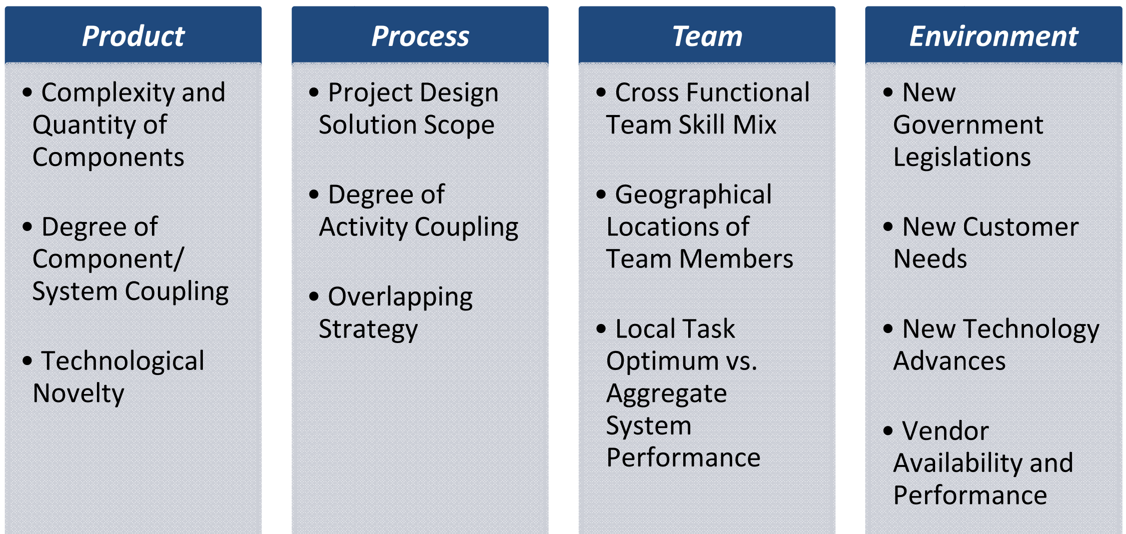


Figure 1: Four Basic Elements of PD Project Complexity

Highly engineered *product* is a complex assembly of interacting components (Hobday 1998; Krishnan and Ulrich 2001). In automobile industry, a fairly typical modern vehicle is composed of more than ten thousand manufactured component pieces, supplied by thousands of outside suppliers. In the face of such great quantities of components, complex products are impossible to be built all at once. They are decomposed into minimally coupled major systems, and then further broken into smaller sub–systems of manageable size and complexity, and finally down to separate components or parts for individual detailed engineering design. On the other hand, the integration of interdependent decompositions within and across system(s) into the final overall solution as well adds up to the level of complexity and requires substantial coordination efforts (Pimmler and Eppinger 1994).

Similarly, a large complex PD *process*, through which all the stages of a product's lifecycle occur, is itself a complex system involving hundreds or thousands of interrelated or interacting activities which transforms inputs into outputs (INCOSE SE Handbook V3.2.2, p. 5). As shown in the PD literature, tremendous research effort has been devoted into exploring the complexity of PD processes, especially in studying both of the advantages and disadvantages of parallel development process (also known as concurrent engineering) or spiral development process (which is applied more often in software industry) as compared with the traditional staged (also known as waterfall or sequential) development process. Some prior research particularly stressed structuring and managing the process through the efforts of minimizing the interdependencies among tasks via process sequencing optimization (Smith and Eppinger 1997, Browning and Eppinger 2002; Cho and Eppinger 2005).

Also, multi–disciplinary *teams* participating in an NPD project are typically composed of numerous decision makers from different functional areas (e.g., marketing, engineering, manufacturing, purchasing, quality assurance, etc.) with varied skill sets (e.g., degree of specialization, depth of knowledge, qualifications, work experience, etc.), responsibilities, and authorities working together and contributing to the achievement of the final product solution. These teams exhibit another set of complex and non–linear organizational behaviors in communication, collaboration, and integration when considering local task decisions as well as task interactions in determining aggregate system performance (Loch, Mihm and Huchzermeier 2003).

Last but not least, an NPD project interacts with its internal (e.g., simultaneous concurrent development of other products within the same organization) and external (e.g., customers/market, competitors, suppliers, and other socio–economic factors such as government regulations, etc.) *environments* throughout the project cycle. The dynamic and sometimes even chaotic competitive environmental factors also contribute significantly to the complexity in the coordination of NPD projects.

1.3.2 Concurrency and Uncertainty

Besides the above mentioned four essential ingredients of a complex PD project that will be explicitly integrated into the simulation model proposed by this dissertation, two key PD process characteristics, concurrency and uncertainty, will also be captured.

The concept of ***Concurrent Engineering*** is characterized by 1) the execution of PD tasks concurrently and iteratively, and 2) the cross–functional integration through improved coordination and incremental information sharing among participating groups. It has been widely embraced by both academia and industry for the well documented advantages of NPD cycle acceleration, risk minimization by the detections of design errors in early stages, and overall quality improvement (e.g. Ha and Porteus 1995; Loch and Terwiesch 1998; Bhuiyan, Gerwin, and Thomson 2004). It is one of the primary process features that are captured and thoroughly analyzed by the model framework proposed in this dissertation.

Complexity drives ***Uncertainty***. Uncertainty is an inherent nature of NPD projects stemming from all aspects of complexity associated with efforts creating a new product as discussed above. The presence of inherent uncertainty in NPD processes is much greater and, interestingly, much more complicated than those in processes of other kinds (e.g., business or manufacturing processes), even though the latter also possess certain degree of inherent unpredictability. Types of uncertainty in engineering design include *subjective uncertainty* derived from incomplete information, and *objective uncertainty* associated with environment (Wynn, Grebici, and Clarkson 2011). Moreover, concurrent processing of NPD activities will further increase the uncertainty of an NPD project by starting activities with incomplete or missing input information. This research explicitly differentiates uncertainty into three types: i) low–level *activity uncertainty* represented by the stochastic activity duration, ii) medium–level *solution uncertainty* that dynamically calculates rework probability, and iii) high–level *environmental uncertainty* captured by the arrival frequency and magnitude of IECs.

1.3.3 Rework and Change Propagation

Evidences show clearly that excessive project budget and schedule overruns typically involve significant effort on rework (Ford 1995; Ford and Sterman 1998, 2003; Reichelt and Lyneis 1999; Park and Peña–Mora 2003; Lin et al. 2007; Lyneis and Ford 2007). Moreover, it is claimed by Reichelt and Lyneis (1999) that “these phenomena are not caused by late scope growth or a sudden drop in productivity, but rather by the late discovery and correction of rework created earlier in the project.” In this dissertation, primary features of NPD projects will be transformed into a simulation model to study their relative impacts on the stochastic arrivals of ***Rework*** (i.e., iterations or EECs).

Rework probability, if included in previous PD process models, is typically assigned a fixed number and remains statically along the process. However, it is calculated in this model by the dynamic, evolving solution uncertainty that includes important feedback effects from other interrelated system variables such as design solutions scope, resource availability, etc. And also, any type of rework is usually discussed on an aggregate level, instead of being categorized into iterations or EECs, and even expanded to include IECs by this dissertation.

A change rarely occurs alone and multiple changes can have interacting effects on the complex change networks (Eckert, Clarkson and Zanker 2004). ***Change Propagation*** is included in this research by considering both of the interdependence of product components/systems and the interrelated NPD activities. A complex product usually consists of several interrelated major systems, and each further contains interconnected subsystems, components, and elements. The interactions, in terms of spatial, energy, information, and material (Pimmler and Eppinger 1994), that occur between the functional and physical elements will cause EC of one product element propagate to the others. Besides highly dependent product configuration, product development activities are also coupled. An EC may propagate to its later activities within the current phase or after. For example, an EC that solves a design fault may trigger further changes to downstream activities in design or production phase.

To conclude, this research is discussed in the context of complex systems and different forms of uncertainties on the decision of how NPD iterations, ECs, and change propagations emerge; their impact on key performance indicators, lead time, cost, and quality; and how should they be effectively managed applying different coordination policies.

1.4 Research Objectives

On the one hand, even though the demand has increased for more effective ECM as an important competitive advantage of product development companies, the existing ECM literature focuses mainly on the following topics: i) multi–step administrative evaluation that supports the formal EC approval, implementation, and documentation process, ii) ECM in product structure and material resource planning, and iii) change propagation and knowledge management. In addition, with a few exceptions (Hegde 1992; Ho 1994; Balakrishinan and Chakravarty 1996; Ho 1997; Barzizza 2001; Bhuiyan, Gatard, and Thomson 2006; Lin et al. 2008) (see *Section 1.5* for detailed discussion of these analytical or computer models), almost all the previous research or empirical studies were qualitatively discussed in a descriptive nature.

On the other hand, despite of a rich body of concurrent engineering literature that emphasizes the iterative nature of NPD, "these models see iterations as *exogenous* and *probabilistic* and do not consider the source of iteration[1]" (Loch, Mihm and Huchzermeier 2003), which causes the identified rework too general and therefore not sufficient for an effective ECM study. As a result, there is a lack of research–based analytical models to enhance the understanding of complex interrelationships between NPD and ECM, especially from an enterprise–level systems perspective.

In response to the increasing calls to close the gap between these two bodies of literature, the objective of this research is to conceptualize and integrate the key features of both NPD and ECM in a way that understanding and knowledge of the dynamic and mutual impacts between these two processes can be improved from a systems perspective. Recognition of two types of rework (i.e., iterations and EECs) and IECs, along with the evolving uncertainty levels of an NPD project that calculate rework probabilities and influence how the development process unfolds (Wynn, Grebici, and Clarkson 2011), are the two underlying problems to be addressed by this work.

To be more specific, this research intends to achieve the following goals:

[1] "Iteration" in this quotation is an equivalent term to "Rework" as defined by this research.

1) To conduct a comprehensive, in–depth study of the main characteristics of ECM problem both *qualitatively* (through field survey research to investigate the current practice and the construction of causal frameworks to enhance the understanding of causes of ECM problems and interdependencies among key process features) and *quantitatively* (through the generation and systematic investigation of computer models to give precise and testable results).
2) To develop a simulation model of the overall NPD process in which stochastic iterations, EECs, and IECs occur according to dynamically evolving uncertainty levels and thus impact work flow of the NPD project. Furthermore, this model framework can be extended into a multiple NPD projects environment.
3) To examine how changes in the model variables affect key project performance measures (i.e., lead time, cost, and quality of the NPD project) from a systems perspective. Different NPD and ECM managerial strategies and coordination policies are to be investigated.

1.5 Methodology

The general research methodology and associated underlying principles (in terms of qualitative or quantitative research design) adopted by this research are introduced in the first subsection, while the justifications of modeling methodology choices are provided in the second subsection.

1.5.1 Research Methodology

Figure 2 depicts the iterative research process that contains four main building blocks of this dissertation: i) *conceptual causal framework* (*Chapter 3*) aiming to find sources of ECM issues by identifying important interacting variables and their causal relationships from a systems perspective, ii) *field survey study* (*Chapter 4*) conducted in automobile and Information Technology (IT) industries in the summer of 2010 and 2011 to collect information and data regarding NPD and ECM processes, iii) *computer simulation model* (*Chapter 5*) that are systematically constructed based on the findings of the above two (i.e., theoretical and practical reasoning); and iv) *numerical application and result analysis* (*Chapter 6*) by importing educated estimates of model parameters, evaluating and comparing of various scenarios to support effective decision analysis of different NPD/ECM coordination policies and managerial strategies.

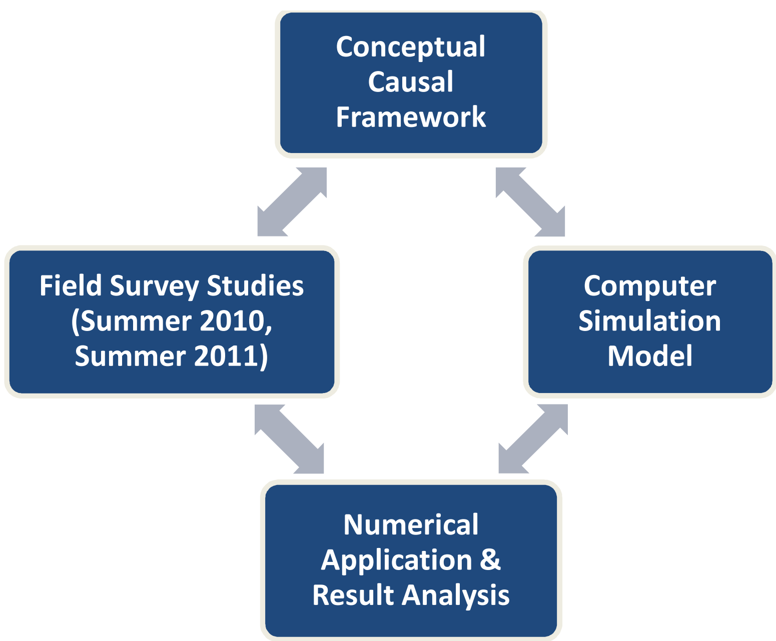


Figure 2: Overview of the Iterative Research Process

Figure 3 illustrates the detailed view of the research method design adopted by this dissertation, which is based on a combination of both qualitative and quantitative analyses. The *qualitative* analyses include:

- Analysis of primary and lower level drivers of ECM issues (e.g., long EC lead time and high EC cost) together with the causal relationships among these factors;
- Analysis of information obtained from observations and informal interviews;
- Use of secondary data in related literature, such as tested or proven definitions, theories, research hypotheses, etc.

The *quantitative* analyses include:

- Analysis of companies' historical data or data collected from structured interviews;
- Experimental design and result analysis of the computer simulation model;
- Use of secondary data in related literature, such as published official statistics, results from field studies and industry surveys, etc.

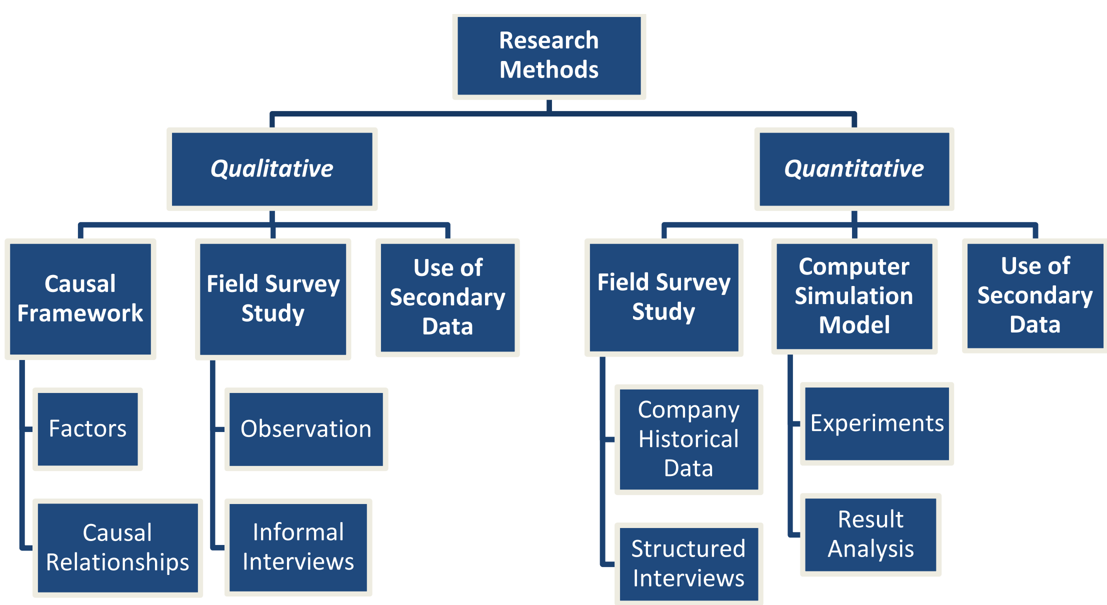


Figure 3: Detailed View of Research Method Design

1.5.2 Comparison of Different Modeling Methodologies

As listed in *Table 1*, there are two main directions, mathematical modeling and computer simulation, that previous researchers took to gain insights and knowledge about NPD and ECM processes in the existing literature. It is important to note that these two approaches are by nature interwoven since computer simulation is innately mathematical models but in a computer–assisted representation.

Formulating a mathematical model, which is to "represent a system in terms of logical and quantitative relationships that are then manipulated and changed to see how the model react, and thus how the system would react" (Law 2007), is one way to define and abstract the problem of interest. Among various algorithm approaches, *linear programming*, which objective function and constraints are all linear functions, is fit to solve "the general problem of allocation limited resources among competing activities in the best possible way" (Hillier and Lieberman 2001). As listed in *Table 1*, several researchers applied linear programming in their studies (Balakrishnan 1996, Krishnan 1997, and Barzizza 2001).

Since the time wasted by waiting in lines for limited servers/resources is one of the major factors in both the long lead time and the low production rates of NPD and ECM, classical *queueing theory* can be considered as another reasonable mathematical representation. By applying queueing formulas using different probability distribution for inter–arrival and service times, average waiting time and number of entities in queue can be obtained to measure the performance of the queue. However, mathematical analyses of queuing network problems could become too complex when the feedback loops among interrelated processes are considered.

Table 1: Modeling Methodology Summary

Reference	Purpose	Description
Mathematical Modeling (Analytical Solution)		
Hegde 1992	Statistical analysis to quantify the impact and interaction of various time drivers for ECO on shop delays.	• Empirical analysis of descriptive statistics • Single/Multiple variable(s) regression of idle time–in process (queue time)
Balakrishinan and Chakravarty 1996	An analytical optimization model to investigate the impact of an EC on market opportunities and manufacturing costs when deciding	• Linear programming • Objective function: maximize *revenues* and minimize *total cost (backorders, subcontracts, inventory holding*, and *obsolescence)*
Ho 1997	An analytical procedure to compute progressive probabilities of ECs.	• Equation for calculating the progressive probability of EC for each item • Sensitive analysis
Krishnan et. al 1997	A mathematical model of an overlapped NPD process using evolution and sensitivity to identify overlapping strategy for optimal product development performance.	• Linear programming • Objective function: minimize *development lead time* $\lambda = t_n + d_n$
Barzizza 2001	A mathematical model aims at suggesting use–as–is ECs' implementation at the best time, with the least impact on firm costs.	• Linear programming • Objective function: maximize *total saving* S_N resulting from the production of *N* units of pre–change product in place of post–change product.
Bhuiyan 2001	A mathematical technique for studying and evaluating the performance of a concurrent process and a sequential process considering overlapping and functional interaction.	• Expected Payoff Method (Decision Theory) in the form of a quadratic function • No rework and no interaction between phases in a sequential process as simplifying assumptions
Computer Simulation		

Table 1: Modeling Methodology Summary (cont'd)

Ho 1994	A simulation experiment to examine the effect of different frequencies of ECs on the performance of multi–level Material	• Simulation experiment • Analysis of variance (ANOVA) of *total cost* and *obsolescence cost*

	Resource Planning system under various operating environment.	
Bhuiyan et. al 2004	A stochastic computer model to study concurrent engineering and how the key features of overlapping and functional interaction affect development time and effort under four uncertainty conditions.	• Discrete–event simulation • Information–process view of NPD • Three types of rework: churn, design versions, and overlap spins • Rework probability is pre–determined by two model variables: overlapping and functional interaction
Cho and Eppinger 2005	A process modeling and analysis technique to compute the probability distribution of NPD lead time in a stochastic, resource–constrained activity network where iterations take place sequentially, in parallel, or in an overlapped fashion.	• Latin hypercube sampling for duration sampling • Parallel discrete–event simulation • Streamlined interface between information–based Design Structure Matrix structuring analysis and network –based project scheduling analysis • Iteration probabilities and rework amount vary in each iteration
Bhuiyan et. al 2006	A stochastic computer model to compare the behavior of two methods of managing an ECR process, individually or in a batch.	• Discrete–event simulation • Based on the framework developed in Bhuiyan et. al 2004 • ECRs only go to the start of the present or any previous phase • ECRs and design versions have the same probabilities of occurrences
Lin et al. 2008	A dynamic development process model for managing overlapped and iterative product development based on the well accepted and validated "Rework Cycle" framework.	• System dynamics simulation • Rework due to development errors and rework due to corruption • Overlapping and investment policy analysis • Model validation using real world data

Although there is no particular study in literature that adopts the queuing theory mathematically, it is integrated within almost all of discrete–event simulation models. These simulation packages allow the construction and statistical analysis of complex queuing network problems.

This dissertation uses computer simulation to model and study the dynamics between NPD and ECM. Simulation has several advantages over other approaches. First of all, To gain insights into the operation of a very complex and dynamic real world system without too much over simplification, computer simulation appears to be a more effective and powerful tool than pure mathematical approach that is often from a single viewpoint. While a computer simulation is based on some mathematical algorithms, very complex modeling of stochastic inputs and detailed operations are possible. Second, compared to optimization models, simulation is especially valuable to identify how feedback effects, nonlinearities, and delays interact to produce dynamics that persistently resist solution (Sterman 1991). Third, simulation models can easily incorporate separate random inputs that follow almost any desired probability distribution for model replications, thus enabling a more valid representation of reality. Lastly, computer simulation provides better control in comparing alternatives and scenarios by changing the model structure and parameter settings. This feature gives simulation superiority in the investigation of different managerial strategies and coordination policies over other methodologies.

The *"information flow"* view of an NPD process (Clark and Fujimoto, 1991; Krishnan, Eppinger, and Whitney 1997) is adopted by this research. From this information processing perspective, an NPD project is considered as an evolutionary process with disaggregate design information being generated, transformed, and converged into the final product solution, proceeding through time and across functional areas. However, we are not interested in how the initial inputs in terms of market opportunities or new product ideas are continuously evolving into the eventual deliverable, but rather in those discrete points in time (t) when entities of the system (i.e. an NPD project or IECs) start or finish an activity and the corresponding change of the state of the system. At discrete time (t), duration of each activity, functional resource consumption from all involved departments, current value of the solution uncertainty, and real time work flow will be captured.

Also, the repeatable nature of an NPD process provides the validity for decomposing an NPD process into successive design and development phases, each containing several sequentially repeating activities. Nevertheless, it is important to note that NPD is typically an iterative process rather than a purely linear one, with unforeseen uncertainty and ambiguity (Terwiesch and Loch 1999). This feature can be represented by the routing of work flow back to those already completed activities in the form of iterations and EECs in this model.

1.5.3 Justification for Utilization of Discrete Event Simulation

Among various kinds of computer modeling approaches, a dynamic, stochastic ***Discrete Event (DE)*** simulation, which is based on the concept of entities, resources, queues, and block charts describing entity flow and resource sharing (Borshchev 2004), has been employed over others approaches, such as *System Dynamics (SD)* modeling and *Agent–Based (AB)* modeling, for the following reasons:

1) The *abstract level scale* of DE modeling is able to meet the requirements of the problem in discussion. DE modeling is capable of presenting the NPD and ECM process structure

as an activity/queuing network that accounts for precedence relationships among activities.

2) DE is also flexible in modeling *variability among individual components* compared with SD approach. Differentiation of activity modules is achieved by assigning different duration and resource requirement while differentiation of entities is resulting from assigning different processing and routing with different priority.

3) User–defined individualized attributes and global variables can be incorporated to further reflect peculiar characteristic of the process, which add up the capabilities of creating *cause–effect feedback loops* among variables and the occurrences of events to describe the dynamic flow of work within a DE model.

1.6 Organization of Thesis

This dissertation is organized in seven chapters as follows.

Chapter 2 extensively reviews the literature along two main directions: i) engineering change management, and ii) process modeling and simulation of NPD and ECM. In particular, three influential modeling approaches of product development process that highlight the effects of iterations and overlapping are discussed in detail.

Chapter 3 presents the conceptual causal framework of four major ECM problems: occurrence of ECs, long EC lead time, high EC cost, and occurrence frequency of iterations and ECs. Open–loop causal diagrams and closed causal feedback loops are created to determine the key contribution factors to these ECM problems and interdependencies among them from a systems perspective.

Chapter 4 presents several field studies conducted in the summer of 2010 and 2011 in automobile and IT product/service industries, based upon which the need for improved modeling effort toward ECM was identified.

Chapter 5 introduces the building blocks of the model framework and logics behind each model variable in detail. The discrete event simulation model includes two major components: NPD section with rework, and IEC section.

The proposed simulation model is then illustrated in its entirety by a 3–phase and 3–activity example in *Chapter 6*. It is followed by the experimental control and manipulation of model variables, together with analysis and evaluation of running results.

Chapter 7 discusses research and managerial implications of this work, and presents conclusions, restrictions, and future work of this research.

CHAPTER 2

LITERATURE REVIEW

2.1 Overview

An extensive search of the literature is conducted along two main directions:

1) ***ECM***, and
2) ***NPD Process Modeling*** for project management.

Due to the fact that the former has received much less attention from research and industrial communities than the latter, different search and review strategies are applied to the two categories. A comprehensive survey was conducted to broadly cover major ECM topics, followed by a detailed review of only several highly–cited influential theories and NPD models proposed in literature that recognize process features critical to this dissertation work, including *concurrent engineering*, *rework and iterations*, and *uncertainty*.

Under each category, related papers are further grouped into various topics as shown in *Table* 2 and *Table 3*, respectively. Nearly 50 papers[2] were reviewed thoroughly, among which ninteen core references are shown in **bold** formatting. Because of the content overlap, papers may appear in more than one topic.

[2] This number doesn't indicate the entire length of the references cited in this dissertation.

Table 2: List of Papers Covered in *ECM* Literature Review

General Administration Guidelines	Huge 1977; Diprima 1982; Reidelbach 1991; Balcerak and Dale1992; Hegde, Kekre, and Kekre1992; **Wright 1997** (*Paper Review*); Huang and Mak 1999; **Loch and Terwiesch 1999**; **Terwiesch 1999 and Loch**; Barzizza 2001; Huang, Yee, and Mak 2003; Tavcar and Duhovnik 2005; Klein, Poltrock, and Handel 2007
ECM in Product Structure and Material Requirements Planning	Harhalakis 1986; Maull, Hughes, and Bennett 1992; Ho 1994; **Balakrishnan and Chakravarty 1996**; Ho and Li 1997; Rutka et al. 2006; **W□nstr□m and Jonsson 2006**
Change Propagation and Knowledge Management	Saeed, Bowen, and Sohoni 1993; Ho and Li 1997; Peng and Trappey 1998; Clarkson, Simons, and Eckert 2001; Do 2002; Rouibah 2003; **Eckert, Clarkson, and Zanker 2004**; **Keller, Eckert, and Clarkson 2005**; Bouikni and Desrochers 2006; **Lee, Ahn, and Kim 2006**; Aurich and R□βing 2007; Do, Choi, and Jang 2007; Scholz–Reiter et al. 2007
Computer Aided ECM System	Huang and Mak 1998; Huang, Yee, and Mak 2001; Chen, Shir, and Shen 2002; Rouibah and Caskey 2003; **Lee, Ahn, and Kim 2006**

Table 3: List of Papers Covered in *NPD Process Modeling* Literature Review

General Analytical Frameworks		**Krishnan, Eppinger, and Whitney 1997**; **Browning 1998**; Loch and Terweisch 1998; **Bhuiyan 2001; Browning 2006&2007** (*Paper Review*)
Models /Simulation Methodologies	*Design Structure Matrix*	**Browning and Eppinger 2002; Cho and Eppinger 2005**
	System Dynamics Model	**Black and Repenning 2001**; **Lin et al. 2008**
	Discrete Event Model	**Bhuiyan, Gerwin, and Thomson 2004**; **Bhuiyan, Gatard, and Thomson 2006**

2.2 Engineering Change Management

Papers related to the topic of Engineering Change Management are further divided into four categories: i) *General Administration Guidelines*, ii) *ECM in Product Structure and Material Requirement Planning*, iii) *Change Propagation and Knowledge Management*, and iv) *Computer Aided ECM System*.

2.2.1 General Administration Guidelines

Providing generic descriptions of the problem and making suggestions for effective Engineering Change Control (ECC), thus minimizing EC impact is one of the traditional characterizations of ECM research.

Huge's paper is among the earliest contributions to the ECM fields (Huge 1977). He presented some key ECC concepts including degree of control, change evaluation process, EC incorporation point and effectiveness, ECC procedures in different phases within the product life cycle, the engineering/manufacturing interface, change planning and implementation requirements.

Diprima (1982) developed a framework for proper control and implementation of EC. It contains the steps from EC initiation, approval, implementation, to the final stage which is EC follow–up. Diprima pointed out several essential principles in ECC system including i) the importance of communication, ii) establishment of an EC committee composed of individuals from marketing, engineering, finance, etc., iii) category of ECs: immediate, mandatory, and convenience, iv) cost analysis to determine how an EC should be implemented, v) responsibilities of an EC coordinator, and vi) a checklist prepared for every EC.

Reidelbach (1991) categorized ECs into three groups: i) early, low impact ECs, ii) mid–production ECs, and iii) late, expedited ECs. The author made suggestions on minimizing impact of ECs such as negotiations between customers and suppliers, weeding out undesirable changes, expediting if shortage exists when an EC is authorized, forecasting, and aggregate planning. Despite the effort an EC committee can make, what Reidelbach observed from the real world EC practice also revealed the fact that typical production environment is unpredictable with uncountable variables of pace and human inconsistencies. And operation management always has to face the reluctance to change. To conclude, the author listed 11 guidelines for the management of EC.

Another review of EC fundamentals was conducted by Balcerak and Dale (1992) through a field research. There are several outcomes that are worth mentioning. *First*, previous classification scheme emphasizes too much on the documents affected such as drawing and/or bill of material. The author redefined three EC types in terms of finished components and assemblies to indicate the impact of the change: ECs involving components only, ECs involving assemblies and components, and ECs involving assemblies only. *Second*, the urgency with which a change should be processed, which is defined as EC grade, can be classified into *Grade E* (error correction changes), *Grade M* (mandatory changes), and *Grade P* (phased–in changes). Type and grade can be combined together to evaluate an EC. *Third*, more than one of the six determinants of EC effectivity, i) market forces, ii) drawing office work, iii) availability of replacement parts or raw material, iv) stock run out, v) availability of replacement tools, and vi) tool wear out, need to be considered when deciding the optimum effectivity date. *Forth*, feedback from manufacturing areas is essential to the success of an EC procedure.

Hegde, Kekre, and Kekre (1992) investigated impacts of ECs from *"time drivers"* perspective through a field study in a Fortune 500 company. Based the empirical analysis, they provided two measures of the detrimental impact of ECOs: i) under single variable analysis, each ECO adds 21.88–day delay for a typical part on the shop floor and 22.61–day delay due to material defect; 2) when multiple regressions are conducted, same qualitative conclusions can be obtained. ECOs, defective materials, a route involving visits to bottlenecks, and releasing a job earlier than the planned date all have adverse impact on lead time. However, a close monitor on jobs that visit bottleneck operations will shorten the delay to a considerable extent.

A thorough review of papers until 1995 was done by Wright (1997). The author categorized the EC related papers into two main topics, computer–based *"tools"* for the analysis of EC problems and *"methods"* to reduce the impact of ECs on manufacturing and inventory control. Most of the publications during that time period predominantly focused on the EC control mechanisms and systems. An important observation by Wright is that understanding of the *positive* effect EC can provide for product improvement and enhanced market performance is long omitted by EC research.

Huang and his research group conducted two comprehensive questionnaire surveys on the topic of effectiveness and efficiency of the engineering change management system within UK and Hong Kong manufacturing companies in 1996 and 1999, respectively (Huang and Mak 1999; Huang, Yee, and Mak 2003). The surveys resulted in several observations. *First*, a well structured procedure instead of an ad hoc one is the most important element of an ECM system. *Second*, most of ECM activities are related to the administrative processing; design office. *Third*, industrial/production department, and EC coordinator are the most relevant functions for ECM within an organization. *Fourth*, the processing and implementation of ECs scores highest among strategies for ECM. *Fifth*, the majority respondents use CAD, MRP, and CAM for quick implementation of ECs. *Sixth*, poor communication and late discovery of problem were found to be the two most significant influential factors of ECM to respondents. Their study pointed out the correlation between company size and scope of ECM practices. They also suggested adopting computer support packages and international standards for the establishment of ECM procedure.

An analytical framework that explains the extreme ratio between theoretical processing time and actual lead time was developed by Loch and Terwiesch (1999). They showed how *congestion and batching* influence engineering processes at a more detailed level. Based on the processing network framework, they suggested improvement strategies such as flexible work times, the grouping of several tasks, workload batching, the pooling of resources, and the reduction of setup times.

Terwiesch and Loch (1999) presented a process–based view of ECM. They showed by an industrial case study that a complicated and congested administrative support process is one of the root causes of long lead time and high cost. Based on the field study, they identified five key contributors to lengthy ECO lead time: i) complex ECO approval process, ii) scarce capacity and congestions, iii) setups and batching, iv) snowballing changes, and v) organizational issues.

Barzizza (2001) suggested a new methodology for EC implementation. ECs are classified into three categories, scrap, rework, and use–as–is. EC implementation date and costs are then listed for each kind. The authors also suggested two control points to assure a good dynamic ECM: the *costs control point* and the *time control point*. "Cost control point" indicates the average percentage error in defining EC costs while "time control point" shows the average delay of EC implementation.

Tavcar and Duhovnik (2005) recognized i) concurrent engineering methods, ii) process definition, iii) information system, iv) communication, and v) organization as the five key factors for efficient ECM. These factors were used for optimizing the EC process in individual production, serial production of modules, and manufacture of household appliances composed of elements and modules provided by different suppliers. The authors suggested that different products with varied degree of complexity, interdependency, and number of involved production fields should put different emphasis on the five criteria for effective ECM. In order to yield an optimum decision–making process, they recommend a combination of communication via electronic media and personal consultations, prototyping, easy access to both technical and manufacturing data on the product by internal personnel and external suppliers as well, and recognition of the design level of EC.

Coordination theory is about the collaborations among people or software agents to manage the dependencies between tasks. Klein, Poltrock, and Handel (2007) demonstrated an approach for recognizing the similarities and differences among three ECM processes from a coordination–theoretic perspective. They first defined core tasks of the change process to be propose change, authorize change, and implement change. The key dependencies are a change request flow from the first task to the second and an authorizing change notice flows from the second task to the third. Then he compared three EC processes that manage changes to cost and schedule, processes and tools, and product configuration by applying top–down derivation trees. Two key findings were obtained: i) most of the steps in these processes involved coordination; ii) the differences between processes concerned how they perform coordination and exception handling.

2.2.2 ECM in Product Structure and Material Requirements Planning

There are number of researchers examine ECM problems from the perspective of material requirements planning. Research questions include: how do ECs affect the stability of production planning and inventory control? How many items are required to meet EC demand while how many get obsolete? Which lot–sizing rule should a company follow to maintain the lowest possible cost for the production in progress whose design gets changed?

Maull, Hughes, and Bennett (1992) wrote a paper on the topic of how ECs affect the *stability of the Bill–of–Materials* (BoM), especially the effects of such changes on the computer–aided design (CAD)/computer–aided production management (CAPM) interface.

Ho (1994) raised the question of how to balance the frequency of ECs and *scheduling instability* it causes. He showed that frequent ECs deteriorate MRP system performance through a full factorial simulation experiment along with a sensitivity analysis for validation. Also, the choice of *lot–sizing rule* was found important under different conditions of EC frequency in terms of obsolescence cost and total cost. The experimental factors include: EC frequencies, lead time uncertainty, lot–sizing rule, and the inventory items' setup/carrying cost ratio. The ANOVA analysis indicates that Silver–Meal discrete lot–sizing heuristic (SM) appears to be the best rule when probability of ECs (p) is less than 0.5%. When p exceeds 0.5%, the part–period balancing performs better. Economic order quantity is the worst rule under all levels of EC frequencies. SM and least total cost rule were found to be comparatively sensitive to the length of planned lead time. In a frequent EC situation, obsolescence cost increases in a great degree and selection of lot–sizing rule becomes more important.

In another paper by Ho, an analytical procedure to compute progressive probabilities of ECs was developed (Ho and Li 1997). Progressive EC probabilities are calculated for every component in multi–level product structure in terms of the impacts of part commonality and structures of BoM on EC. They concluded that both the magnitude of EC and the number of immediate parents impact the progressive probability of EC for an item greatly while the depth of product structure has no significant impact.

By providing an analytical optimization model, Balakrishnan and Chakravarty (1996) investigated the impact of an EC on market opportunities and manufacturing costs. They showed the advantage of phasing in the enhanced new product over a period of time to gain more marketing opportunities than replacing the existing old product immediately. And they further showed how the optimal cut–in and cut–out periods are affected by NPD lead time, product market attractiveness as compared to the old product, capacity availability, subcontracting premiums, and backorder costs.

W□nstr□m and Jonsson (2006) conducted a very comprehensive study of the EC impact on *Materials Planning* (MP) process by carrying out a field investigation of three tiers in the supply chain at an automotive company. They started by analyzing the EC situation for the MP through recognizing the characteristics of EC, demand, product, manufacturing, material supply, and MP process and relationships between them. Interchangeability was identified as one of the most crucial characteristics for efficient EC process. Also, product structure in terms of complexity of BoM, high customer service requirements, high demand uncertainty have a negative impact on material scrap costs. Strategy of changing the lot–size 12 days before the phase–out day was found to result in increasing administrative costs and decreasing materials scrap costs differently for different suppliers.

2.2.3 Change Propagation and Knowledge Management

From *Table 2* we can observe a trend of exploring change propagation and knowledge management in the ECM research community. This is a field that combines different disciplines such as engineering, information technology, business administration, and etc. Due to the interconnected nature of product components, the execution of an EC to one part may propagate and cause other product items to change as well. Proper identification and control on change propagation are recognized as critical to ECM, which can be achieved by undertaking explicit knowledge management before, during, and after an engineering change activity.

Saeed, Bowen, and Sohoni (1993) raised a research topic on avoiding ECs through *Focused Manufacturing Knowledge* (FMC), which was defined as the knowledge that an engineer would develop by working in an existing area of manufacturing most related to that engineer's development task. Though the authors didn't come to firm conclusions about providing how much FMC would yield optimum benefit to product development companies, they raised several interesting research questions that are worth further investigation: What are the different types of EC–related costs, both internal and external to the organization? Are there EC process design strategies that result in high degrees of both control and efficiency? How can an organization design and improve an EC process with the goal of effectiveness? What types of knowledge impact the engineering change process, how can they be developed, and what are the tradeoffs in their development?

Peng and Trappey (1998) described an *Engineering Data Management* (EDM) system that consists of six data models: i) product definition, ii) product structure, iii) shape representation, iv) engineering change, v) approval, and vi) production scheduling. EC and approval process was demonstrated through an example for a new version of pencil assembly. However, this prototype model remains as a concept framework without practical implementation.

Based on an empirical study undertaken with GKN–Westland helicopters, Clarkson, Simons, and Eckert (2001) proposed a *Change Prediction Method* (CPM) that uses product data and a model of change propagation to find components relationships and to calculate the risk of direct and indirect change propagation. CPM consists of three steps: i) creating the product model, ii) completing the dependency matrices, and iii) computing the predictive matrices. *Design Structure Matrices* (DSM) are used to present the interconnectivity of the product components and change relationships (in terms of likelihood and impact). Detailed historic case studies of the product model EH101 were then used to validate CPM on fuselage additional items (FAI) changes, equipment and furnishings (E&F) changes, and weapons and defensive systems (W&DS) changes. A high level of agreement between the predicted likelihood and observed results provided support for the CPM method.

In another journal paper published by the same research group, Eckert and his co–workers provided a very thorough case study on the topic of *change and customization* in complex engineering product that is conducted in Westland Helicopters Limited (Eckert, Clarkson, and Zanker 2004). Based on the information gathered from 22 interviews and several documented EC scenarios, the author distinguished sources of change into two main categories: i) *emergent change* caused by state of design, and ii) *initiate change* arising from an outside source. Then different kinds of initiate and emergent changes were located along the time axis. Initiate changes include customer requirements (before or after a contract has been signed), certification requirements, innovations, problems with past designs, and retrofits. Emergent changes include problems in design, in testing, in prototyping, in manufacturing, and in use. *Causes* for problems with change were detected as the following five reasons: i) different representations from different engineering disciplines for design ideas, ii) insufficient communication, iii) no decisions or wrong decisions due to the lack of technical knowledge or overview of the product, iv) insufficient clarification of the task which will lead to unnecessary repetition, and v) inadequate processes which lack appropriate methods. Four types of change propagation behaviors were differentiated corresponding to degree of absorption and propagation: *constants*, *absorbers*, *carriers*, and *multipliers*. They also identified two types of redesign: *forwards partial redesign* which is evaluated and executed orderly, and *backwards patching redesign* which jumps from problem to a solution in an unstructured way. In conclusion, understanding the state of the design, tolerance margins on key parameters, and connectivity between parts were suggested to avoid unexpected change effort.

Due to the fact that different stakeholders may be interested in different portions of the huge amount of product data but not all the details, Keller, Eckert, and Clarkson (2005) examined how multiple views can be used to display change propagation data for complex products. The author introduced the CPM software tool for ECM and adopted it for several industrial cases to show its usage in visualization of change propagation. By linking graphs from *Direct Risk Plot* that show change likelihood and impact values by DSMs to *Case Risk Scatter Plot* that represents the combined change risk, sensitivity analysis of combined risk values is allowed. A tree structure was suggested for the visualization of change propagation paths. Fisheye layouts are used for display of propagation networks.

Bouikni and Desrochers (2006) proposed a *Product Feature Evolution Validation* (PFEV) model for remaining information consistency among disciplines involved in an ECM process throughout the complete product life cycle. The PFEV model consists of three main parts: i) disciplines component, ii) product model component, and iii) *Product Feature Evolution* (PFE) analysis and distribution component. Evolving product information data will be distributed to each impacted discipline with specific views. Those detrimentally impacted disciplines then need to analyze the proposed PFE and negotiate if the PFE is not accepted by any one of them. Once those disciplines agree on the change, a PFE can be validated.

Lee, Ahn, and Kim (2006) conducted a field investigation of the new product development projects at a major Korean automobile company. Lee pointed out several problems that limit the accumulation of knowledge such as i) difficulties in capturing tacit knowledge (e.g., context information on knowledge items, collaborative experiences, and decision–making processes, etc.), ii) poor management and reuse of past experience, and iii) limited searching function by keywords or reference number. Based the findings from the case study, a model called *Collaborative Environment for ECM* (CECM) was developed to facilitate the accumulation and reuse of the knowledge generated in collaborative EC process.

Aurich and R□βing (2007) proposed an engineering change impact and similarity analysis to define EC projects. *Change Impact Matrix* captures the change impact between production elements and then yields the object impact factor of one particular EC for each element. And the sum of object impact factors is the change impact factor of that EC. Similarity matrix measures the similarity between two ECs by counting object impact factors that have a positive value for both ECs and their change impact factor. Two ECs that have change similarity above 0.5 are suggested to be grouped together.

Do’s research group proposed a product data model and an EC propagation procedure that can maintain consistency by propagating ECs in a base product definition to product data views

(Do, Choi, and Jang 2007). The product data views in the proposed model can share product structures in the base product definitions without copying them. EC propagation procedure, on the other hand, uses the change histories of the EC process, which also guarantees the consistency by propagating ECs in a base product definition to product data views.

Ramp–up phase was defined by Scholz–Reiter et al. (2007) as an interface between product development and production that includes process testing, the pre and zero series, and production ramp–up, in which numerous ECs take place. His research focused on ECM during this critical phase. They suggested clear and well–structured knowledge management and long–term preventive acting product change teams to implement.

2.2.4 Computer Aided ECM System

Despite the above mentioned research directions, efforts are also made in the development of integrated information system to help streamline the ECM process.

Based on the observation from a survey conducted among UK manufacturing companies, Huang and Mak (1998) discussed the reasons why computer aids not being widely used in ECM, especially for analytical tasks. Key characteristics of computer aided engineering change control system were pointed out under the categories of functionality, usability, flexibility and focus. Based on this comprehensive industrial survey, a *web–based ECM system* was constructed for better information sharing, simultaneous data access and processing, and more prompt communication and feedback (Huang, Yee, and Mak 2001).

Chen, Shir, and Shen (2002) proposed an *Allied Concurrent Engineering* (ACE) based ECM system methodology to manage and control the EC process, systems, and information in an integrated fashion. The methodology includes a life cycle model for ECM, a hierarchical and distributed management framework, and a reference model for ECM.

Rouibah and Caskey (2003) presented a *parameter–based approach* for ECM that aims to support *multi–company* concurrent engineering efforts. Their approach supports communication among all relevant parties, facilitates information sharing and use, and helps tracing of change propagation using the parameter network.

2.3 NPD Process Modeling

2.3.1 General Analytical Frameworks

Browning introduced the foundational concepts of *Product Development* (PD) process modeling from a systems engineering perspective, and compared various modeling views for decision support (Browning, Fricke, and Negele 2006; Browning and Ramasesh 2007; Browning 2009a; Browning 2009b). He argued that key characteristics of PD process include i) product development versus repetitive business processes, ii) descriptive versus prescriptive processes, iii) activities as actions versus deliverables as interactions, iv) standard versus deployed processes, v) centralized versus decentralized processes, vi) "as is" versus "to be" processes, and vii) multiple phases in product development (Browning, Fricke, and Negele 2006). Eighteen PD process modeling frameworks were reviewed including PERT/CPM, DSM, and IDEF. They also introduced a framework for modeling PD process that supports many purposes, such as scheduling, budgeting, resource loading, and risk management.

The comprehensive survey (Browning and Ramasesh 2007) reviewed nearly 200 research works and categorized them into four major groups based on modeling purpose: i) visualization, ii) planning, iii) execution, and iv) control, and project development. Five research directions were highlighted for future study: i) activity interactions, ii) global process improvements, iii) process models as an organizing structure for knowledge management, iv) modeling in cases of uncertainty and ambiguity, and v) determining the optimum amount of process prescription and structure for an innovative project.

Krishnan, Eppinger and Whitney (1997) presented a model–based framework to manage the overlapping of coupled product development activities (in terms of the pattern of information exchange) to maximize lead time, cost, and quality performance. The authors studied the overlapping problem based on two properties of the information exchanged between product design phases: i) *upstream information evolution* and ii) *downstream iteration sensitivity*. The mathematical model and conceptual framework of the overlapped process were illustrated with industrial examples to provide managerial insights such as: i) effect of overlapping should be the basis for disaggregating the exchanged information; ii) only those parts whose early freeze would produce very little quality loss should be frozen early; and iii) upstream information exchanged in a preliminary form should be chosen such that changes in its value may be absorbed without substantial increase in the downstream effort.

2.3.2 Detailed Review of Simulation Models Addressing Iteration

2.3.2.1 Design Structure Matrix

Design Structure Matrix (DSM) (Steward 1981; Eppinger, Whitney, and Smith 1994) is a simple yet compact representation of the elements (e.g., product decompositions, process activities, and cross–functional work teams/groups) in a complex system (i.e., NPD projects) and the interdependencies among them. According to the categorization by Browning (2001), *static* DSMs that represent system elements existing simultaneously (e.g., product or organizational architecture) can be analyzed through *clustering* algorithms (Pimmler and Eppinger 1994) while *time–base* DSMs that represent elements flowing through time (e.g., process activities, product parameters) are typically analyzed using *sequencing* algorithms (Smith and Eppinger 1997, Browning and Eppinger 2002; Cho and Eppinger 2005).

The most closely related works to the concurrent and iterative nature of PD process addressed by this research are the two generations of DSM–based simulation model (Browning and Eppinger 2002; Cho and Eppinger 2005) for optimal activity sequencing by analyzing the effect of iteration and overlapping. To re–sequence, a complete list of *activities* is needed beforehand as appeared in *Figure 4a*. So is the *rework probabilities* (probability of rework for an activity due to a change in another activity) and *rework impact* (percentage of activity to be reworked) in the form of DSM (*Figure 4b*).

This model was later expanded by Cho and Eppinger (2005) to include additional important process features such as resource constraints and rework concurrency. *Figure 5* illustrates the application of this process model by incorporating it into an integrated project management framework for managerial decision making.

Activities		Durations (days)			Costs ($k)			
ID#	Name	BCV	MLV	WCV	BCV	MLV	WCV	IC
1	Prepare UCAV Preliminary DR&O	1.9	2	3	8.6	9	13.5	35%
2	Create UCAV Preliminary Design Architecture	4.75	5	8.75	5.3	5.63	9.84	20%
3	Prepare & Distribute Surfaced Models & Int. Arngmt. Drawings	2.66	2.8	4.2	3	3.15	4.73	60%
4	Perform Aerodynamics Analyses & Evaluation	9	10	12.5	6.8	7.5	9.38	33%
5	Create Initial Structural Geometry	14.3	15	26.3	128	135	236	40%
6	Prepare Structural Geometry & Notes for FEM	9	10	11	10	11.3	12.4	100%
7	Develop Structural Design Conditions	7.2	8	10	11	12	15	35%
8	Perform Weights & Inertias Analyses	4.75	5	8.75	8.9	9.38	16.4	100%
9	Perform S&C Analyses & Evaluation	18	20	22	20	22.5	24.8	25%
10	Develop Balanced Freebody Diagrams & External Applied Loads	9.5	10	17.5	21	22.5	39.4	50%
11	Establish Internal Load Distributions	14.3	15	26.3	21	22.5	39.4	75%
12	Evaluate Structural Strength, Stiffness, & Life	13.5	15	18.8	41	45	56.3	30%
13	Preliminary Manufacturing Planning & Analyses	30	32.5	36	214	232	257	28%
14	Prepare UCAV Proposal	4.5	5	6.25	20	22.5	28.1	70%

		1	2	3	4	5	6	7	8	9	10	11	12	13	14
Prepare UCAV Preliminary DR&O	1														
Create UCAV Preliminary Design Architecture	2	.4								.2					
Prepare & Distribute Surfaced Models & Int. Arngmt. Drawings	3		.5		.4										
Perform Aerodynamics Analyses & Evaluation	4	.3		.5											
Create Initial Structural Geometry	5	.4		.5			.1		.1				.3	.1	
Prepare Structural Geometry & Notes for FEM	6	.1				.4									
Develop Structural Design Conditions	7	.4					.4								
Perform Weights & Inertias Analyses	8						.5						.5		
Perform S&C Analyses & Evaluation	9	.4		.5	.5				.5						
Develop Balanced Freebody Diagrams & Ext. Applied Loads	10				.1		.5	.2	.1			.4			
Establish Internal Load Distributions	11						.5	.5	.5		.5				
Evaluate Structural Strength, Stiffness, & Life	12	.4					.4	.5			.5	.4			
Preliminary Manufacturing Planning & Analyses	13	.5				.5							.4		
Prepare UCAV Proposal	14	.3	.4	.4	.4	.4	.4	.4	.4	.4	.4	.4	.4	.4	

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1														
2	.5								.1					
3		.3		.5										
4	.4		.8											
5	.1		.1			.1						.3	.1	
6	.1				.3									
7	.5					.8								
8						.5						.5		
9	.3		.3	.3				.3						
10				.1		.5	.4	.3			.3			
11						.5	.5	.3		.3				
12	.5					.3	.5			.5	.5			
13	.9				.9							.3		
14	.5	.8	.8	.8	.8	.8	.8	.8	.8	.8	.8	.8	.8	

Figure 4a/b: Activity Data/Rework Probabilities & Rework Impacts

(*IEEE Transactions on Engineering Management*: Browning and Eppinger 2002)

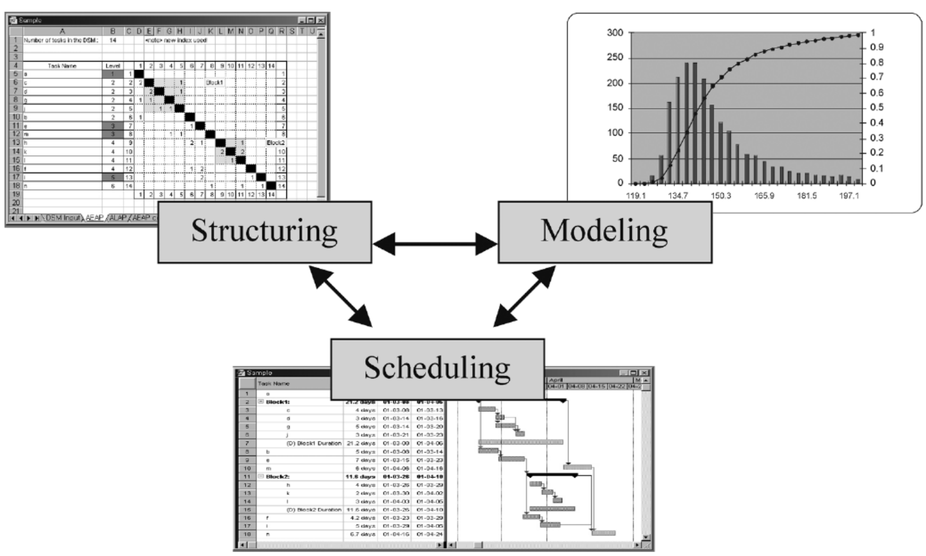


Figure 5: Integrated Project Management Framework

(*IEEE Transactions on Engineering Management*: Cho and Eppinger 2005)

DSM–based process model has been demonstrated as an effective tool through significant research and practice efforts in the field of complex project management.

However, for the analysis of ECM problems in relation with NPD from an enterprise level, it is not an appropriate approach because of the following two reasons:

1) The formation of DSM requires extensive knowledge of the system and could be expensive to maintain since the activity–based information need to be as accurate as possible to ensure reasonable results.
2) DSM is not able to capture the dynamic complexity arising from iterations and ECs. It is a tool of generating optimal sequencing given certain activity properties (i.e., precedence constraints, rework probabilities and impacts, learning curves, etc.). However, it is not suitable for operational analysis of ECM caused by different levels of uncertainty or strategic analysis of ECM and NPD policies.

2.3.2.2 System Dynamics Modeling of PD process

There is a rich body of research in the area of project management that successfully adopts *System Dynamics* (SD) methodology for modeling integrated development processes that accounts for dynamic features of process, resources, scope and targets (Ford and Sterman 1998). Among various drivers of project performance, *rework cycle* (Cooper 1993) has been identified and extensively analyzed as the core feature in almost all project development models to understand the schedule and cost overrun issues (e.g., Ford 1995; Ford and Sterman 1998, 2003; Reichelt and Lyneis 1999; Park and Peña–Mora 2003; Lin et al. 2007; Lyneis and Ford 2007).

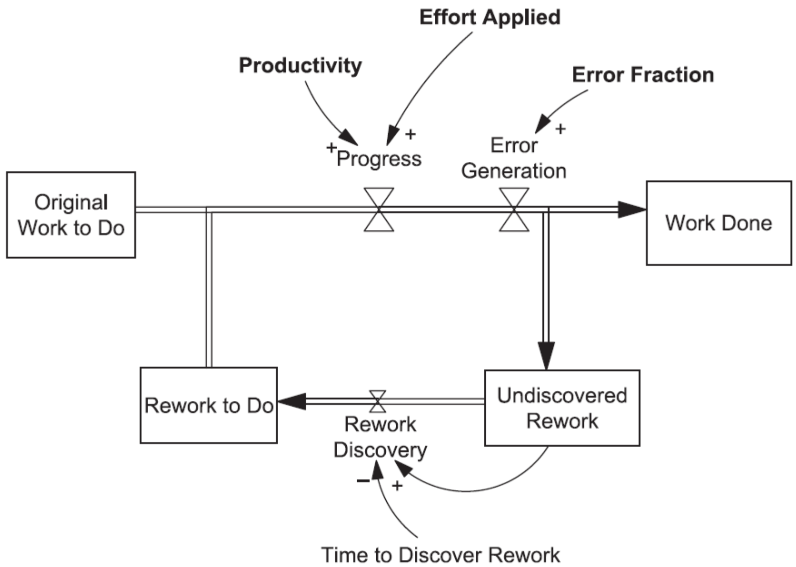


Figure 6: The Rework Cycle

(*Project Management Journal*: Cooper 1993)

As shown in *Figure 6*, backlog tasks from two stocks, “Original Work to Do” and “Rework to Do”, are continuously being handled at the rate of Progress determined by Productivity and amount of Effort Applied. After completion, they will flow into either “Work Done” or “Undiscovered Rework” at the rate of Error Generation determined by Error Fraction. Flawed but not yet recognized tasks then flow into “Rework to Do” at the rate of Rework Discovery determined by Time to Discover Rework.

Figure 7 illustrates an extension of the Rework Cycle proposed by Lin et al. (2008) for managing overlapped iterative product development, which is the most related work to this research. Causes of rework were explicitly categorized into i) *development errors* identified through review and testing, and ii) *corruptions* resulted from rework tasks in the upstream phase. Base case used to validate the model involves three phases (Concept Development, Detailed Design, and Pilot Production) and each phase is composed of completion, rework, and testing activities. The effects of different policies were studied among distinct types of NPD projects. First one is the how overlapping policies of when to start pilot production impact project performance in terms of percentage of reworked tasks. Second, different levels of overlapping in pilot production were examined. Lastly, investment activity policies were evaluated for improving activity quality and duration by introducing a new prototype machine and thus finding the quality problems earlier.

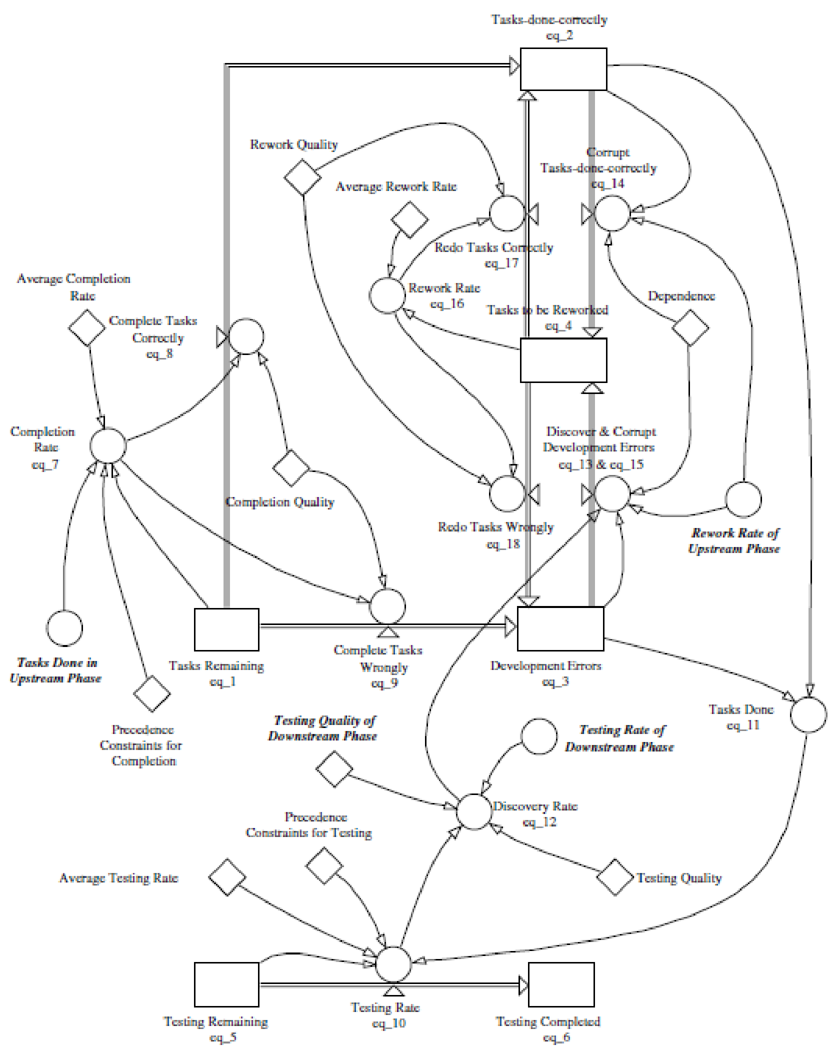


Figure 7: Dynamic Development Process Model (DDPM)

(*European Journal of Operational Research*: Lin et al. 2008)

Instead of tasks, Black and Repenning (2001) proposed a macro–level SD model using *engineered parts* as the unit of work flow to analyze different policies organizations may adopt for earlier problem resolution, better quality and performance in a *multi–projects* environment. Their model is composed of overlapping NPD projects which contains two phases (Early Phase and Current Phase) during a fixed development cycle of two years.

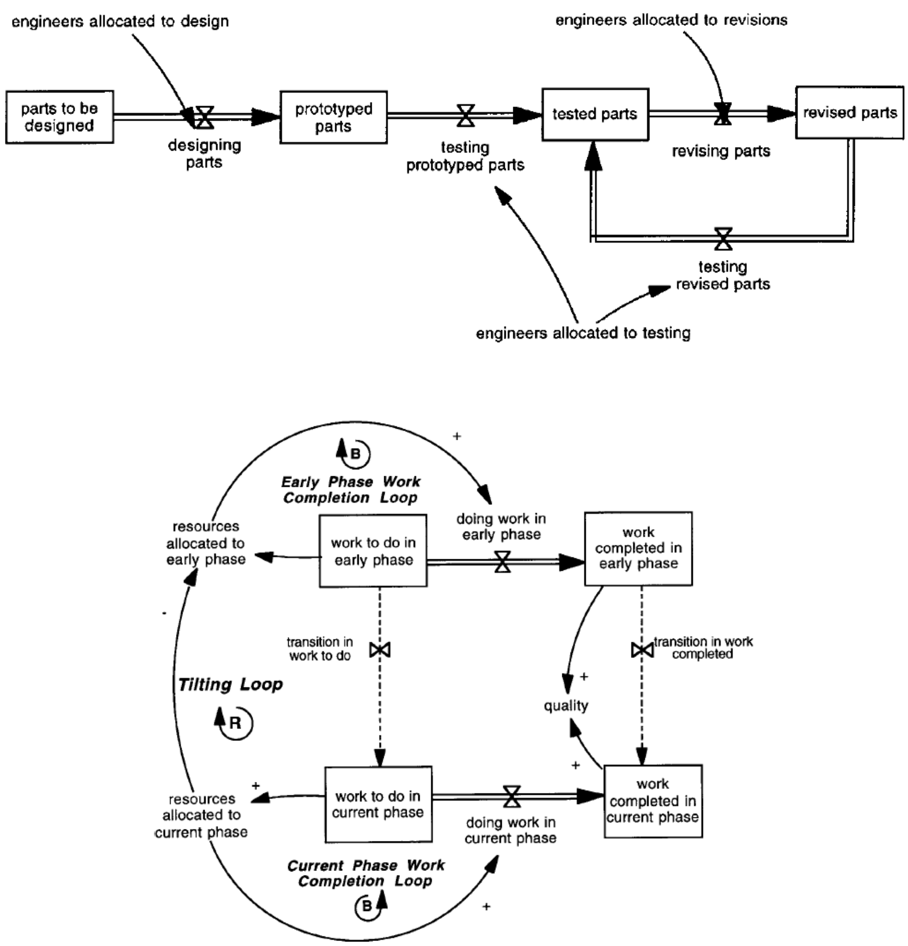


Figure 8a/b: Work Flow within a Phase/"Tilting Loop" Arising from Resource Allocation

(*System Dynamics Review*: Black and Repenning 2001)

Each single phase has the identical process structure as shown in *Figure 8a* with stocks representing accumulation of parts in various states of the PD process. Early phase is differentiated from current phase by fewer interdependencies among parts, a longer lead time for processing, and a lower priority for allocating resources. One–time increase in workload is used as test inputs to represent variations driven by changing requirements and discovery of problems. Quality, measured as the percentage of correctly launched parts, is employed as an indicator of project performance. *Figure 8b* illustrates the reinforcing *tilting loop* that drives more and more work to be completed in the last months before launch. Different policies were analyzed using this SD model framework.

While SD methodology has the advantage of strategic policy analysis over DE from a system–level aggregate view by highlighting the feedback loop and dynamic complexity arising from the influences among variables (Sweetser 1999; Tako and Robinson 2008), I claim that it is deficient by its nature in representing certain critical features of iterative concurrent NPD process and ECM problems.

1) Units of work flow of an SD model, no matter in the form of development tasks or product parts, don't have individual characteristics.
2) SD requires high abstract–level deterministic estimates of the process rate of the flow between stocks and therefore describes only deterministic behavior of the system. Any level of randomness or uncertainty, which is the nature of NPD and ECM processes, has to be included only via model structural changes such as building additional model compositions.
3) Even though the iterative NPD process can be represented by rework due to different causes (Lin et al. 2008), the phenomenon of change propagation due to the interconnected product components and the interrelated activities cannot be simulated as explicitly as in a DE model.

In the meanwhile, one of the shortfalls of DE methodology, the generally claimed lack of inherent closed–loop feedbacks which is valuable for studying the dynamic complexity of NPD and ECM processes, will be overcome in this research work by defining interacting process features and model variables.

2.3.2.3 Discrete–Event Macro–Model of an NPD Process

The generic NPD process demonstrated in this dissertation work is adapted from the model structure developed by Bhuiyan, Gerwin, and Thomson (2004). They presented a discrete–event simulation model that intends to determine how ***Overlap (OL)*** and ***Functional Interaction (FI)*** affect the performance measures of development time and effort under varying conditions of ***Uncertainty***.

Unlike many previous works that investigated only local performance by handling a subset of activities, the entire development process was modeled for studying both *Sequential Engineering (SE)* and *Concurrent Engineering (CE)* work methodologies. SE and CE versions shown in *Figure 9* are composed of building blocks of phases/activities and decision points, along with unidirectional information flows among blocks.

Three types of probabilistic rework were modeled: i) *Design Versions* that require one or more phases to be redone resulted from process reviews at the top level, ii) *Churn* that represents redoing an activity resulted from the communication between functions from the bottom level only when FI exists, and iii) *Overlap Spins* that occur in CE version among overlapped activities repeatedly for a maximum of three times.

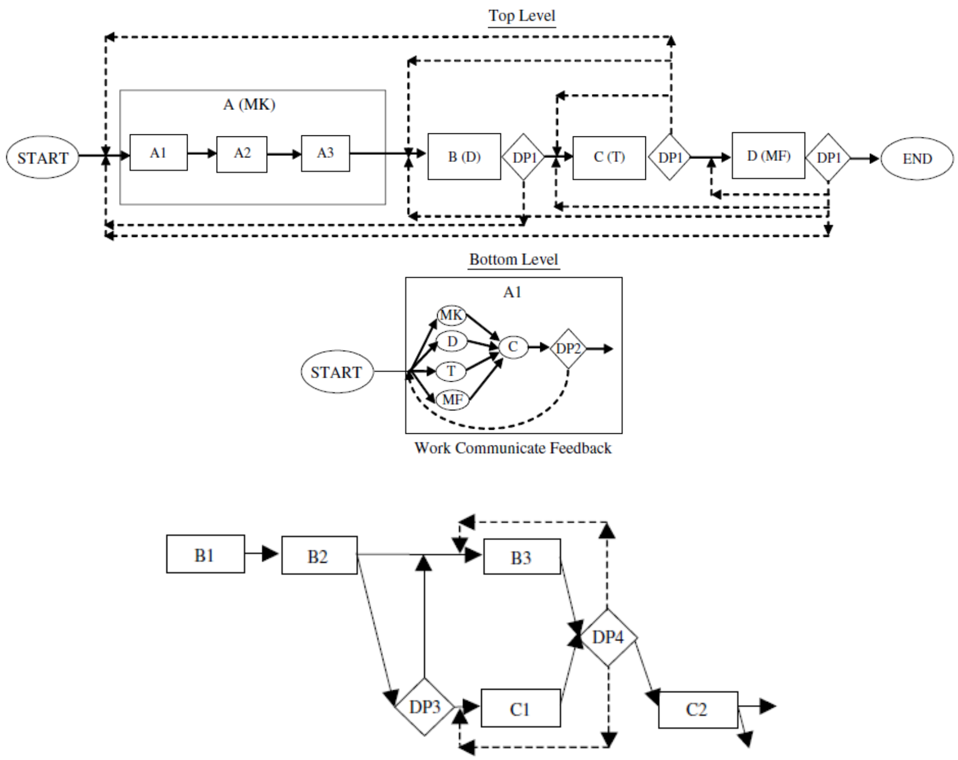


Figure 9: Sequential and Concurrent Process Model for NPD

(*Management Science*: Bhuiyan, Gerwin, and Thomson 2004)

As shown in *Figure 10*, rework probability at each decision point depends on i) two process characterizations: FI and OL (only affects probabilities of design versions), and ii) four uncertainty conditions of the NPD process which was first identified in (Krishnan, Eppinger, and Whitney 1997): (1) "slow evolution – low sensitivity", (2) "slow evolution – high sensitivity", (3) "fast evolution – low sensitivity", and (4) "fast evolution – high sensitivity".

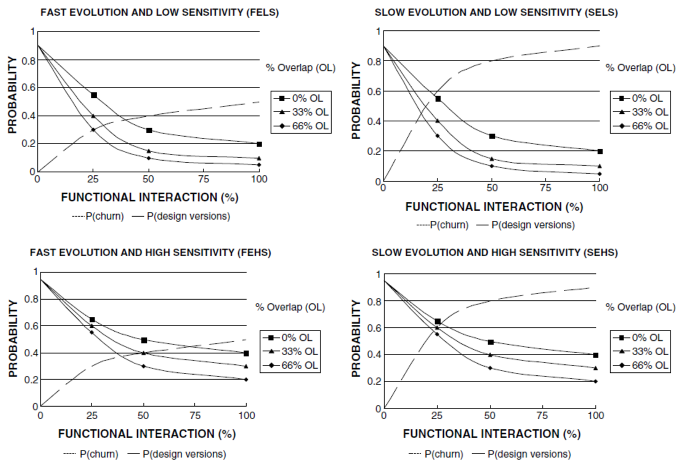


Figure 10: Probabilities of Churn and Design Versions

(*Management Science*: Bhuiyan, Gerwin, and Thomson 2004)

There are a number of problems inherent in this work. Main problems include:

1) Even though the model takes uncertainty, which implies the degree of completeness of information, in to consideration, it is accounted only for by making implicit assumptions: “slow information evolution causes high churn probabilities, and vice versa”, “low information sensitivity results in low design version probabilities, and vice versa”, and “as overlapping increases, probabilities of design version decreases”, as indicated in the shapes of the curves in *Figure 10*.
2) For each NPD process configuration with certain percentage of OL and FI under a particular uncertainty condition, the probabilities of churn and design versions are all identical for any activities within any phases, and remain static as the project evolves. This oversimplification represents a big departure from actual NPD processes in real life.
3) The model assumed an unlimited amount of resources, which is also unrealistic in practice.

In a later paper (Bhuiyan, Gatard, and Thomson 2006), this process modeling approach was expanded to include additional *Engineering Change Requests* (ECRs) other than design versions and churn by adding decision points for ECR control flow as shown in *Figure 11*. This simulation study aimed at comparing two ECM methods: i) immediate individual processing as they occur or ii) batch processing in groups. The probabilities of occurrence for ECRs were assumed to be the same as those for design versions.

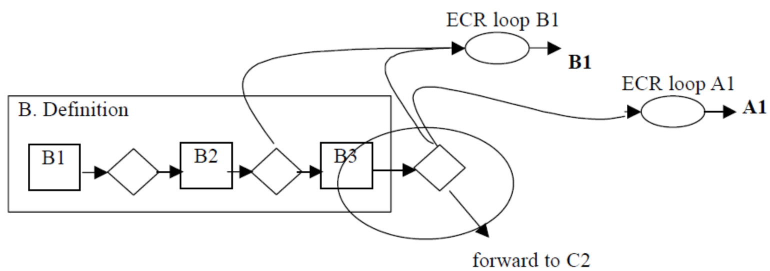


Figure 11: ECR Decision Points

(*European Journal of Innovation Management*: Bhuiyan, Gatard, and Thomson 2006)

This work is presumably the first study of ECM using computer simulation. However, besides those of the original model, it suffers following two major limitations:

1) Their model doesn't contemplate on the reasons for occurrence of ECRs, but only management mechanisms (i.e., perform changes individually or in a batch on a periodic basis) to lessen their impact. All the design changes, no matter how it is made to adapt to changes in requirements/technologies/ environment or issued for problems and errors, are treated exactly the same way.
2) The research topic of immediate or batch processing ECRs represents only a small subset of subjects in ECM domain comparing with the huge amount of existing

problems related to ECM and opportunities for improvement as introduced and discussed in the ECM literature.

2.4 Summary

Literature review in ECM is grouped into four major topics: i) general administration guidelines, ii) ECM in product structure and material requirements planning, iii) change propagation and knowledge management, and iv) computer sided ECM system. It reveals themes of change in the perspective from viewing ECs as disruptions to ECs as great opportunities of continuous improvement for an organization to keep a competitive edge. Also, a trend of shifting from considering ECs as material requirements planning problems mostly occurred in manufacturing environment to the exploration of change propagation and knowledge management in much earlier phases of NPD cycle is observed. This trend is partially due to centralized and integrated modern information systems.

However, this review demonstrates a lack of research–based ECM process models, especially from an enterprise–level systems perspective. This is evidence in the second part of this literature review in the area of *NPD Process Modeling*. Three influential but yet very different modeling approaches that highlight the effects of the concurrent and iterative nature of a PD process are reviewed in detail in terms of their modeling logic, major findings and limitations for shedding light on motivations of conducting the research work presented in this dissertation.

CHAPTER 3

CAUSAL FRAMEWORK

3.1 Introduction

This chapter investigates, from a systems perspective, the causes of four major ECM issues that companies and people involved in new product development need to address, i) occurrence of ECs, ii) long EC lead time, iii) high EC cost, and iv) the occurrence frequency of iterations and ECs. The conceptual discussion reported here is accomplished by creating *causal diagrams* (open loop) and *causal loop diagrams* (closed feedback loop) to study how external factors and internal system structure (i.e., the interacting variables comprising the system and the *cause–and–effect relationships* among them) contribute to specific behavioral patterns of the system. It is the first step toward the actual construction of a "real" simulation model described in *Chapter 5*, which is quantitatively augmented by *algebraic relationships* among the interrelated variables.

3.2 Causes of Occurrence of ECs

Causal Diagrams are not simulation models. But they are very helpful in conceptualizing the influences among interrelated variables that contribute to a certain system behavior diagrammatically and qualitatively. The arrow pointing from variable a to variable b indicates that a causes b. The plus **(+)** or minus **(–)** polarity at the tip of arrow indicates a positive or negative causality between the two variables, under the assumption that all else remain equal. A *positive causal link* means that the two variables change with the same trend (i.e., if variable a increases, variable b also increases). On the other hand, a *negative causal link* means that the two variables change in the opposite directions (i.e., an increase in variable a will cause a decrease in variable b).

In this section, an exploratory analysis is conducted by assimilating knowledge gained from relevant ECM literature and experience obtained from two rounds of field survey studies[3] to identify major drivers that cause the occurrence of both types of ECs (i.e., Emergent ECs and Initiated ECs) and the influences between interrelated contribution factors. A list of causal relationships is presented in

Table 4. Resulting from the verbal descriptions, a complete causal diagram for the occurrence of ECs is then developed, as shown in *Figure 12*. This diagram is created using Vensim simulation software by Ventana Systems, Inc. (Harvard, Massachusetts).

There are altogether four 1st level drivers (main causal factors as opposed to lower levels of factors deriving from the main stimulus) identified for the occurrence of EECs: i) *Design Errors*, ii) *Change Propagation*, iii) *Late Provisions of Product Specifications*, and iv) *Vendor Availability and Performance*.

Meanwhile, five 1st level drivers for the occurrence of IECs are: i) *New Government Legislations*, ii) *New Customer Needs*, iii) *Suggestions to Cost Reduction*, iv) *Suggestions to Quality/Function/Reliability Improvement*, and v) *Problems with Past Design*.

---

[3] The field studies will be discussed in detail in Chapter 4.

Table 4: Causes of Occurrence of ECs

<table>
<tr><th></th><th>Levels of Causes</th></tr>
<tr><td rowspan="4">Emergent ECs</td><td>I. Design Errors[4]
• Product Complexity[5]
▪ Complexity and Quantity of Components
▪ Degree of Component/System Coupling
▪ Technological Novelty
• Level of Expertise
▪ Technological Novelty
• Resource Availability
▪ Multiple Concurrent Projects
▪ Occurrence of ECs
• Team Complexity
▪ Cross Functional Team Skill Mix
▪ Geographical Locations of Team Members
• Process Complexity
▪ Project Design Solution Scope
▪ Degree of Activity Coupling
▪ Overlapping Strategy
• Overlapping Strategy</td></tr>
<tr><td>II. Change Propagation
• Product Complexity
• Process Complexity
• Common Components Across Product Lines</td></tr>
<tr><td>III. Late Provisions of Product Specifications
• Overlapping Strategy</td></tr>
<tr><td>IV. Vendor Availability and Performance
• Environment Complexity
• Internationalization and Localization</td></tr>
<tr><td rowspan="3">Initiated ECs</td><td>I. New Government Legislations[6]
• Environment Complexity
• Internationalization and Localization</td></tr>
<tr><td>II. New Customer Needs</td></tr>
<tr><td>III. Suggestions to Cost Reduction
• Internationalization and Localization
• New Technology Advances</td></tr>
</table>

[4] “*Design Error*” is a generalized term used to describe all categories of problems or errors detected in activities whose outcomes fail to meet the pre-determined criteria but have already been released to the downstream phase. It may occur in any stage of the product life cycle: design, testing, prototyping, manufacturing, and useful time.

[5] Since some of the factors contribute to more than one category of time drivers (e.g., Product Complexity affects both I and II), to avoid meaningless repetition, decomposition of their root causes is only given at the first appearance.

[6] Government legislations include safety standards, certification requirements, environmental regulations, etc.

<table>
<tr><td rowspan="2"></td><td>IV. Suggestions to Quality/Function/Reliability Improvement
• New Technology Advances</td></tr>
<tr><td>V. Problems with Past Design</td></tr>
</table>

3.3 Causes of Long ECM Lead Time

Table 5: Causes of Long EC Lead Time

<table>
<tr><th></th><th>Levels of Causes</th></tr>
<tr><td rowspan="4">Long EC Lead Time</td><td>I. Expected Mean and Variance of EC Duration
• EC Complexity
• Knowledge of EC
  ▪ Level of Expertise
    o Technological Novelty
  ▪ Technological Novelty
• Quality of Interdisciplinary Collaboration</td></tr>
<tr><td>II. Mean and Variance of Propagated ECs
• EC Scope
  ▪ Change Propagation
    o Process Complexity
    o Product Complexity
  ▪ Number of Undergo ECs</td></tr>
<tr><td>III. Approval Time
• EC Scope
• Completeness of EC Impact Assessment
• Quality of Communication
• Quality of ECM Process
• Resources Allocated to ECM
  ▪ EC Scope
  ▪ Flexible Resource Capacity
  ▪ EC Priority
  ▪ Expected Resource Requirement of EC
    o EC Complexity
  ▪ Undergo ECs
• Accumulated ECs</td></tr>
<tr><td>IV. Waiting Time
• Organization Inertia
  ▪ EC Scope
• Batching Size (and Setup Time)[7]
• Resources Allocated to ECM
• Quality of ECM Process
• Accumulated ECs</td></tr>
</table>

[7] Setup time needs to be considered whenever a certain amount of time is required to prepare a device, machine, or system to be ready to process an EC or a batch of ECs. For instance, many testing activities should include setup times in addition to measurement times.

	• *Randomness of EC Arrival Patterns*

EC lead time is defined as the delay between the initiation and execution of an EC. To be more specific, it is a delay starting from the EC proposal till the official implementation of the change. *Table 5* shows the key external and internal factors that are responsible for long EC lead time and the mutual influence among them. There are four main categories of drivers of long lead time: i) *Expected Mean and Variance of EC Duration*, ii) *Mean and Variance of Duration of Propagated ECs*, iii) *Approval Time*, and iv) *Waiting Time*. The whole picture of the multidirectional causal relationships between all variables is provided in *Figure 13*.

3.4 Causes of High ECM Cost

Generally speaking, researchers in design process management have devoted more attention to identify challenges and risks to achieve desired milestones on *schedule*, as compared to performance indicators of product development projects such as *cost* (or *development effort*) and *quality*. Even with the scarcity of research in systematic analysis of EC cost, Saeed, Bowen and Sohoni (1993) argued that the past research only focused on EC–related costs in terms of engineer time or labor cost while other different types of EC–related costs (i.e., costs with regards to material and equipment), both internal and external to the organization, should be also taken into account.

High ECM cost can be attributed to three main contributors according to the existing literature (Balakrishnan and Chakravarty 1996; Loch and Terwiesch 1999): i) *Material Cost*, ii) *Labor Cost*, and iii) *Equipment Cost*, even though some of the sub–causes are being interwoven.

Note that Lateness of EC Arrival and Impact of EC occur as a third level sub–cause of almost every second level sub–causes. This phenomenon is confirmed by empirical findings of "extremely high cost of change and time pressure resulted from late ECs" reported in industrial case studies (Clark and Fujimoto 1991; Reidelbach 1991; Pikosz and Malmqvist 1998). Verbal and visual are shown in *Table 6* and *Figure 14,* respectively.

Table 6: Causes of High EC Cost

	Levels of Causes

High EC Cost	***I. Material Cost***
	• *Inventory Fluctuation*
	  ▪ *Lateness of EC Arrival and Impact of EC*
	• *Obsolescence Cost*
	  ▪ *Lateness of EC Arrival and Impact of EC*
	• *Rework Cost*[8]
	  ▪ *Lateness of EC Arrival and Impact of EC*
	• *Backorder Cost*[9]
	  ▪ *Lateness of EC Arrival and Impact of EC*

	II. Labor Cost
	• *Mental Setup/Retraining Cost*
	  ▪ *Lateness of EC Arrival and Impact of EC*
	• *Staff Motivation*
	  ▪ *Engineer Time*
	• *Engineer Time*[10]
	  ▪ *Lateness of EC Arrival and Impact of EC*

	III. Equipment Cost
	• *Additional Prototype Tools*
	  ▪ *Lateness of EC Arrival and Impact of EC*
	• *Additional Production Tools*
	  ▪ *Lateness of EC Arrival and Impact of EC*

[8] Rework cost includes any development and manufacturing cost incurred during the course of an EC (e.g., direct material cost, manufacturing overhead, testing cost, etc.) except additional engineer time which is counted under Labor Cost.

[9] Backordering impacts both committed–orders and forecast–demands. The backorder cost for committed–orders is usually higher than that for forecast–demands (Balakrishnan and Chakravarty 1996).

[10] EC-related engineer time is the sum of all the additional cross functional labor time dedicated to an EC, including design, testing, manufacturing, sales/marketing, financial, suppliers, etc.

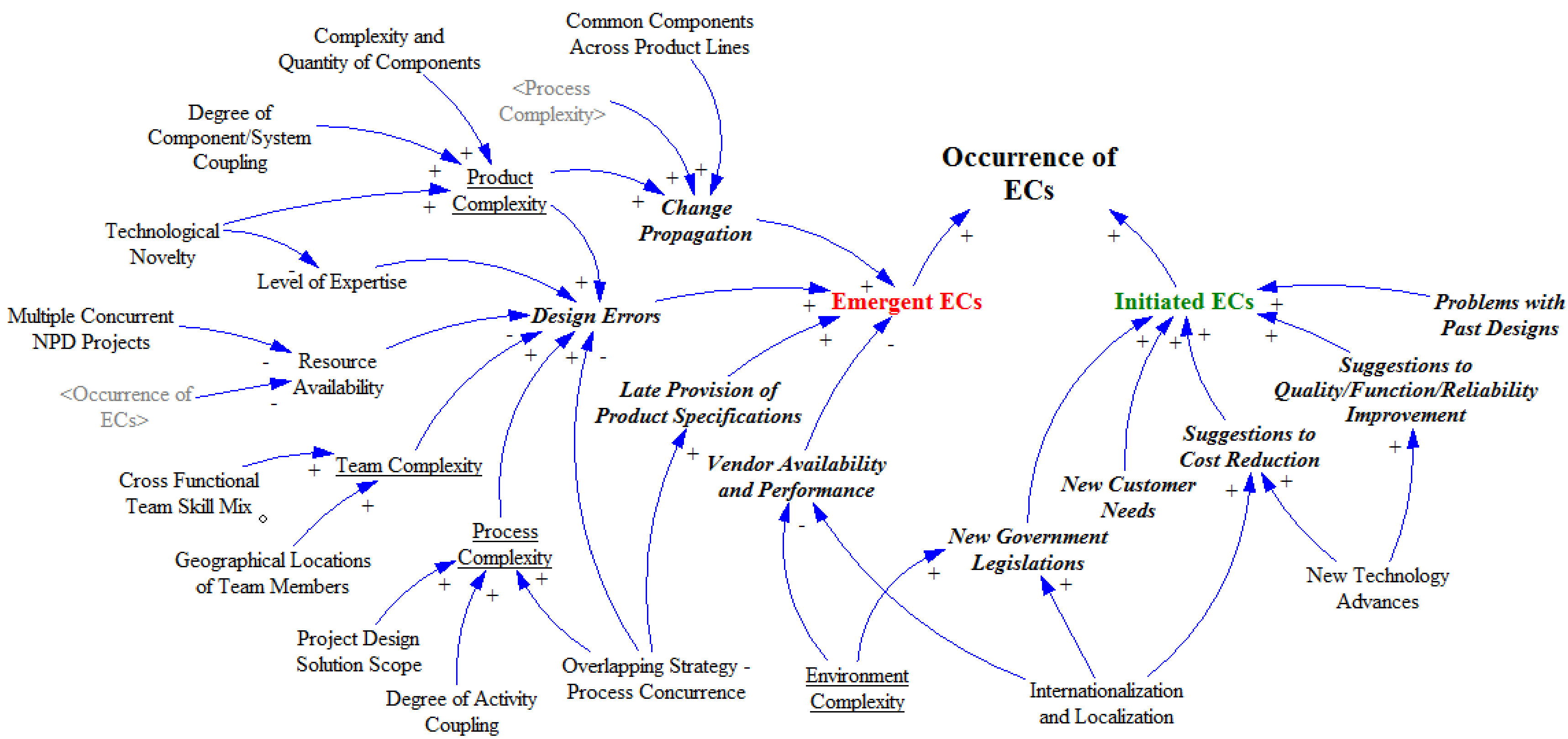


Figure 12: Causal Diagram of Occurrence of ECs

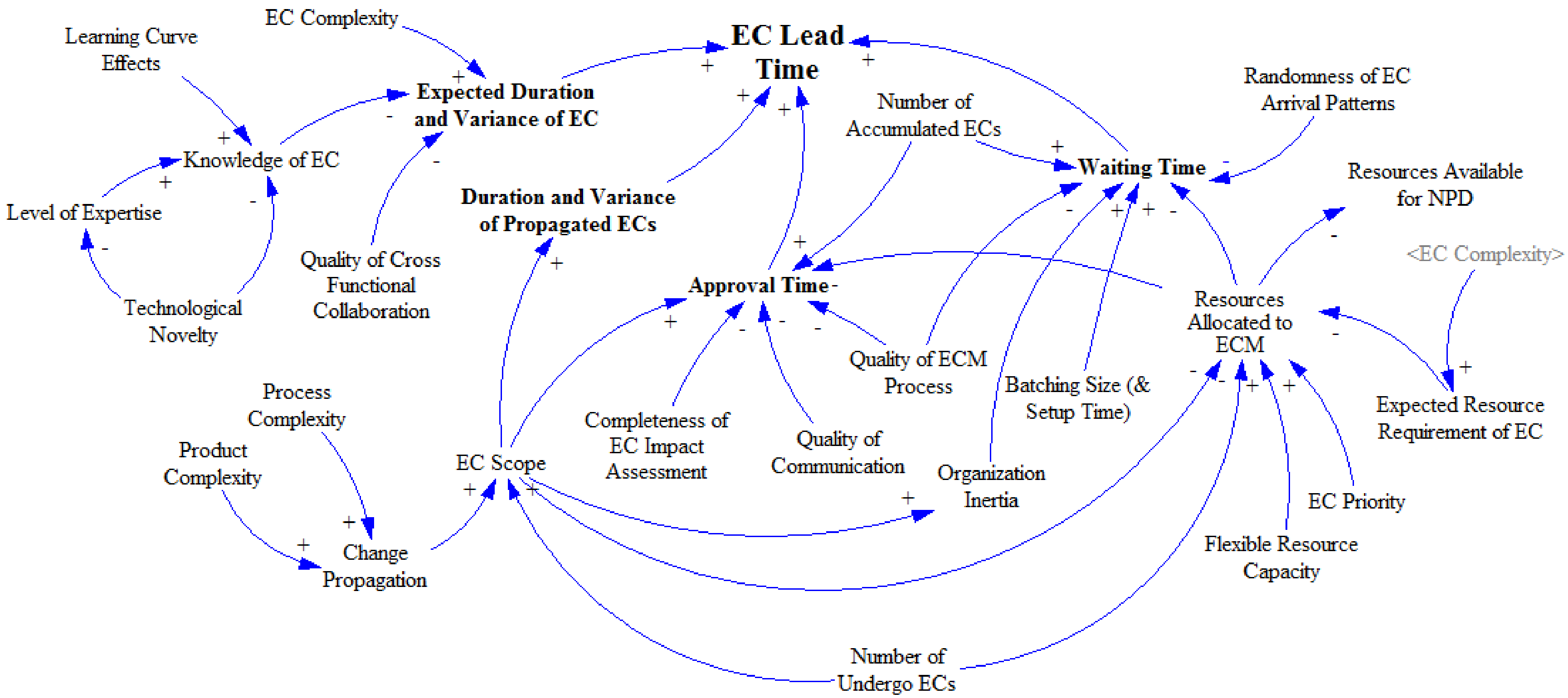


Figure 13: Causal Diagram of Long EC Lead Time

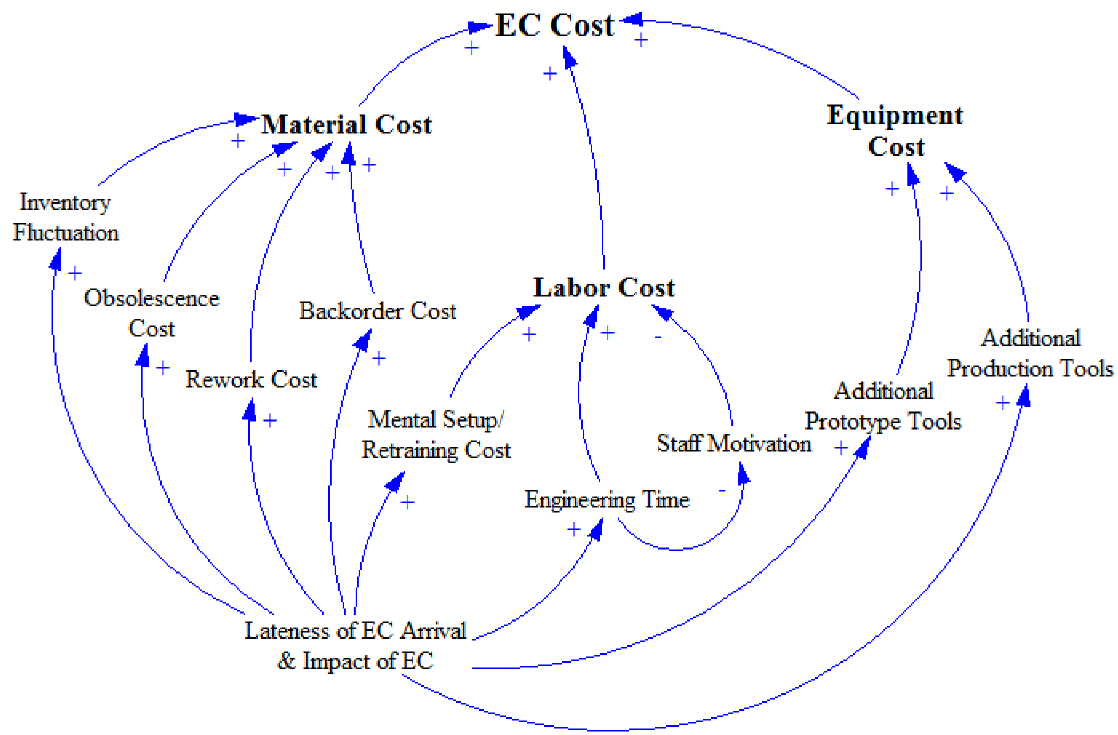


Figure 14: Causal Diagram of High EC Cost

3.5 Causal Loops of the Occurrence of Iterations and ECs

This section constructs and interprets causal diagrams /causal feedback loop diagrams of the occurrence of iterations and ECs in a resource–constrained environment.

3.5.1 Overview

A ***Causal Loop Diagram*** is a closed sequence of causes and effects that form a feedback structure of information flow among interrelated dynamical system variables. Again, causal loop diagrams are not simulation models. However, they can be utilized to facilitate system thinking, and more specifically, to develop a better understanding of how a system variable dynamically responds to feedback from other interrelated variables which are in turn influenced by it, by employing feedbacks between cause–and–effect relationships and therefore closing the loop. Instead of looking for external issues that cause certain behavior of a system (e.g., the ECM issues such as occurrence of ECs, long EC lead time, and high EC cost as have been shown along the previous sections) using causal diagrams, a causal loop feedback diagram has the advantage of identifying how the "internal structure of the system" (Kirkwood, 2010) generates the patterns of behavior. For instance, as ECs arrive more frequently, we are interested in studying how the interrelated system variables (such as resource availability, design solution scope, solution uncertainty, etc.) are affected, and thus loop back and further influence the EC occurrence frequency.

A causal loop diagram can be either reinforcing or balancing, depending on how many numbers of negative/positive links it consists. An even number of negative links results in ***Reinforcing (Positive) Loops*** that are associated with an exponential change same as the original assumption. An odd number of negative links results in ***Balancing (Negative) Loops*** that cause the trend to be stable and contradict the initial assumption. The ***Length*** of a causal loop is the number of variables contained within that loop. In later subsections, in additional to the loop number, length of the loop is also included in generating its identifier (i.e., Loop # of Length *n*).

By including five interdependent variables related with NPD and ECM processes (Iteration/EEC Size, Solution Completeness, Design Solution Scope, Learning Curve Effects, and Resource Availability), and the three levels of uncertainty (Activity Uncertainty, Solution Uncertainty, and Environmental uncertainty), *Figure 15* illustrates the primary cause–and–effect loops that drive the occurrence of iterations, EECs and IECs.

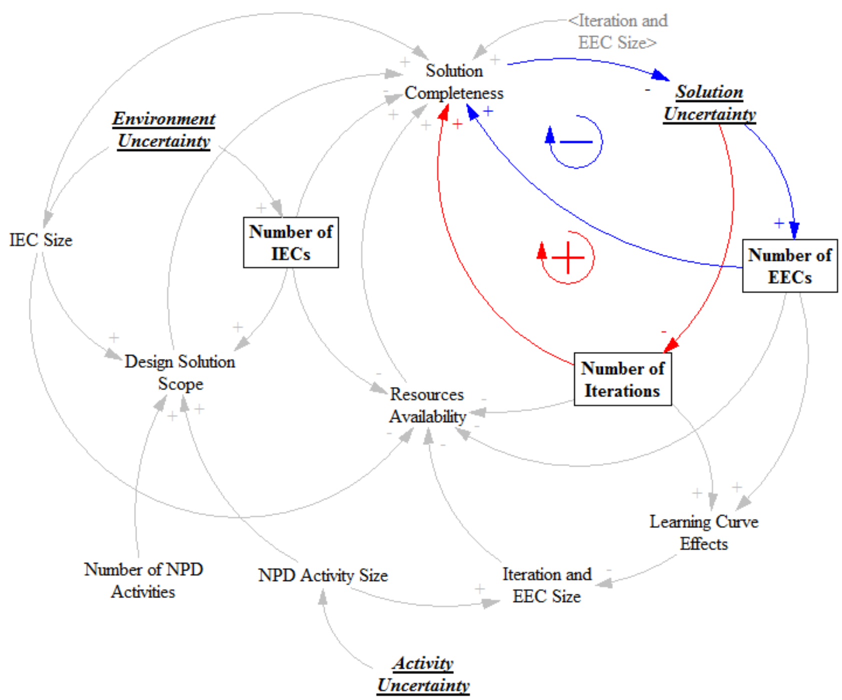


Figure 15: Demonstration of Balancing Loop (–) and Reinforcing Loop (+) of the Occurrence of Iterations and EECs

The balancing loop of length 2 in blue depicts the reduction in the number of incoming EECs as a result of handling EECs. It is due to the fact that processing of more EECs leads to an increase in the solution completeness of the NPD project; and thus solution uncertainty decreases.

Given the definition of EEC probability which is assumed to be exponentially decreasing as the project's solution uncertainty decreases, the influence is along the same direction, and therefore number of EEC occurrence decreases. On the contrary, the reinforcing loop of length 2 in red indicates an opposite phenomenon when considering the occurrence of iterations: even more iterations will appear as a consequence of performing iterations. It results from the assumption of an exponentially increasing iteration probability as solution uncertainty decreases. Processing of more iterations leads to solution uncertainty reduction, and therefore the number of iteration occurrence increases.

However, these two feedback loops alone don't generate the overall behavior of EEC and iteration occurrences. The inclusion of both resource constraints and learning curve effects makes the cause–effect relationships among variables more complicated and the net consequence hard to predict. A full list of feedback loops will be given with explanations in the following subsections.

3.5.2 Causal Loop Diagrams of EEC Occurrence

Four feedback loops of various lengths that drive the patterns of EEC occurrence are first examined. There are altogether five interdependent variables that form the loops: i) *EEC Size*, ii) *Solution Completeness*, iii) *Solution Uncertainty*, iv) *Learning Curve Effects*, and v) *Resource Availability*.

3.5.2.1 Balancing Loops

<u>*Loop 1 of Length 2*:</u> **# of EECs → Solution Completeness → Solution Uncertainty → # of EECs**

Explanation: As explained in section 3.1.

<u>*Loop 2 of Length 5:*</u> **# of EECs → Learning Curve Effects → Iteration & EEC Size → Resource availability → Solution Completeness → Solution Uncertainty → # of EECs**

Explanation: As a result of increasing learning curve effects, an increase in the occurrence of EECs leads to a reduction in later EEC durations compared with the original level (i.e., the basework duration of that particular activity). Then resource availability increases because less time is taken for completing EECs, which in turn accelerates the rate of solution completeness and thus results in a closed loop back to the decreasing occurrence of EECs.

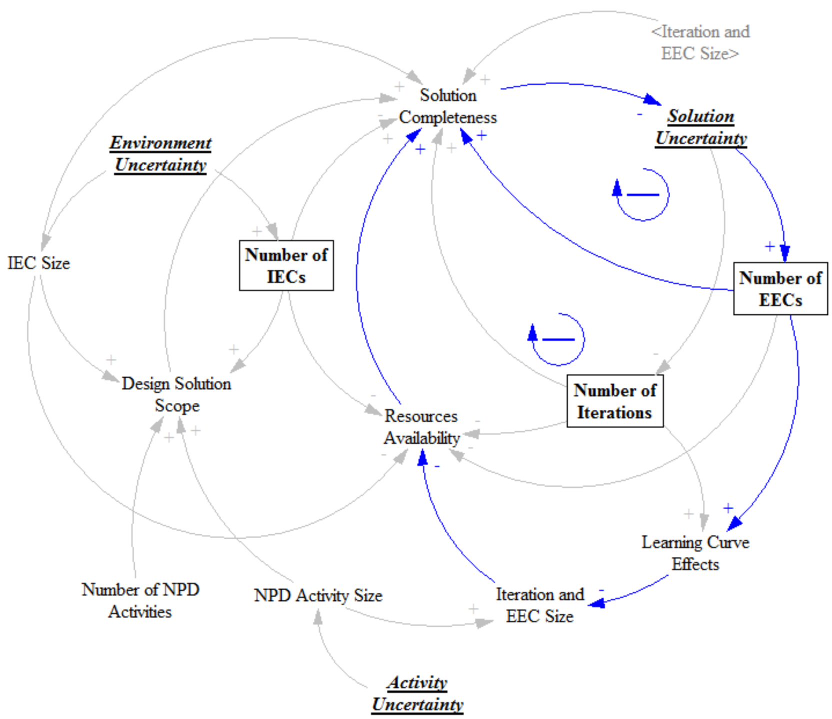


Figure 16: Balancing Feedback Loops of EEC Occurrence

3.5.2.2 Reinforcing Loops

Loop 3 of Length 3: **# of EECs → Resource availability → Solution Completeness → Solution Uncertainty → # of EECs**

Explanation: While the explanation of Balancing Loop 2 is based upon the indirect positive impact of EEC occurrence on resource availability through the reduction of later EEC durations owing to learning curve effects, this reinforcing feedback loop can be interpreted by the direct negative influence of EEC occurrence on resource availability: the more EECs occur, the more resource will be allocated to process them. As opposed to Loop 2, a decrease in resource availability decelerates the rate of solution completeness and thus causes an increasing occurrence of EECs.

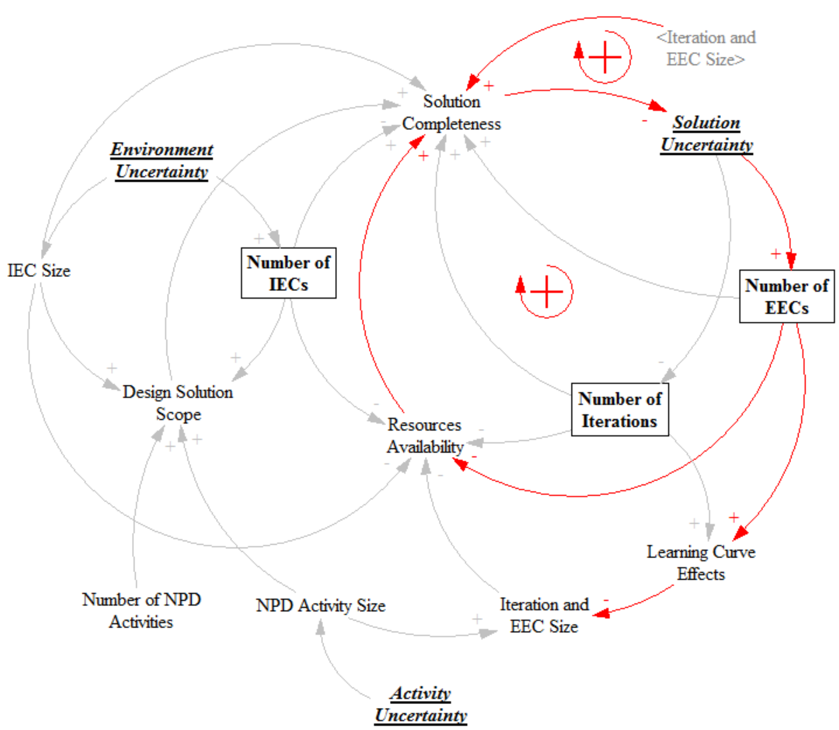


Figure 17: Reinforcing Feedback Loops of EEC Occurrence

<u>Loop 4 of Length 4:</u> **# of EECs → Learning Curve Effects → Iteration & EEC Size → Solution Completeness → Solution Uncertainty → # of EECs**

Explanation: Despite the indirect effects of EEC size reduction (which results in an increase in the resource availability) on an accelerating solution completeness rate that has been described by Loop 2, a decrease in EEC size also has a direct negative impact on solution completeness because of less contribution to close the information deficiency towards the final design solution. Again, a decreasing rate of solution completeness causes an increasing occurrence of EECs.

3.5.3 Causal Loop Diagrams of Iteration Occurrence

Due to the fact that both iterations and EECs relate to the four other variables in the exactly same fashion except solution uncertainty, which is assigned with an opposite causal relationship (i.e., iteration/EEC probability increases/decreases as solution uncertainty decreases), all the cause–effect feedback loops remain the same but of a switched growth pattern. That is, balancing loops for EEC occurrence become reinforcing loops for iteration's, and vice versa.

This section only provides verbal description and two figures that highlight the two loop types[11]. Explanation and interpretation of each loop can be consulted from its counterpart of EEC occurrence.

3.5.3.1 Balancing Loops

<u>Loop 5 of Length 3</u>: **# of Iterations → Resource availability → Solution Completeness → Solution Uncertainty – # of Iterations**

[11] The same situation applies to Section 3.5.5: Causal Relationships between IEC and Iteration Occurrences

Loop 6 of Length 4: **# of Iterations → Learning Curve Effects → Iteration & EEC Size → Solution Completeness → Solution Uncertainty → # of Iterations**

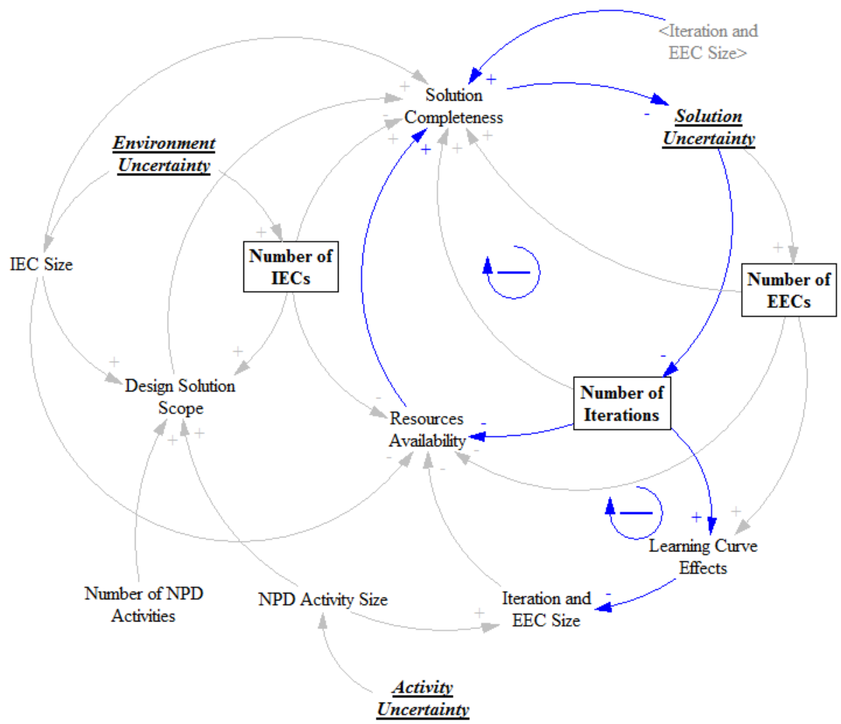


Figure 18: Balancing Feedback Loops of Iteration Occurrence

3.5.3.2 Reinforcing Loops

Loop 7 of Length 2: **# of Iterations → Solution Completeness → Solution Uncertainty → # of Iterations**

Loop 8 of Length 5: **# of Iterations → Learning Curve Effects – Iteration & EEC Size → Resource availability → Solution Completeness → Solution Uncertainty → # of Iterations**

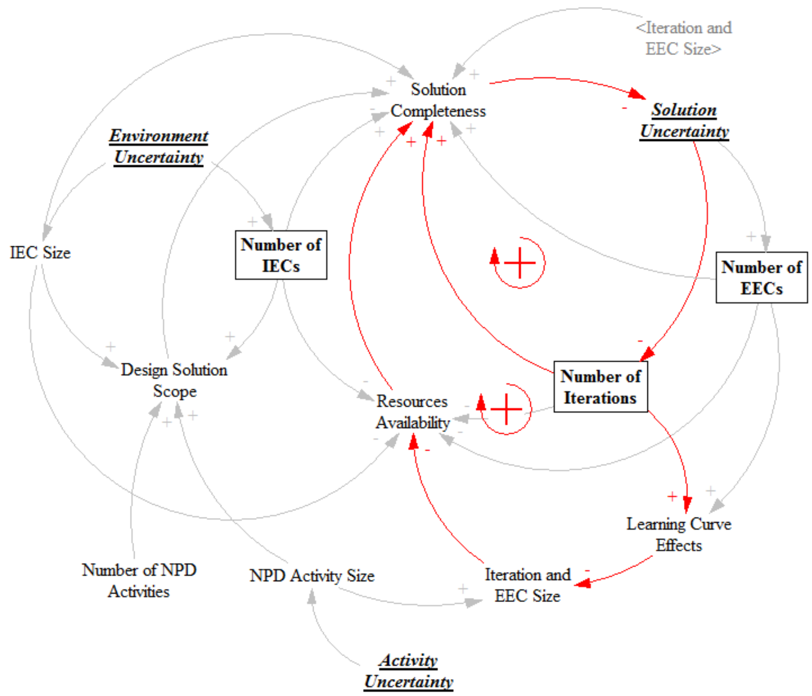


Figure 19: Reinforcing Feedback Loops of Iteration Occurrence

3.5.4 Causal Relationships between IEC and EEC Occurrences

This research implicitly makes a simplifying assumption that environmental uncertainty is the only cause of IEC arrivals. In addition, it is assumed to be an *exogenous* external factor that is not affected by any other internal system variables (e.g., resource availability, design solution scope, solution uncertainty, etc.), nor is it calculated by the model. As a result, there is no causal loop existing for the occurrence of IECs. However, exogenous factors influence other variables in the model. For instance, a raise in IEC arrivals does affect the number of coming EECs by increasing the design solution scope, decreasing the resource availability. Figures indicating each of the positive or negative links of various lengths, and the associated interpretation will be presented in the following subsections.

3.5.4.1 Positive Links

<u>*Causal link 1 of Length 4*:</u> **# of IECs → Design Solution Scope → Solution Completeness → Solution Uncertainty → # of EECs**

Explanation: The processing of an increasing number of IECs will increase the design solution scope by accomplishing additional solution goals in response to outside sources. Then solution completeness decreases correspondingly, and therefore leads to an increase in the occurrence of EECs as explained before.

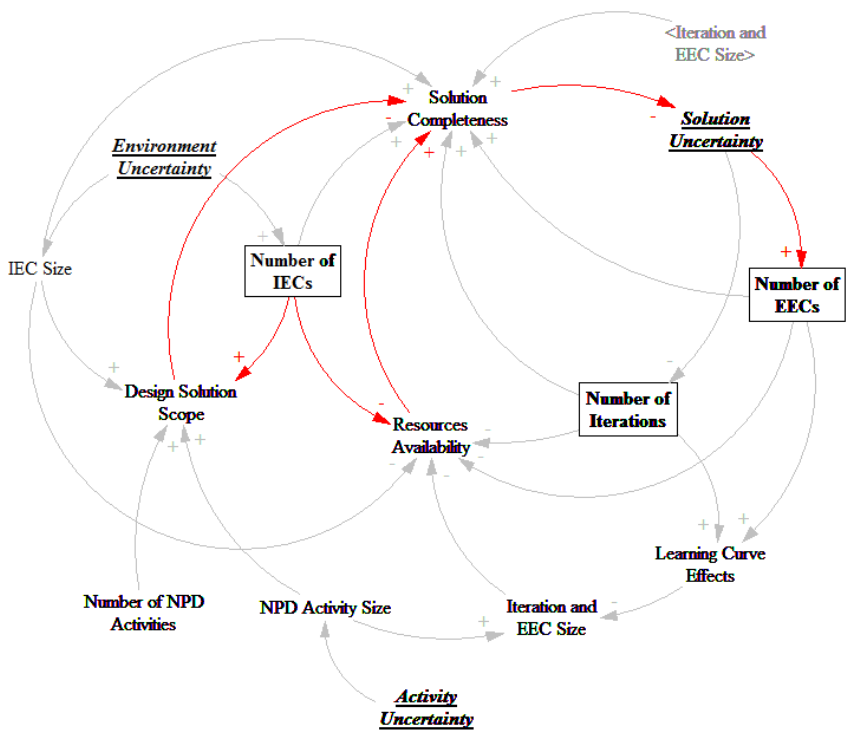


Figure 20: Positive Links between the Occurrence of IECs and EECs

<u>*Causal link 2 of Length 4*:</u> **# of IECs → Resource Availability → Solution Completeness → Solution Uncertainty → # of EECs**

Explanation: Handling an increase in the occurrence of IECs will decrease the resource availability which again causes a decelerating rate of solution completeness, and thus results in further EEC occurrence.

3.5.4.2 Negative Links

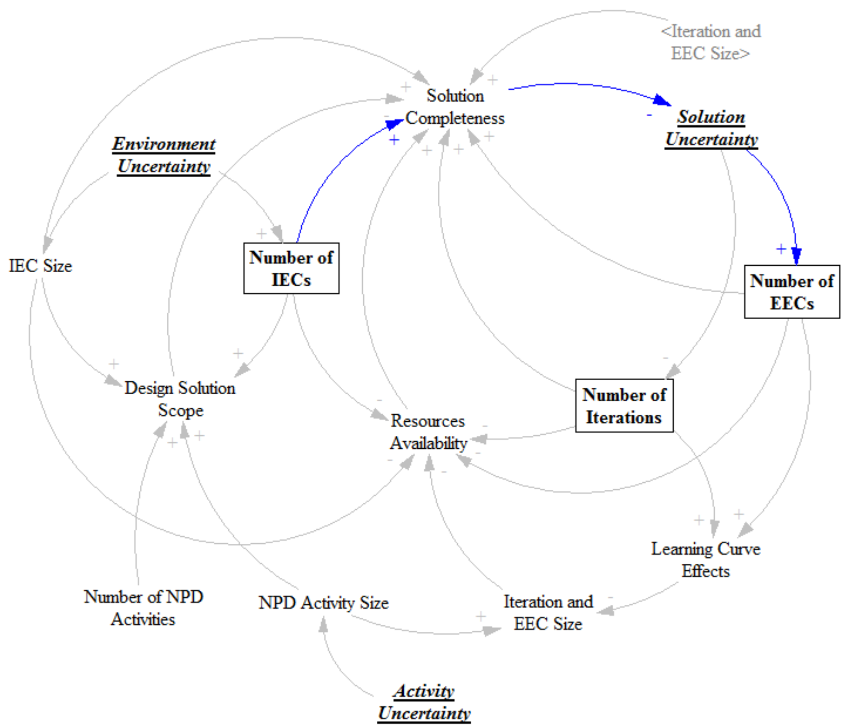


Figure 21: Negative Links between the Occurrence of IECs and EECs

<u>*Causal link 3 of Length 3*:</u> **# of IECs → Solution Completeness → Solution Uncertainty → # of EECs**

Explanation: The processing of an increasing number of IECs will contribute not only to the increase of the denominator of solution completeness as explained by Causal Link 1, but also to the increase of the numerator by the same amount. Whether the combined effect of both links is a net increase or decrease depends on how late the stage NPD project is in. When the numerator is less than the denominator, an increase in IECs leads to a net increase of solution completeness, and vice versa. That is to say, early IECs, though unexpectedly consumes resource capacity, tend to decrease the solution uncertainty. However, late IECs will further increase solution uncertainty.

3.5.5 Causal Relationships between IEC and Iteration Occurrences

Figures of causal links between IEC and Iteration occurrences and verbal descriptions are presented as follows. Explanation and interpretation of each causal relationship can be consulted with its counterpart causal link between IEC and EEC occurrences.

3.5.5.1 Positive Links

Causal link 4 of Length 3: **# of IECs → Solution Completeness → Solution Uncertainty → # of Iterations**

3.5.5.2 Negative Links

Causal link 5 of Length 4: **# of IECs → Design Solution Scope → Solution Completeness → Solution Uncertainty → # of Iterations**

Causal link 6 of Length 4: **# of IECs → Resource Availability → Solution Completeness → Solution Uncertainty → # of Iterations**

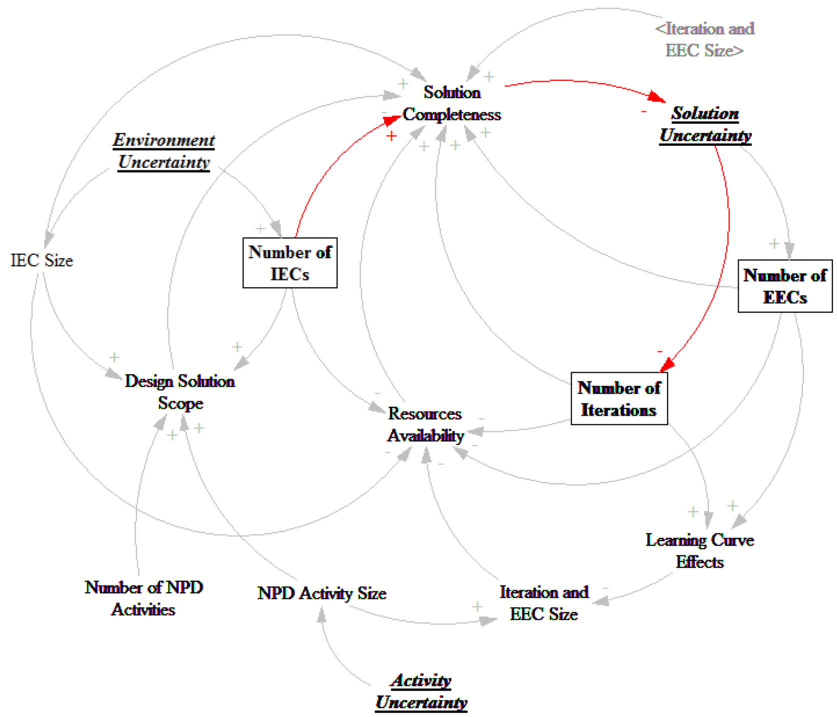


Figure 22: Positive Links between the Occurrence of IECs and Iterations

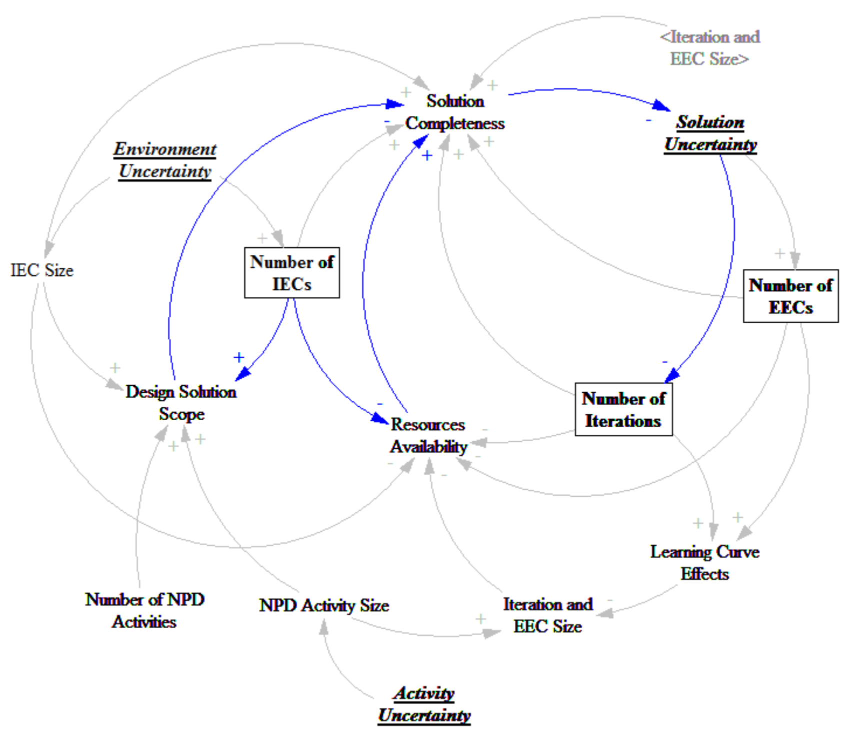


Figure 23: Negative Links between the Occurrence of IECs and Iterations

3.6 Summary

ECM problems should never be analyzed in isolation. This chapter explored the causes of four major ECM problems: i) occurrence of ECs, ii) long EC lead time, iii) high EC cost, and iv) occurrence frequency of iterations and ECs. Specifically, highly dynamic causal couplings in closed–loop relationships are identified for problem of “occurrence frequency of iterations and ECs”, and they will be used as the conceptual foundation to define variables of the simulation models developed in later chapters.

It is important to note that all the factors and their causal relationships are identified based on either theories or hypotheses proven in the existing literature, and/or insights obtained from the field surveys (which will be detailed in *Chapter 4*) or reported by previous empirical studies. For sure, they are limited both in breadth (i.e., the comprehensiveness of first level of causal contributions) and depth (i.e., levels of causality ranging from two to at most four are explored), reflecting only current understanding and are always subject to change because of the altering nature of the complexity rooted in four main elements of PD projects (i.e., product, process, team and environment). Despite of the limitations, this reported exploratory analysis reflects a common understanding between industry and academia of the key contribution factors to these ECM problems from a systems view.

In particular, the closed causal feedback loops constructed in *Section 3.5* depict how the initial occurrence of iterations/ECs will lead to the subsequent modification of occurrence frequency by taking into account other interrelated variables (e.g., EC size, solution completeness, solution uncertainty, learning curve effects, resource availability, and etc.) and presenting simple cause–and–effect relationships between them. A combination of both positive and negative feedback loops indicates that the complex and dynamic interrelationships among variables make the prediction of iteration/EEC occurring patterns not so straightforward. This phenomenon points out the necessity of constructing a simulation model that can help further quantitative analyses.

CHAPTER 4

FIELD SURVEY STUDIES

4.1 Introduction

Three field survey studies presented in this chapter were conducted within automobile and IT product/service industries, in which ECM problems are commonly encountered (Balakrishnan and Chakravarty 1996), during a 4–week period in the summer of 2010 and 10–week period in the summer of 2011. Information and data were collected concerning:

- Overview of the organization structure and its respective industry;
- Descriptions of the products it develops (e.g., product complexity, production process complexity, technological novelty, supplier involvement, etc.);
- The company's standard NPD and ECM processes and practices; and
- Information and data of specific NPD projects and ECM practice (e.g., list of activities, durations, and resource loading of typical NPD projects; arrival frequency of change requests and estimates of effort; etc.).

To collect data, the following five steps were taken:

1) Explain the scope and objective of the field survey study;
2) Review NPD and ECM process documents;
3) Hold several rounds of structured and unstructured interviews with related staff members;
4) Send data collection forms for detailed information gathering; and
5) Perform statistical analyses of the collected data and draw conclusions from the analyses.

Among the three case studies, the one carried out in *Company A* had to be halted after step three due to confidentiality issues as requested by the organization, while the others conducted in *Company B* and *Company C* went through the entire five steps.

4.2 Localization of an Automotive Engine in Company A

This section reports descriptions gathered from *Company A* regarding how ECs of various kinds to an automotive engine are managed due to *product globalization and localization*, which lead to transferring design, development, and procedure functions to target markets in different regions (Yusuf, Altaf, and Nabeshima 2004). The subject company is a wholly owned operating subsidiary corporation based in Shanghai, China. It manages the procurement logistics (e.g., make/buy decisions, supplier management, sourcing and purchasing activities, order control, etc.) to supply inexpensive and high–quality parts to its assembly plants in Europe. The interviewee is a product engineer responsible for communicating engine integration strategy and ECM issues with the company's local suppliers in China.

Power train is acknowledged as one of the most complex sub–systems in a motor vehicle, among which an engine is the critical component to realize its power generation function. The automotive engine *Alpha* discussed in this case study consists of 148 buy–level components and parts, and great percentages (about 95%) of them are being produced locally. As a result, redesign/EC efforts are required for most of the localized components and parts. Typical types of ECs include change of suppliers, change of drawing versions, change of materials, change of dimensions, changes affecting fit/form /function, changes affecting installation on vehicle, changes affecting price, changes caused by field failures, changes affecting safety and quality characteristics. ECs are addressed by the Joint Change Review Board, which is composed of Program Manager, Chief Engineer, Process Manager, Launch Manager, Quality Manager, Cost Engineer, Purchasing Manager, Finance Controller, Portfolio Planner, and Key Account Manager. The ECM period in *Company A* is the entire duration from the project start date but not beyond gamma prototype construction.

Started in June of 2010, *Project X* was aimed at completing the localization of engine *Alpha* within 8 months. According to the project calendar, fabrication testing of the first batch of gamma prototypes is planned to start by the middle of February in 2011. Falling behind the original schedule, most of the components and parts were delayed till the end of February. Some were delayed till the middle of March for the first round of gamma engine installation testing. And some others were even delayed till June of 2011 for the second round of Gamma engine installation testing owning to the difficulties encountered in production process design and mold making.

There were altogether 22 *Change Request Forms* (CRF) approved for the entire redesign/EC process. On the basis of project schedule and planned milestones, a CRF should include as many components as possible for approval to save administration costs and potential change propagation. However, those critical and problematic components were supposed to be processed individually in separate CRFs to avoid significant and costly delay. Information related to technical data, price and investments, timing, drawing, and 3D data will be gathered for the preliminary evaluation.

Once initiated, a CRF would be circulated between individual departments (i.e., Product Engineering, Manufacturing Engineering, Quality Assurance, Plant, Purchasing, Finance, Sales

& Marketing, etc.). Approval time depends on the technical and manufacturing complexity of affected components and the availability of influenced team members. For small changes such as to a bolt or a rack, the approval process usually takes only two to three days. However, for changes to more complex components such as oil pump, it may require more than a week. Even longer approval time, including several rounds of meetings with key stakeholders, is necessary for critical components.

4.3 NPD Process in Company B

Company B is an automotive steering system supplier to more than ten well–known Chinese automotive manufacturers, among which the most famous one is the joint venture of Nanjing Automobile Group Corporation with a European automaker. Three major series: i) manual module, ii) hydraulic power module, and iii) electric power steering, and nearly a hundred different kinds of steering systems are designed, developed and assembled in this company with an annual production capacity of nearly 300,000 sets.

A typical power rack and pinion steering system, as seen in *Figure 24,* contains around 105 product components, around 20% of which are produced in house while fabrication activities of the rest are subcontracted to more than 30 second– or third–tier suppliers before they are shipped back to the home site for test operations and final assembly.

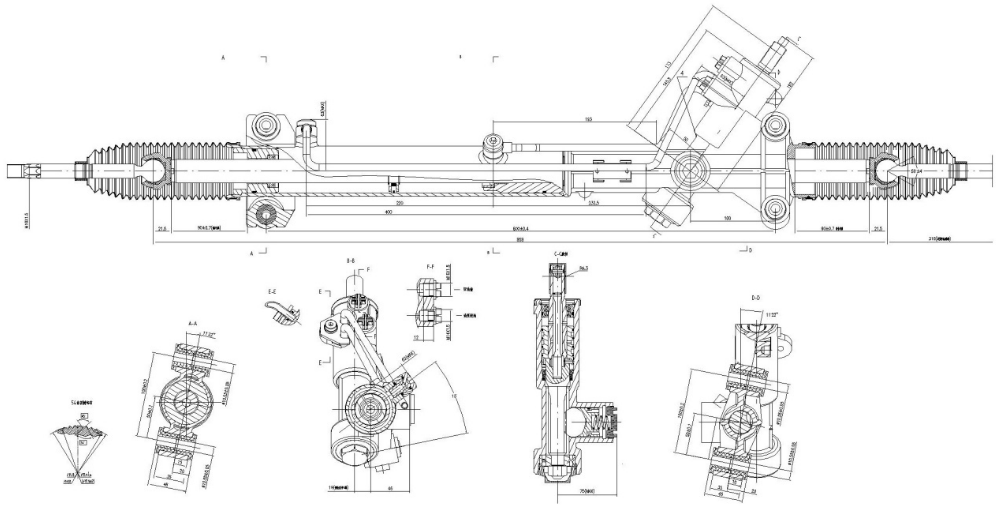


Figure 24: Typical Power Rack and Pinion Steering System

(A copy of mechanical drawing provided with permission by *Company B*)

4.3.1 NPD Process Data

Table 7 displays a list of phases and activities, estimates of duration, and resource loading of typical NPD projects of designing, developing and manufacturing steering systems. All the information and data appeared in this table are based on three primary sources: i) process documents provided by the company; ii) informal conversations and formal interviews with chief engineer, product engineers, and project managers; and iii) questionnaires filled out by project managers. In *Company B*, NPD projects are generally classified into two types according to the project budget: **A** for large or midsize development projects with a budget *greater than $20K*, and **B** for smaller/enhancement projects with a budget *less than $20K*.

Table 7: Activity List and Data of the NPD Process of an Automotive Steering System

Phase	Activity	Description	Estimated Activity Duration (A/B)[12]	Resource Requirements[13] & Involvement Percentage[14]	Stage–Gate Point
1. Project Plan (2 weeks)	***1.1 Customer Requirements Collection***	• Communication with Customers; • Formal/Informal Meetings	3 Days /2 Days	Product Engineer *1–2–3–4–5* (30%) Marketing *1–2–3–4–5* (20%)	Project Plan Approval
	1.2 Planning	• Market Analysis; • Government Requirements and Regulations; • Competitor Analysis; • Similar Project Experiences; • Technology Evolution Analysis	3 Days /2 Days	Marketing *1–2–3–4–5* (100%)	
	1.3 Cost Analysis	• Financial Negotiation; • Cost and Investment Analysis	3 Days /2 Days	Marketing *1–2–3–4–5* (70%, 20%)	
	1.4 Project Setup	• Project Team Building; • Project Scheduling; • Responsibility Allocation Matrix	5 Days /5 Days	Engineering *1–2–3–4–5* (50%) General Management *1–2–3–4–5* (50%)	
2. Conceptual Design (2 weeks)	***2.1 Data Gathering***	• Output of the Project Plan phase; • Project Technical Documents Transferred from Customer	5 Days /3 Days	Product Engineer *1–2–3–4–5* (100%) Marketing *1–2–3–4–5* (20%)	Design Input Review and
	2.2 Technology			Engineering *1–2–3–4–5*	

[12] For simplicity, difference in level of effort between type A and B is only reflected in the activity duration estimations. Resource requirements and the corresponding involvement percentages are assumed to remain the same for both types of projects. Two numbers of *Estimated Activity Duration* are provided by the project manager for each activity according to project classification: A or B, respectively. For example, the average duration of activity "1.2 Planning" in Project Plan phase for A projects is 3 days, and it is around 2 days for B projects.

[13] This column records the type and amount (highlighted in yellow) of resources engaged in this activity.

[14] *Involvement Percentage (IP)* is the proportion per hundred of an individual's (a unit of resource) total effort that is dedicated to an activity. When there are more than one unit of resource allocated to the activity, estimates of IPs should be provided for each one respectively.

	Transformation; Design Plan Draft Formulation		4 Days /2 Days	(100%)	Approval
	2.3 Customer Confirmation; Design Plan Formulation		5 Days /3 Days	Engineering *1–2–3–4–5* (100%) Marketing *1–2–3–4–5* (10%)	Design Plan Approval
3. Product Design and Development (3 months)	***3.1 Product Plan Formulation***	• Product Plan; • Design Calculation Sheet	15 Days /10 Days	Product Design Engineer *1–2–3–4–5* (100%)	Product Plan Approval
	3.2 Detail Design	• Design Failure Mode Effects Analysis (DFMEA); • DFMEA Check List; • Engineering Drawings; • Casting Drawings; • Bill of Material (BOM); • List of Special Product and Process Characteristics; • List of Parts and Sub–Systems; • Testing Schedule; • Technical Regulations	25 Days /20 Days	Product Design Engineer *1–2–3–4–5* (100%, 50%, 20%)	Processing Plan Draft Approval
	3.3 Prototyping (3–5)		40 Days /40 Days	Plant Engineer *1–2–3–4–5* (20%, 20%, 20%, 20%, 20%) Process Engineer *1–2–3–4–5* (20%, 20%, 20%, 20%, 20%) Product Design Engineer 1–2–3–4–5 (10%) Quality 1–2–3–4–5 (20%)	Prototype Approval
	4.1 Processing Plan Formulation		10 Days /10 Days	Process Engineer 1–2–3–4–5–6–7 *(50%, 50%, 50%, 50%, 50%, 30%, 20%)* Plant Engineer 1–2–3–4–5	

4. Process[15] Design and Development (3–4 months, overlapped with Phase 3)				*(10%, 10%, 10%, 10%, 10%)*	Processing Plan Approval
	4.2 Process Flow Chart Formulation		2 Days /2 Days	Process Engineer 1–2–3–4–5 *(50%, 50%, 50%, 50%, 50%)*	
	4.3 Process Failure Mode Effects Analysis (PFMEA)		5 Days /5 Days	Process Engineer *1–2–3–4–5–6–7* (80%, 80%, 80%, 80%, 80%, 50%, 30%) Quality *1–2–3–4–5* (80%)	
	4.4 Tooling Development; Processing Plan Approval	• New Equipment, Tooling and Facility Requirements	40 Days /40 Days	Plant Engineer *1–2–3–4–5* (20%, 20%, 20%, 20%, 20%) Process Engineer *1–2–3–4–5* (20%, 20%, 20%, 20%, 20%) Product Design Engineer *1–2–3–4–5* (100%) Quality *1–2–3–4–5* (20%, 20%,20%)	
	4.5 Internal & External Logistic Plan Formulation	• Tooling Design; • Sample Manufacturing/ Purchasing; • Sample Testing and Review; • Purchasing	Not typical, 15 days when necessary	Purchasing *1–2–3–4–5* (30%) Process Engineer *1–2–3–4–5* (20%, 20%)	
5. Off–Tool Sampling (3 months,	***5.1 Technical Documents Formulation***	• Confirmation of Manufacturing and Controlling Methods	13 Days /8 Days	Process Engineer *1–2–3–4–5–6–7* (80%, 80%, 80%, 80%, 80%, 50%, 30%)	OTS
	5.2 OTS Manufacturing (5–30 units)		40 Days /40 Days	Plant Engineer *1–2–3–4–5* (30%, 30%, 30%, 30%, 30%) Process Engineer *1–2–3–4–5* (20%, 20%, 20%, 20%, 20%) Product Design Engineer *1–2–3–4–5* (5%) Quality *1–2–3–4–5* (50%)	

[15] Process Design and Development typically starts when its predecessor phase, Product Design and Development, is half-way done. It is also commonly overlapped by its successor development phase, Off-Tool Sampling.

overlapped with Phase 4)	***5.3 Sample Testing***		3 Days /3 Days	Plant Engineer *1–2–3–4–5* (30%, 30%, 30%, 30%, 30%) Process Engineer *1–2–3–4–5* (20%, 20%, 20%, 20%, 20%) Product Design Engineer *1–2–3–4–5* (5%) Quality *1–2–3–4–5* (50%)	Approval
	5.4 Sample Reliability Testing		30 Days /30 Days	Process Engineer *1–2–3–4–5* (10%)	
6. Pilot Production (2 weeks)	***6.1 Planning***	• Technical Documents Confirmation; • Sample Approval; • Pilot Production Plan Formulation	5 Days /5 Days	Process Engineer *1–2–3–4–5–6–7* (80%, 80%, 80%, 80%, 80%, 50%, 30%)	Pilot Production Approval
	6.2 Manufacturing Process Approval		10 Days /10 Days	General Management *1–2–3–4–5* (50%, 50%, 50%, 50%, 50%)	
	6.3 Pilot Production (30–200 units)		Not for sure (Depends on sales).	Plant Engineer *1–2–3–4–5* (30%, 30%, 30%, 30%, 30%)	

4.3.2 Observations and Reflections

1) Due to the strict regulations of auto steering industry and relatively stable and well–developed product architecture and technology, EC is not a very common phenomenon that the subject company will encounter after the release of design. It is also observed that average resource consumption of ECM is much fewer as compared to regular NPD activities. And so is the average EC lead time.
2) Three major causes of ECs that occur mostly during the Product Design and Development phase and Process Design and Development phase are: i) new customer requests, ii) manufacturing cost reduction without sacrificing product performance, and iii) error correction in design.
3) Along the supply chain, an upstream supplier is more likely to initiate and also benefit later from the *"push–type"* (innovation–oriented or improvement–oriented) IECs. Such a company usually has greater flexibility in design, and stays more motivated as well, to improve the product function and performance or adopt new technology advances by handling ECs, as compared with downstream suppliers/ manufacturers, in which more *"pull–type"* (error correction–oriented and cost saving–oriented) EECs occur.
4) Even though this company has a well–structured ECM procedure that describes the formal coordination steps to be applied on any incoming ECs to the product or process design, observations indicate that there are quite a large number of possible execution sequences of ECM workflow in the actual processing of an EC, depending on its complexity, and resource availability and change progress status at that moment.
5) In the subject organization, all ECM documents, including mechanical CAD drawings, BOM, production resources, shop floor planning, Material Requirements Planning (MRP),

Supply Chain Management (SCM) in terms of inventories and orders, etc., are organized and tracked in a central computerized documentation management system. However, only a very small portion of the overall ECM knowledge is captured and able to be later retrieved. Most tacit and unstructured communication among relevant personnel and knowledge of the problem solving process are no longer retained after the EC is approved and implemented.

4.4 Change Request Management in Company C

This section summarizes the field study conducted in the Information Technology (IT) division at a Fortune 500 US company in the summer of 2011. In *Company C*, new projects and major updates to the product that included new features and services are released on a monthly basis. A variety of development methodologies are adopted that cover the spectrum from the traditional plan–driven side to the newly emerging design–centered adaptive side. To be more specific, most teams deliver products following a sequential waterfall (also known as “stage–gate”) PD process or a combined waterfall–iterative method, while a few pilot teams adopts agile (also known as “scrum”) principles and practices. Data were collected under three main topics: i) change request arrival patterns, ii) change request approval process, and iii) agile development process.

4.4.1 Change Request Arrival Pattern

A *Change Request* (CR) should be submitted for review whenever the project scope, schedule, cost, architecture, or quality of the baseline work plan is affected by the proposed change once the scope plan of a release is frozen, typically 4 months prior to the scheduled release date. And this time interval is called *Change Request Management* (CRM) period. In the subject organization, CRs are recorded, processed, and archived in an own proprietary software–based CRM system. To respect the company's confidentiality policy, any CR information about actual product lines or activity content of the changes will not be included in this report. Only numeric data related to date, counting of numbers, and estimates of effort in hours are presented.

Figure 25 shows the arrival pattern of CRs in form of histogram. Numbers appeared in the graph are work days between the date a CR is submitted to the central CRM tool and its target release date. This time interval is recognized as the CR *"cycle time"*. Since the CRM tool was put into full use from the beginning of 2011, data were gathered from altogether 442 CRs over a period for about 8 months (01/05/2011 – 08/17/2011).

Figure 25 indicates that CR cycle time has an average of *55.8* days with a standard deviation of *25.7* days. The normality test rejects the hypothesis of normality (i.e., the data don't fit the normal distribution) since the p–value is less than 0.005. To a large extent, it is due to the sensitivity of Anderson–Darling test to extreme values (e.g., the upper outliners in the graph). *Figure 25* also shows that the most frequent occurring value of CR cycle time is around 30 days, which is worth giving attention. Recall that cycle time is the time difference between submission and implementation of a CR, which includes not only the resolution and implementation of a CR by requirement analysts, system architects, programmers and testers but also the whole approval process among the CRM Committee. This issue, among others, will be later discussed in detail.

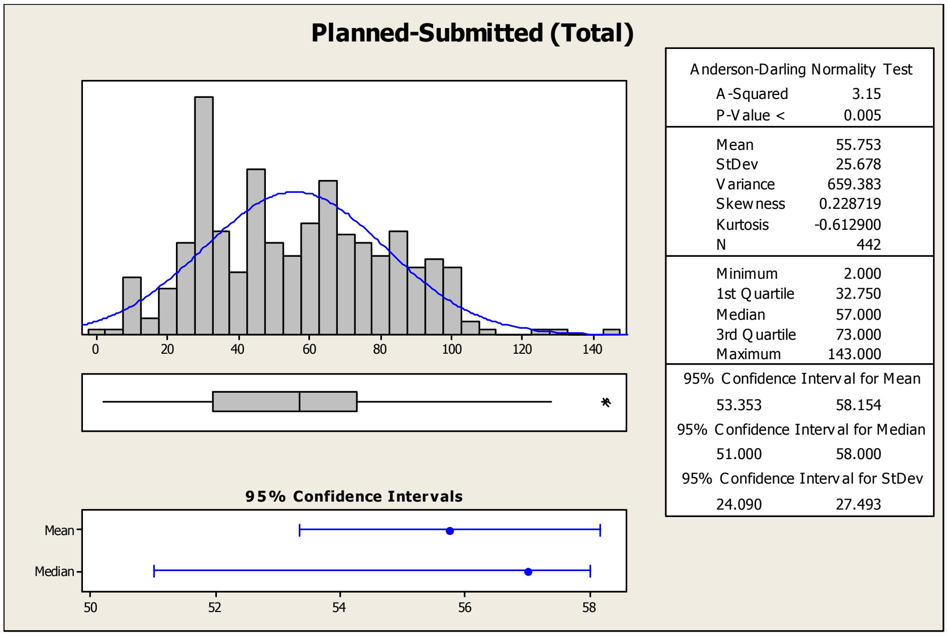


Figure 25: CR Cycle Time Statistics

In addition to a high–level overview of the aggregate CR data presented in *Figure 25*, more detailed information, such as number of incoming CRs by types and submitted/approved CR effort estimates in hours, is further grouped by monthly releases as shown in *Table 8*. Monthly releases from *May* to *August* are analyzed because raw data on these four releases are complete, but only partial CR data are available for previous ones.

CRs are classified into two types: *adds* (increased scope of activities/deliverables) and *removes* (deletion or delay of activities/deliverables). Status of a CR, as appeared in the table, can be one of the following: **(a)** "Approved w/ Documents Updated"; **(b)** "Approved"; **(c)** "Pending"; or **(d)** "Disapproved". Since all of the four releases have been closed at the time of data were extracted, only statuses **(a)** and **(d)** are valid.

Table 8: Summary of CR Data by Release

	May Release	**June Release**	**July Release**	**August Release**
Date Range of Data Have Been Recorded	1/13/2011–5/6/2011	2/17/2011–6/3/2011	3/24/2011–7/8/2011	4/21/2011–8/5/2011
Total # of CRs[16]	59	62	60	101
# of CR–Adds[17] **((a) / (d))**	40 (**37** / 3)	37 (**28** / 9)	46 (**41** / 5)	62 (**52** / 10)
# of CR–Removes **((a) / (d))**	19 (18 / 1)	25 (24 / 1)	14 (14 / 0)	39 (32 / 7)
Total Submitted Hour Estimation[18]	**6860**	**6799**	**4996**	**8173**
Total Approved Hour Estimation[19]	**6506**	**6020**	**4005**	**7332**

Besides the comparison information of the four releases provided in *Table 8*, *Figure 26* further details the raw data on CR arrivals and effort estimations of submitted and approved CR–Adds versus time (in workdays).

[16] This row represents the total number of CRs, including both adds and removes.
[17] Figures in Blue represent the total number of approved CR-Adds (histogram of arrivals is shown in the top parts of *Figure 26*).
[18] This row represents the total man-hour effort estimates of all the submitted CR-Adds (histogram as shown in the middle parts of *Figure 26*).
[19] This row represents the total man-hour effort estimates of approved CR-Adds (histogram as shown in the bottom parts of *Figure 26*).

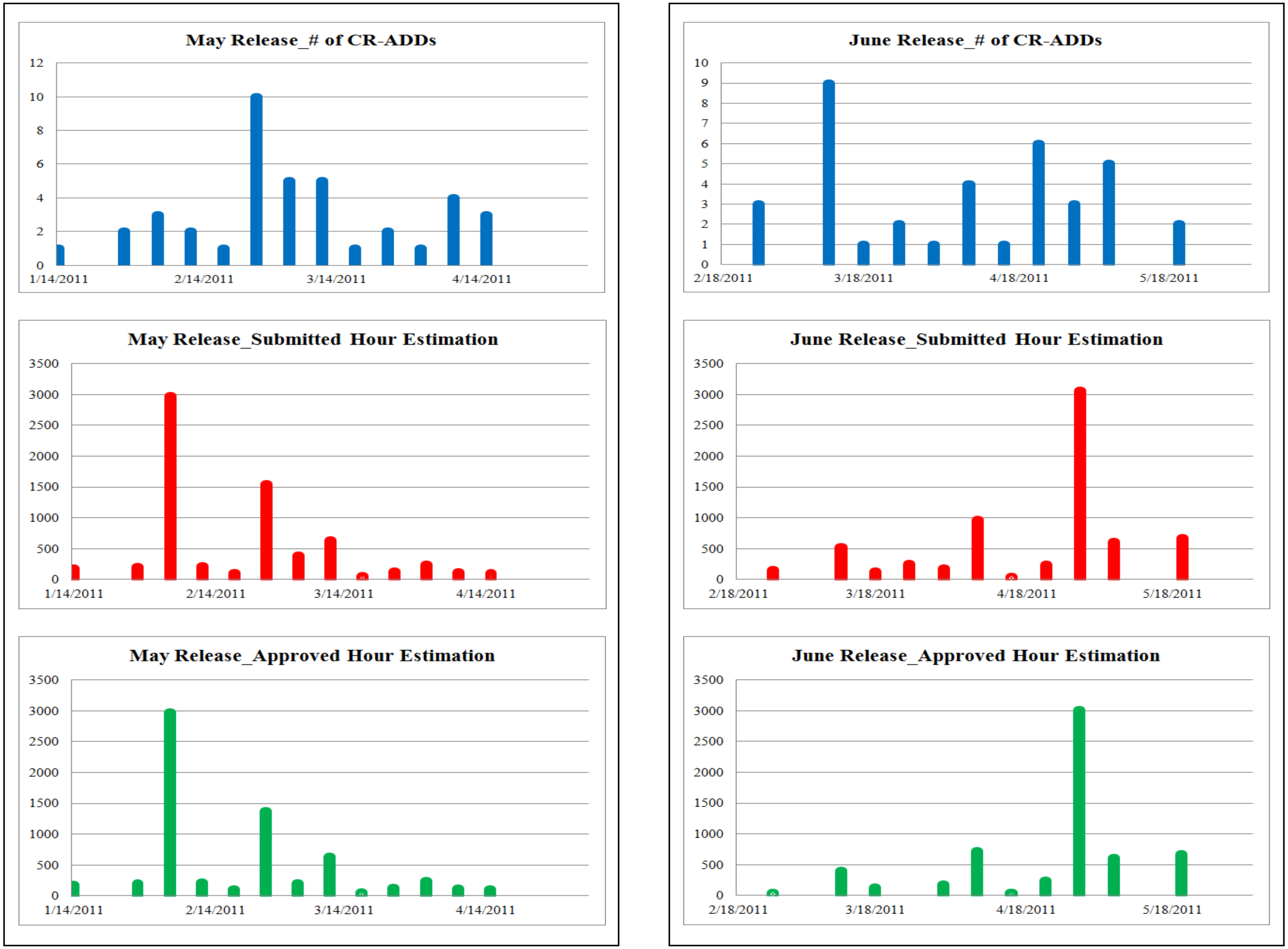


Figure 26: CR–Adds Data along Time by Release

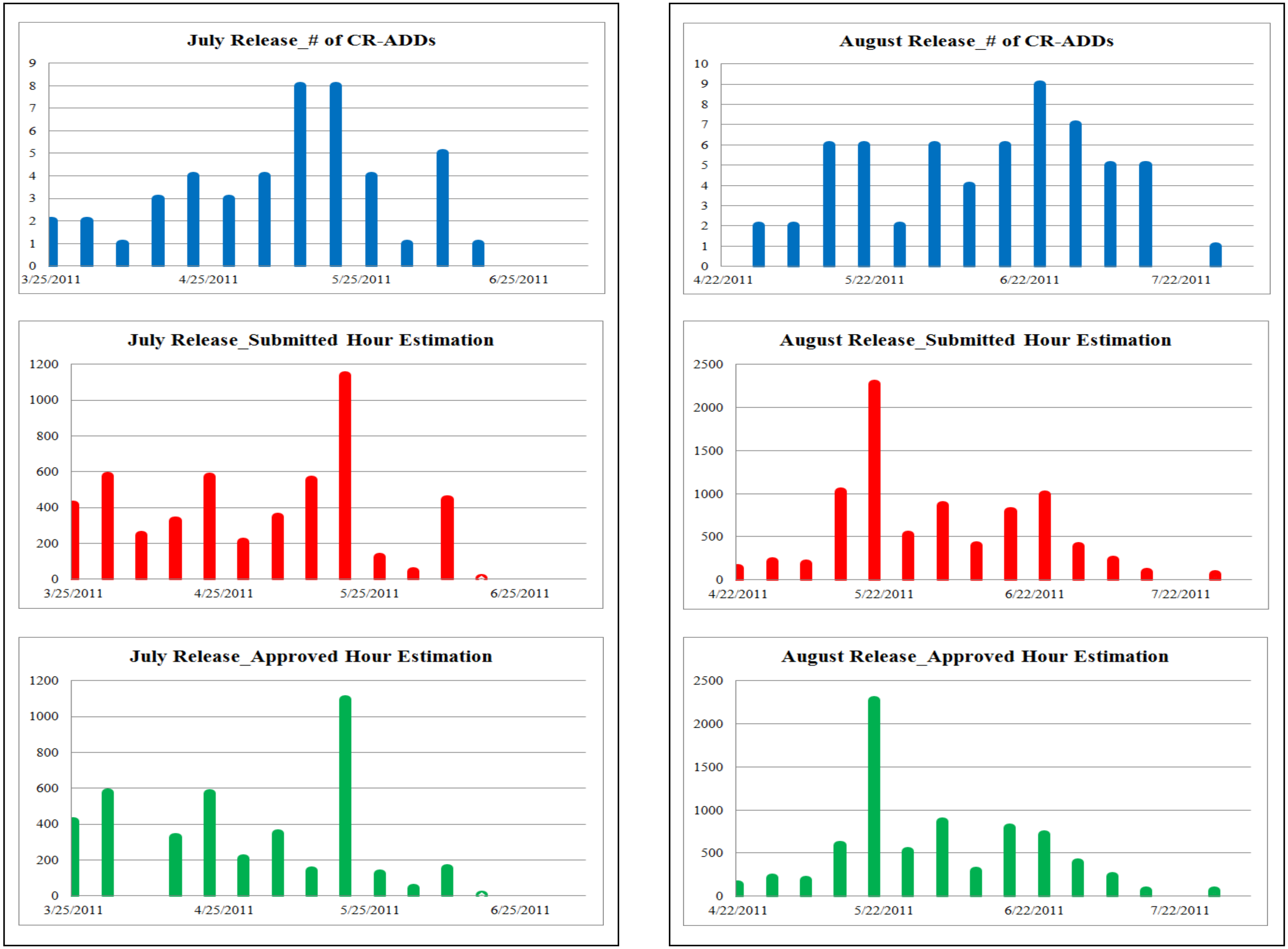


Figure 26: CR–Adds Data along Time by Release (Cont'd)

Table 9: Percentage of Approved Hour Estimation by Months before Release

Months before Release	**May Release**	**June Release**	**July Release**	**August Release**
Work Plan Scope Lock Date				
4	56.6%	9.4%	47.2%	46.4%
3	34.6%	15.5%	44.8%	33.7%
2	8.7%	**64.0%**	8.1%	**19.1%**
1 (/3 weeks)	0.0%	**11.0%**	0.0%	0.8%
0: Release Date				

Table 9 compares the percentages of estimated man–hour efforts of approved CR–Adds by release countdown, i.e., 1(3 weeks in some cases) to 4 months before release. According to the data, a majority of approved CR effort estimates (in units of hours) were identified in the first half of the CRM period except June Release, which is way more challenging than expected in a sense of "final firefighting" that requires significant additional resources in processing major emergent changes. Around 75% of total CR effort estimates were identified within two months before the scheduled June Release date. The August release also shows some challenging aspects by having approximately 20% of CRs identified within two months before the release date.

4.4.2 Change Request Approval Process

A disguised version of CR approval process is illustrated in *Figure 27*. Circulation of individual approval decisions among CRM committee members is facilitated by automated notification emails sent by the CRM software tool.

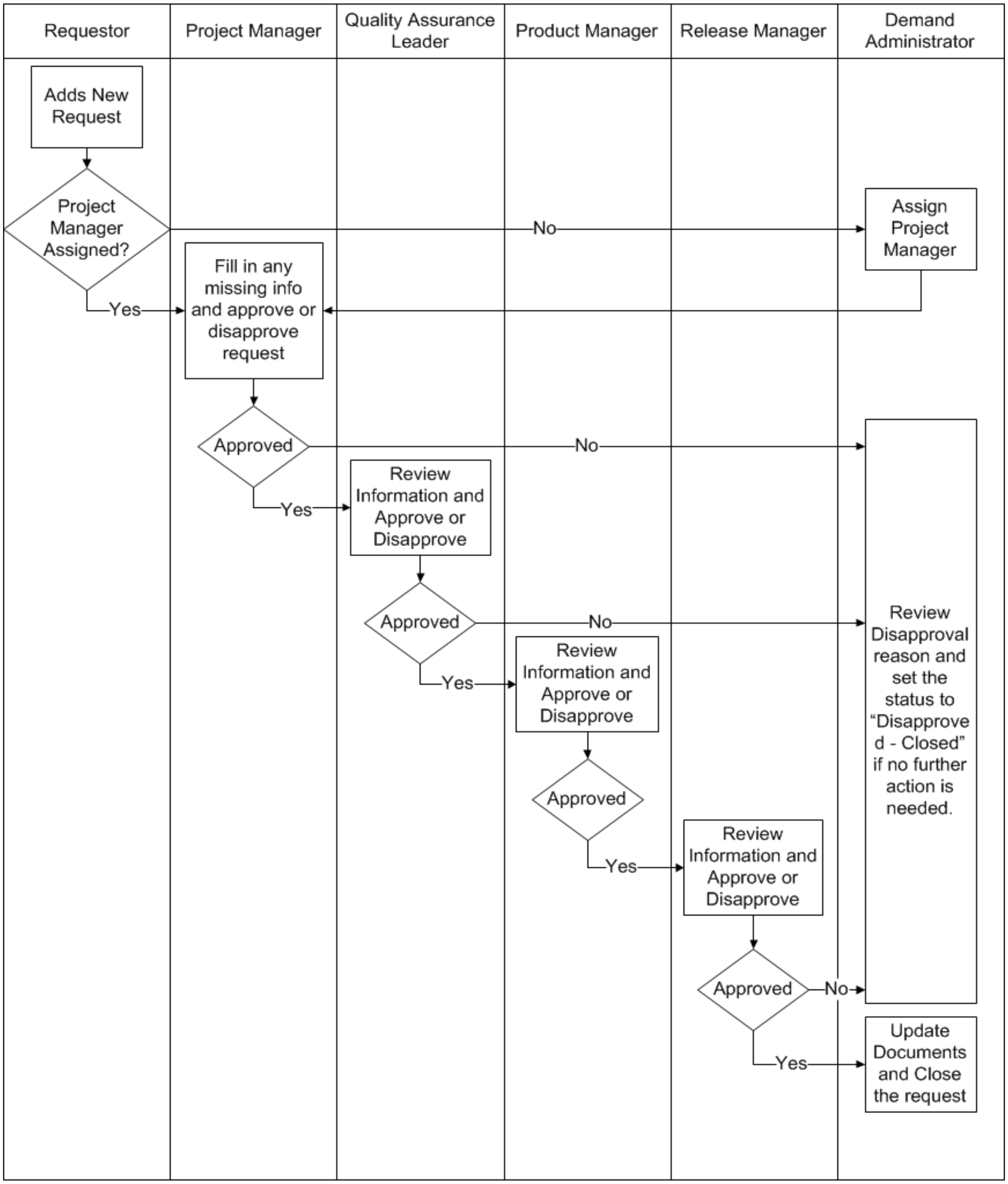


Figure 27: CR Approval Process

(A disguised version of CR approval flowchart provided with permission by *Company C*)

The flow begins with an incoming request from the CR initiator (mainly proposed by business leaders according to new needs and a few raised by developers or testers due to errors) along with cross impact estimates in terms of effort hours that are together determined by Demand Management, and ends up also in the Demand Administrator with the final decision of either approval or rejection. The CRM committee is composed of key personnel representing related support function areas: Project Manager, Quality Assurance Leader, Product Manager, Release Manager, and Demand Administrator.

The approval time of each committee member is then automatically recorded by the CRM tool, from which the time interval between every two individual approval decisions (which is the calendar duration instead of the amount of time it actually takes to process) can be obtained. Again, the numbers shown in *Table 10* (i.e., maximum, average, and standard deviation of approval duration) are based on data collected from 442 CRs over a period for 8 months.

Table 10: CR Approval Durations by CRM Committee

	(a) Approved w/ Documents Updated						**(d)** Disapproved[20]
Duration *(Work Days)*	Project Manager	Quality Assurance Leader	Product Manager	Release Manager	Demand Administrator	**Total**	**Total**
Max	27	30	22	70	12	84	40
Avg	**3.49**	**3.53**	**2.56**	**3.71**	**3.14**	**12.33**	**11.05**
StDev	4.48	4.09	2.79	5.70	3.32	10.37	9.43

Figure 28 shows the distribution of individual approval status by taking a close look at the 44 disapproved CRs (out of a total number of 442). Note that CRM committee members may mark disapprovals simultaneously on one CR. Also, a disapproved CR is not necessarily resulting from a clear “Disapprove” decision (as indicated by red color in the graph). Some CRs with status “Pending” (as seen in green) ended up getting disapproved.

[20] Since the CRM software tool doesn’t record date of disapproval decisions made by CRM committee members, only Total Duration information is available for disapproved CRs.

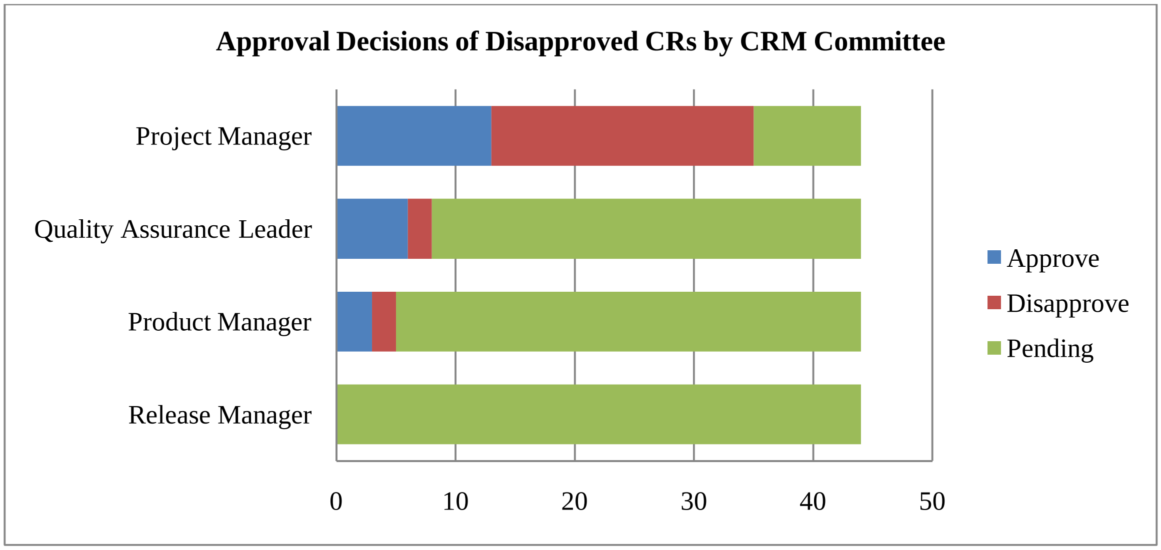


Figure 28: CR Approval Decisions by CRM Committee

To obtain a deeper understanding of the causes of long CRM approval process and the main reasons behind disapprovals, face–to–face interviews were arranged with key CRM committee members from representative function areas. Please see *APPENDIX A* for the complete list of open–ended interview questions. There are several general insights that can be drawn from the numeric information presented above and the feedback from interviewees:

1) CRM committee members are not devoted to handling CRs. They typically spend only a few hours in reviewing cumulated CRs in their mailbox on a weekly basis, in contrast to the average individual approval (calendar) duration of *3.29* work days.
2) There are two most frequently mentioned road blocks that are commonly experienced before a CR approval decision can be made: (i) funding constraints or scarce resource capacity, and (ii) lack of assessment of multiple cross impacts associated with the CR (i.e., the demand process is not completed).

3) Project managers play an important role in holding CRs' approval mainly owing to their concerns about (i) high volume of CRs that greatly affect team capacity, (ii) unclear requests without estimates or impact assessment, and (iii) dependency issues (e.g., coding content dependent on other teams' work).
4) On the other hand, product managers expressed their anxiety about project managers' and quality assurance leaders' approval decision under incomplete knowledge of the change and associated technical difficulties may be encountered down to the product level. If product manager disapproves the CR, it will go all the way back to the CR initiator for another round of administrative processing which leads to a considerable longer throughput time of approval and evaluation.
5) Product managers also suggest more partnerships with business in handling CRs, i.e., informal but effective face–to–face conversation and corporation among parties at the working level, instead of purely relying on the information system tool as the only means to communicate, negotiate, record, and track CRs.

4.5 Summary

This chapter presents results obtained from three field studies that were conducted during the summers of 2010 and 2011 regarding the current practice of NPD and ECM in two typical areas for change management: manufacturing and software development industry. Data of the development project concerning product, process, team, and environment were collected. Findings are based on the both qualitative and quantitative analyses of on–site observations, documentation review, companies' historical data, and informal or structured interviews.

CHAPTER 5

MODEL DESCRIPTION

5.1 Introduction

Based upon the causal relationships of iteration and EC occurrences due to different levels of evolving uncertainty identified in *Chapter 3* and the field survey findings discussed in *Chapter 4*, this portion of dissertation introduces the building blocks of the discrete event simulation model proposed by this research and the underlying logic of model structure governing the relationships between variables in greater and more precise details.

This chapter begins with a brief introduction of the notation and an overview of general assumptions and properties of the model. Development of the two major model components (NPD sector and IEC sector), their working mechanisms, along with the mathematical formulation of critical model variables that link the two components together are then presented.

5.2 General Assumptions and Model Properties

This model has two constituent sections:

1) ***NPD Section with Reworks*** (i.e., iterations and EECs), and
2) ***IEC Section***

It incorporates three levels of uncertainties that are critical to NPD and ECM processes: i) low–level *activity uncertainty* represented by the stochastic activity duration (i.e., value–added processing interval), ii) medium–level *solution uncertainty* that dynamically calculates rework probability, and iii) high–level *environmental uncertainty* captured by the arrival frequency and magnitude (i.e., units of resource required) of IECs.

Primary model assumptions underlining are listed below.

1. The overall structure of NPD process can be systematically planned beforehand in an activity–based representation according to historical data from previously accomplished projects of similar products and teams' expertise as well. All NPD phases and activities, their expected durations and units of resource required, and interdependencies relationships among them are obtainable and remain stable as the NPD project evolves. Therefore, optimization of process sequencing and scheduling is not pursued by this research.
2. There is no overlapping between activities within a same phase. An NPD activity only receives finalized information from its upstream activity within one phase, but downstream action can start with information in a preliminary form before all activities in upstream phase are completed. In addition, there is no information exchange in the middle of an activity.
3. Demand on resource for an NPD activity is assumed to be deterministic fixed. However, the activity duration varies stochastically subject to both activity uncertainty and learning curve effects which improve as the number of attempts to that particular activity increase until an upper limit.
4. The dynamic progress of an NPD project is reflected by the work flow within and among NPD phases. Workflow routing is probabilistically altered by either intra–phase iterations

or inter–phase EECs according to the dynamically updated rework probability, which is calculated based on the current value of solution uncertainty.

5. Each IEC is initially associated with a directly affected NPD activity (and a directly affected product component when product structure is modeled), and may further propagate to any downstream activities according to randomly assigned probabilities. IECs are modeled within a parallel co–flow structure similar to the NPD counterpart. IEC work flow is restricted by precedence constraints of the original NPD process.

5.3 Notations

Based on these general assumptions and model boundary, notations of important model parameters and variables used in the mathematical formulation of the model are introduced as follows.

5.2.1 Model Parameters

I: number of NPD phases

J_i: number of NPD activities within phase i (for $i = 1, 2, \dots, I$)

M: number of participating departments

R_m: total number of resources available from department m (for $m = 1, 2, \dots, M$)

r_{ijm}: units of resource required from department m to complete activity j (for $j = 1, 2, \dots, J_i$) in phase i

d_{ij}:[21] time expected to complete activity j in phase i when resource requirement is met

D_{ij}: mean value of d_{ij}, $D_{ij} = \beta k$

[21] The Erlang distribution $ERLANG\ (\beta, k)$ is used as a description of NPD activity duration

5.2.2 Model Variables

i_t/j_t: the latest–finished activity basework j_t in phase i_t at time t

n_t: number of reworks finished at time t

x_1/y_1: the first rework for activity y_1 in phase x_1

x_{n_t}/y_{n_t}: the latest–finished rework for activity y_{n_t} in phase x_{n_t} at time t

$(R_m)_t$:[22] the cumulative functional effort of the ongoing rework(s) at time t

L_t: number of IECs finished at time t

g_{l_1}: the activity in which IEC l is initiated

$g_{l_{G_l}}$: the last propagated activity of IEC l number of activities IEC l propagates to ($G_l \leq I \times J_i$)

s_{lgm}: resources required from department m to complete IEC l (for $l = 1, 2, \dots, L_t$) to activity g (for $g = g_{l_1}, g_{l_2}, \dots, g_{l_{G_l}}$)

w_{lg}:[23] time expected to complete IEC l to activity g

$(I_m)_t$: [24] the cumulative functional effort of the ongoing IEC(s) at time t

5.4 Design Solution Scope

Design Solution Scope is defined as the overall extent of an NPD project. It is measured in terms of total effort required (person–days), by completing of which the entire set of product goals will be met. It depends not only on the number of constituent activities, but also the expected duration and units of resources needed to produce the desired outputs of each activity. In a sense, design solution scope indicates one facet of the NPD project complexity with regards

[22] An aggregate term consists of ongoing rework(s)/rework propagations each one corresponding to its current stochastic functional effort value.

[23] The Triangular distribution $TRIANGULAR\ (Min, Mode, Max)$ is used as a description of IEC duration.

[24] An aggregate term consists of ongoing probabilistically dependent IEC(s)/IEC propagations each one corresponding to its current stochastic functional effort value.

to its content (as a function of activity duration d_{ij} and demand for resource r_{ijm}). Of course, project complexity is also indicated by its architecture (i.e., the coupling among product components or the process precedence constraints), which will be discussed more in later subsections on the topics of overlapping and rework probabilities.

The estimated functional effort to complete the whole NPD project is obtained as follows:

$$EN_m = \sum_{i=1}^{I} \sum_{j=1}^{J} e_{ijm} = \sum_{i=1}^{I} \sum_{j=1}^{J} (r_{ijm} \times d_{ij}) \tag{1}$$

Let's assume that L_t is the total number of incoming IECs that have been processed at time t, g_{l_1} is the activity to which a randomly occurring IEC l (for $l = 1, 2, \dots, L_t$)is directly related, and $g_{l_{G_l}}$ is the last activity along the IEC propagation loop. Through the estimation of IEC duration w_{lg} and s_{lgm} number of resource required from department m, the functional effort needed to process IEC l to activity g (for $g = g_{l_1}, g_{l_2}, \dots, g_{l_{G_l}}$) is $e_{lgm} = s_{lgm} \times w_{lg}$. By a double summation over both l (of the entire set of completed IECs) and g (including the original incoming IEC and a sequence of its propagations), the cumulative functional IEC effort at time t can be represented as

$$(EI_m)_t = \sum_{l=1}^{L_t} \sum_{g=g_{l_1}}^{g_{l_{G_l}}} e_{lgm} + (I_m)_t = \sum_{l=1}^{L_t} \sum_{g=g_{l_1}}^{g_{l_{G_l}}} (s_{lgm} \times w_{lg}) + (I_m)_t \tag{2}$$

Note that besides the first term $\sum_{l=1}^{L_t} \sum_{g=g_{l_1}}^{g_{l_{G_l}}} e_{lgm}$ which describes the total functional effort spent on those already completed IECs, another aggregate term $(I_m)_t$, which represents the cumulative functional effort of the ongoing IEC(s) at time t, is used to avoid the inherently tedious expression of such a set of stochastic, probabilistic, and discrete events in a mathematical formula. Difficulties encountered here in translating these random occurrences into a precise math equation, once again, confirm the advantages of using computer simulation as the research methodology in studying the interrelated and dynamic ECM problems.

Based on EN_m and $(EI_m)_t$, a dynamic NPD property, ***Functional Design Solution Scope*** $(S_m)_t$, can be obtained as appeared in Eq (3) by making the following assumptions:

1) Design solution scope of an NPD project reflects the amount of effort (in person–days) needed to meet the entire set of product goals, including both original pre–defined goals during project initiation and those additional ones determined along the course of the project[25].
2) Both iterations and EECs are mandatory error–correction oriented type of rework to achieve the same pre–defined goals, and thus there is no overall increase in design solution scope. However, they will be taken into account when calculating the actual cumulative functional effort.
3) IECs are carried out to accomplish additional product goals in response to outside requirements such as altering market demands, growing customer needs, new legislations, or rapid advances in technology. IEC arrivals cause increases of design solution scope.

$$(S_m)_t = EN_m + (EI_m)_t \tag{3}$$

Compared with the original estimate EN_m of planned NPD activities, which is a static project property assessed before the time the project starts, design solution scope $(S_m)_t$ is discretely increasing by taking into account the extra functional efforts devoted to those unplanned IECs at

[25] Therefore, it can be used as a measure of final product quality in later discussion.

any time of occurrence. $(S_m)_t$ will be used later as the functional effort baseline for comparison to calculate the solution uncertainty as discussed in *section 4.3*.

5.5 NPD Framework with Iterations and EECs

From an "*information processing*" view, the generic activity network proposed in (Bhuiyan 2001; Bhuiyan, Gerwin, and Thomson 2004; Bhuiyan, Gatard, and Thomson 2006) is adopted as the fundamental modeling structure. By doing so, the NPD process can be decomposed into *I* numbers of *Phase* P_i $(i = 1, 2, \dots, I)$ with certain degrees of overlapping. Each phase is further made up of J_i sequentially numbered *Activities* $P_i A_j$ $(j = 1, 2, \dots, J_i)$ to represent several chronological stages in design and development. The present study assumes that there is no overlapping among activities within each phase. That is, within a single phase an NPD activity begins only after the completion of its predecessor. However, NPD phases can be overlapped by letting the successor phase begin with only preliminary information before activities in the upstream phase are all finished.

The completion of an NPD activity for the first time is called ***NPD Basework***. Any later attempt, no matter in the form of *intra–phase iteration* or *inter–phase EEC*, is referred as ***Rework***. When work flow is routed back by probability, it is assumed that some of the previously completed activities have encountered errors and the farthest upstream one will be identified as the "starting point" of the rework loop. All the downstream activities are supposed to be "corrupted" and have to be reattempted before the NPD projecy can move on. *Figure 29* illustrates this I – phase and J_i – activity NPD framework.

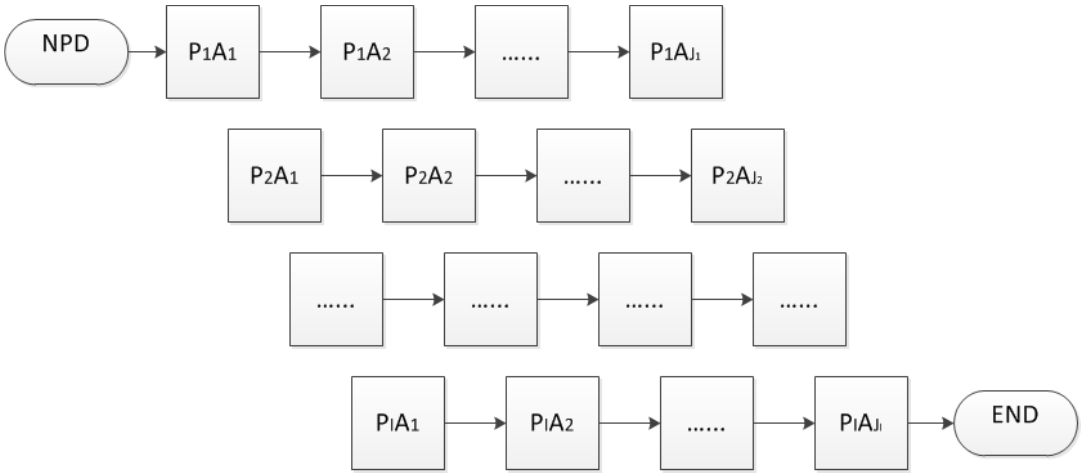


Figure 29: I – Phase & J_i – Activity NPD Framework

It is important to note that since rework is embedded within the same framework as NPD basework, both of them are processed following the same overlapping strategy (i.e., intra– or inter–phase precedence relationship). That is to say, for a sequential NPD process, in which all of the basework are completed one at a time, iterations and EECs are handled sequentially too. For example, if an error in P_2A_3 has been identified after completing P_3A_2, then P_2A_3 and the succeeding activities within phase P_2, together with both $P_3A_1$and P_3A_2, will be reattempted one following another. However, if the NPD baseworks are carried out concurrently, rework will also be handled in the same concurrent fashion when the work flow is routed back according to rework probability. Again, using the previous example, in the situation in which phases P_2 and P_3 are overlapped by executing P_2A_3 and P_3A_1 simultaneously, EECs to P_2A_3 and P_3A_1, and EECs to P_2A_4 and P_3A_2 will also be handled concurrently.

5.5.1 NPD Activity Duration and Learning Curve Effect

Low–level activity uncertainty is represented by the random variation of the activity duration around its estimate. Stated thus, for each NPD activity its duration d_{ij} is sampled from a pre–

determined probability distribution. In this research, the Erlang distribution $ERLANG\ (\beta, k)$ is used as a description of the activity duration. Employment of the Erlang distribution to represent activity interval is based on the hypothesis that each NPD activity consists of k number of random tasks, everyone individually having an identical exponentially distributed processing time with mean β. These mutually independent tasks can be considered as the lowest un–decomposable unit of an NPD process. Number of tasks k comprising each activity and the anticipated task duration β should be estimated by process participants and provided as model inputs.

According to the learning curve theory, the more often an activity is performed, the less time it requires to complete it, and thus the lower will be the cost. This well–recognized phenomenon is considered as a process characteristic to improve the comprehensiveness of this research. As in Cho and Eppinger (2005), ***Learning Curve Effect*** is modeled in the form of a linearly diminishing fraction, $0 < L_f < 1$, of the original duration whenever an activity is repeated until the minimum fraction, $0 < L_{min} < L_f < 1$, is hit and the rework processing time remains unchanged afterward. That is to say, the learning curve improves through each round of rework until it reaches the minimum fraction of the basework duration which is indispensible for the activity execution. Let n be the number of times an activity is attempted, Learning Curve Effect can be expressed as

$$LCE = \max\left(\left(L_f\right)^{N_{ij}-1}, L_{min}\right) \tag{4}$$

And therefore, the processing time of a rework to an NPD activity depends on two variables: the stochastic basework duration d_{ij} of the activity and the number of times N_{ij} it is attempted.

Any types of NPD rework, no matter intra–phase iterations or inter–phase EECs, are subject to the same learning curve effect. The combined effects of rework probability and learning curve on project performance measures will be analyzed in *Chapter 5*.

5.5.2 Overlapping and Cross–Functional Interaction

Overlapping is defined as the partial or full parallel execution of nominally sequential development activities (Krishnan and Ulrich 2001). The underlying risk of overlapping raised by Krishnan that "the duration of the downstream activity may be altered in converting the sequential process into an overlapped process" (Krishnan, Eppinger, and Whitney 1997) is real, but the effect is addressed in a slightly different way from directly increasing downstream duration and effort by a certain calculated value (e.g., Roemer and Ahmadi 2004). The more number of activities start with information in preliminary form or even missing information, the less is the design solution completeness, which will in turn affect rework probabilities as discussed in detail in the next section. The parallel execution of activities is achieved by the use of "Separate" and "Batch" modules.

The concept of cross–functional integration among different functional areas during an NPD process is defined as ***Departmental Interaction***. One of the m departments takes major responsibility for the phase in its own area with specialized knowledge, and is called *Major Department* during that phase. However, the other $m-1$ departments, defined as *Minor Departments*, also need to participate but with less level of resource requirements. Cross–functional integration enables a decentralized NPD process by facilitated communications among involved departments. Similar to the activity duration, recourse consumption in the form of departmental interaction is again an estimate from process participants. Resources can represent staffs, computer/machine, documentation support, or any other individual server. It's assumed that each resource is qualified to handle all the NPD activities within all phases.

5.5.3 Solution Uncertainty

In the process modeling literature, NPD is often considered as a system of interrelated activities that aims to increase knowledge or reduce uncertainty about the final design solution (Krishnan, Eppinger, and Whitney 1997; Browning 1998; Wynn, Grebici, and Clarkson 2011). This research assumes that any knowledge or experience accumulation through an NPD activity, no matter accepted to be transferred to the next activity/activities or rejected for a rework, will contribute to the common knowledge base of the NPD project towards its final design solution. No development effort is ever wasted. In this context, knowledge/experience accumulation is simply measured by the cumulative effort that has been committed to the project in terms of person–days.

Functional Solution Completeness is defined as a criterion to reflect the effort gap between the actual cumulative functional effort accomplished to date and the evolving functional design solution scope $(S_m)_t$. Due to the fact that some activities are attempted by multiple rounds of rework and there are extra efforts spent on IECs, solution completeness may exceed one in later stages of an NPD process.

The exact expression for $(C_{ijm})_t$ is determined by the amount of overlap between NPD activities. The more concurrency a process has, the more complicated the expression will be. Eq (5) is an illustration of solution completeness at time t for the easiest case: a sequential process. It indicates that $(C_{ijm})_t$ is improved by knowledge or experience accumulation through performing NPD basework (indicated by the first term in Eq (5)) and rework (the second term) plus handling IECs (the third term).

A generalized abstract term $(R_m)_t$ is used here to represent the cumulative functional effort of the ongoing rework(s) at time t.

$$\left(C_{ijm}\right)_t = \frac{\left(\sum_{i=1}^{i_t-1}\sum_{j=1}^{J} e_{ijm}+\sum_{j=1}^{j_t} e_{i_t jm}\right)+\left(\sum_{x=x_1,y=y_1}^{x=x_{n_t},y=y_{n_t}}\left(\sum_{i=x+1}^{i_t}\sum_{j=1}^{j_t} e_{ijm}+\sum_{i=x}^{I}\sum_{j=y}^{j_t} e_{ijm}\right)+(R_m)_t\right)+(EI_m)_t}{(S_m)_t} \tag{5}$$

On the contrary, ***Functional Solution Uncertainty*** $(U_{ijm})_t$ reflects the degree of functional effort absence towards the dynamically evolving design solution scope. Therefore, the solution uncertainty of activity j in phase i at time t is

$$(U_{ijm})_t = 100\% - (C_{ijm})_t \tag{6}$$

5.5.4 Rework Probability

After each activity, there is a rework review decision point (or gate) that decides whether the activity output is acceptable and if the NPD project entity gets through or needs to flow back for a rework according to a weighted rework probability determined by the current level of functional solution uncertainty. A critical assumption is made here that the *iteration probability* of an activity is negatively proportional to the NPD project's latest level of solution uncertainty. That is, chance of an activity gets to iterate before it is released to the next phase will increase as the project unfolds with more information available and its solution uncertainty decreases. Two arguments are presented here to backup this assumption: i) as the project unfolds, more information will be available to justify further iteratively refinement of the design solution for each component (Wynn, Grebici, and Clarkson 2011); and ii) since a product often has multiple conflicting targets that may be difficult to meet simultaneously and thus requires further trade–offs, "design oscillations" on a system level may occur due to the interdependencies among local components and subsystems even after the achievement of individual optimum (Clark and Fujimoto 1991; Loch, Mihm, and Huchzermeier 2003). Functional iteration probability is formulated by a negative exponential function of uncertainty as appeared in Eq (7), where $0 < \alpha < 1$ is a process–specific ***Iteration Probability Constant (IPC)*** that should be determined beforehand as a model input:

$$(PI_{ijm})_t = \alpha^{(U_{ijm})_t + 1} \tag{7}$$

Since NPD activities are decentralized through the cross–functional integration among participating departments, so is the decision making process of carrying out rework. The overall iteration probability of activity *j* in phase *i* is the weighted mean by the number of resources each department commits to the activity.

$$(PI_{ij})_t = \frac{\sum_{m=1}^{M}(r_{ijm} \times (PI_{ijm})_t)}{\sum_{m=1}^{M} r_{ijm}} \tag{8}$$

Similarly, *EEC probability* is characterized by an ***EEC Probability Constant (EPC)*** $0 < \gamma < 1$. However, as opposed to iteration probability, it is assumed to be exponentially decreasing as the project's solution uncertainty decreases. That is to say, the chance of revisiting NPD activities, whose outputs have already been frozen and released to their successor phase(s), is the highest after the first activity of the second phase and continuously reduces according to the continually increasing design solution completeness.

$$(PE_{ijm})_t = \gamma^{(C_{ijm})_t + 1} = \gamma^{2 - (U_{ijm})_t} \tag{9}$$

$$(PE_{ij})_t = \frac{\sum_{k=1}^{n}(r_{ijm} \times (PE_{ijm})_t)}{\sum_{k=1}^{n} r_{ijm}} \tag{10}$$

Given the overall rework (i.e., iteration or EEC) probability, the next step is to identify which upstream activity generates the design problem/error/defect disclosed by the rework review and therefore becomes the starting point of correction loop. For simplicity, it is assumed that each upstream activity gets an equal chance of initiating an intra–phase iteration loop or an inter–phase EEC loop. Also, the present study assumes that every activities downstream are contaminated by wrong information from the initiating activity identified, and therefore the rework loop requires redoing the entire set of activities between and including the rework–initiating activity and the one after which the rework is identified.

5.5.5 Rework Criteria and Rigidity of Rework Review

According to the rationale explained in previous sections and causal loop diagrams created in Chapter 3, the occurrences of both iterations and EECs are governed by a combination of balancing and reinforcing loops. Take Loop 7 described in Chapter 3 as an example, the iteration probability of an activity will increase as the solution completeness increases, while redoing the activity will further add to the solution completeness by contributing more information, and thus close a positive feedback loop of the occurrence of iterations.

To avoid the dominance of such reinforcing loops which will eventually lead to a net effect of overall divergence with no termination condition, ***Rework Criteria*** are established as the first step of rework review after the completion of an activity to check whether the cumulative functional effort committed to the deliverable is high enough to provide a satisfying outcome and therefore let the project pass rework evaluation. If the cumulative devoted effort fails to meet the pre–determined criteria (i.e., the cumulative effort is less than the expected amount), the project will be evaluated at the rework decision–point and go for iteration or EEC if necessary according to the rework probability calculated by solution completeness. If the committed effort is higher than the pre–set amount, the NPD project will conditionally pass rework evaluation and continue executing the next activity or group of activities.

Unger and Eppinger (2009) defined *rigidity* by the degree to which deliverables are held to previously–established criteria as metrics to characterize design reviews. By putting it in a slightly different way, rigidity of rework review is considered in this research as the strictness of pre–defined rework criteria with respect to the amount of cumulative effort committed to a particular NPD activity. It is considered as an important NPD process characteristic and will be analyzed later for its impact on key performance indicators.

5.6 IEC Framework

Unlike iterations and EECs, IECs are studied through a different process framework other than the NPD framework. The IEC framework explores how IECs emerging from outside sources after the NPD process begins are handled and how an initiating IEC to a specific activity of a product item will cause further change propagation in its downstream activities and other dependent items.

As described earlier in section 3, since IECs deals with emerging issues and requirements in response to additional project goals that are not anticipated and included in the original design solution scope during project planning, extra functional efforts demanded for handling IECs should be added into the evolving design solution scope.

5.6.1 IEC Processing Rules

IECs affecting activities in different NPD phases are modeled to arrive in randomly at any time after the NPD project starts. A checkpoint is inserted before the processing of an IEC to verify whether the directly affected NPD activity has started yet. Incoming IEC(s) will be hold until the beginning of processing of that particular activity.

On the other hand, during the NPD rework reviews the upcoming NPD activity will also be hold from getting processed if there are IECs currently being handled with respect to any of its upstream activities until new information from these IECs becomes available (i.e., the completion of IECs). The purpose of such an inspection is to avoid unnecessary rework as a result of expected new information and updates. However, an NPD activity will not pause in the middle of its process due to the occurrence of IECs to any of its upstream activities.

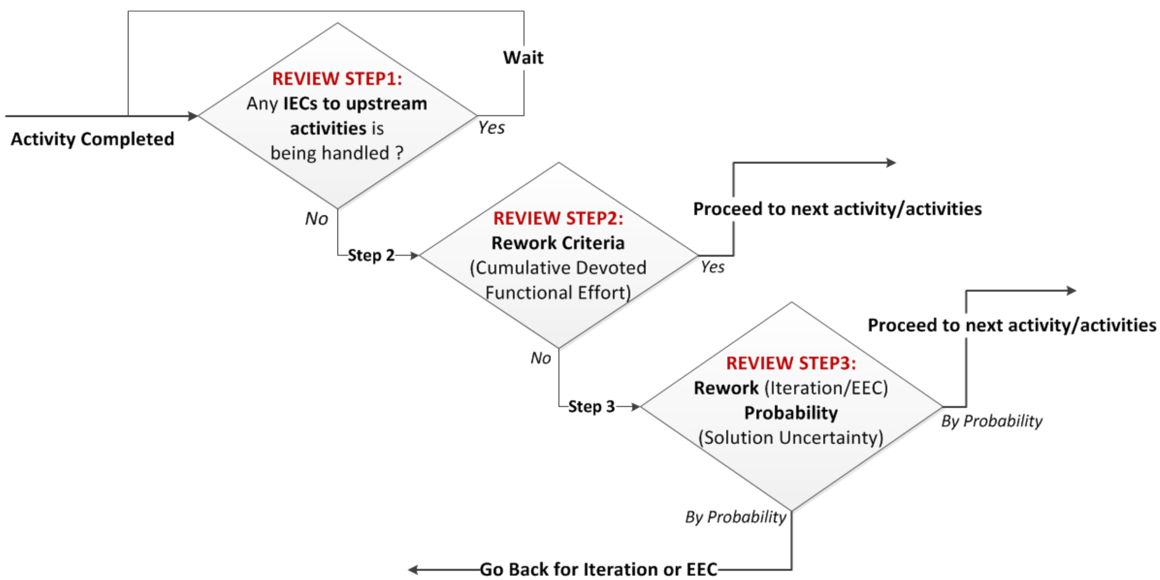


Figure 30: 3–Step NPD Rework Review Process

Figure 30 summaries in detail the entire review process that includes three major steps as discussed before:

1) Check if there are currently any IECs being handled with regards to any of its upstream activities. If the condition is true, then wait until new information from all of these IECs becomes available; if condition is false, then go to the next step.
2) Compare the cumulative devoted functional effort so far to the pre–determined rework criteria. If the condition is true, then the work flow conditionally pass the rework review and directly proceeds to next activity/activities; if the condition is false, then go to the next step.
3) As a result of cross–functional negotiation and integration, calculate rework probability according to the current levels of functional solution uncertainty. NPD project entity will, by probability, either be fed back to the identified activity which contains engineering problems for rework or move to the next activity/activities.

5.6.2 Frequency and Resource Consumption of IEC

Compared with NPDs that are much more likely to adhere to a planned schedule, IECs can occur without any plans. Therefore, the Exponential distribution is used to represent IECs' arrival interval. IEC's processing time is assumed to follow the Triangular distribution, where there is a most–likely time with some variation on two sides, represented by the most likely (Mode), minimum (Min), and maximum (Max) values respectively. The Triangular distribution is widely used in project management tools to estimate activity duration (e.g., Project Evaluation and Review Technique, Critical Path Method, etc.).

The amount of resources required for an IEC to be processed is called ***IEC Effort***. When there are not enough resources available for both processes, resource using priority needs to be assigned to either NPD or ECM to seize necessary resource first.

5.6.3 IEC Propagation

Change Propagation (CP) described in this research is assumed to be rooted in either interrelated *activities* of a PD process or closely dependent constituent product *components and systems*. That is, modifications to an initiating activity or product item are likely to propagate to other activities within the same or different stages along the PD process, and may require further changes across to other items that are interconnected with it through design features and product attributes (Koh and Clarkson 2009).

This phenomenon is simulated by two layers of IEC propagation loop. Firstly, CP review decisions are performed after the completion of an IEC and then propagate to one of its downstream activities by pre–assigned probabilities. We restrict ourselves to only unidirectional change propagation based on process structure. That is to say, an IEC to one NPD activity will propagate only to its successor activities within current or next phase. For example, an IEC to enhance a particular design feature may result in substantial alterations in prototyping and manufacturing. However, innovations in manufacturing process will only cause modifications within production phase but not changes in design.

Secondly, the first–level activity IEC propagation loop is then nested within an outer loop determined by particular dependency properties of the product configuration. Once an IEC to one product item and its CPs to affected downstream activities are completed, it will further propagate to item(s) that is/are directly linked to it.

Partial effects of IECs propagating through activities are explored from *SS 16.3.3* to *SS 16.3.5*, and the impacts of the entire IEC propagation phenomenon will be explicitly discuss in *SS 16.3.6*.

5.7 Summary

To conclude, the discrete event simulation model presented in this chapter identifies several important NPD and ECM process characteristics and translates them into the following eight key aspects of the model mechanism:

- Stochastic activity duration due to activity uncertainty,
- Dynamic, non–linear feedback of solution uncertainty causing rework for activity (can be either intra–phase iterations or inter–phase EECs) due to solution uncertainty,
- Random IEC arrivals due to environmental uncertainty,
- Concurrent and collaborative PD process applying various overlapping and functional integration strategies,
- Learning curve effects,
- Limited resource availability,
- NPD rework review rigidity, and
- IEC propagation due to the couplings of either PD activities or product configuration.

CHAPTER 6

NUMERICAL APPLICATIONS AND RESULT ANALYSIS

6.1 Introduction

In this chapter, a numerical example is presented to illustrate how NPD and IEC sections of the discrete event simulation model discussed in the previous chapter can actually be applied to facilitate policy analysis. A combination of different process, product, team, and environment characteristics are tested through design of experiment. Sensitivity analysis is also conducted to investigate how variations in the model inputs and various parameter settings affect the final model output. NPD project lead time, cost (or engineering effort in some cases), and quality are generated by the model as the three key performance measurements of the project under study to evaluate overall product development efforts.

In particular, impacts of the following managerial strategies and coordination policies on the responses of interest are investigated, and the root causes behind the performance of measurement system are explored:

- Impact of NPD process characteristics such as learning curve effects, rework likelihood and *overlapping strategy* (*Subsection 16.3.1*);
- Impact of rework review rigidity – *rework review strategy* (*SS 16.3.2*)[26];
- Impact of IEC arrival frequency (*SS 16.3.3*);

[26] The first two strategies are analyzed with only the NPD section of the model.

- Combined impact of IEC arrival frequency and magnitude (i.e., resource commitment) – *IEC batching policy (SS 16.3.4)*;
- Impact of functional resource constraints – *resource assignment Strategy (SS 16.3.5)*;
- Impact of change propagation due to interconnected product configuration (i.e., coupling among product components or systems) (*SS 16.3.6*).

6.2 Model Illustration by Numerical Examples

6.2.1 NPD Section

The NPD section is demonstrated by a simple application of three representational phases of an NPD process: i) concept design and development (***Concept***), ii) detailed product design (***Design***), and iii) production ramp up (***Production***). Each phase consists of three sequentially numbered and chronologically related activities. The information flow between every two activities is indicated by solid arrows as shown in *Figure 31*.

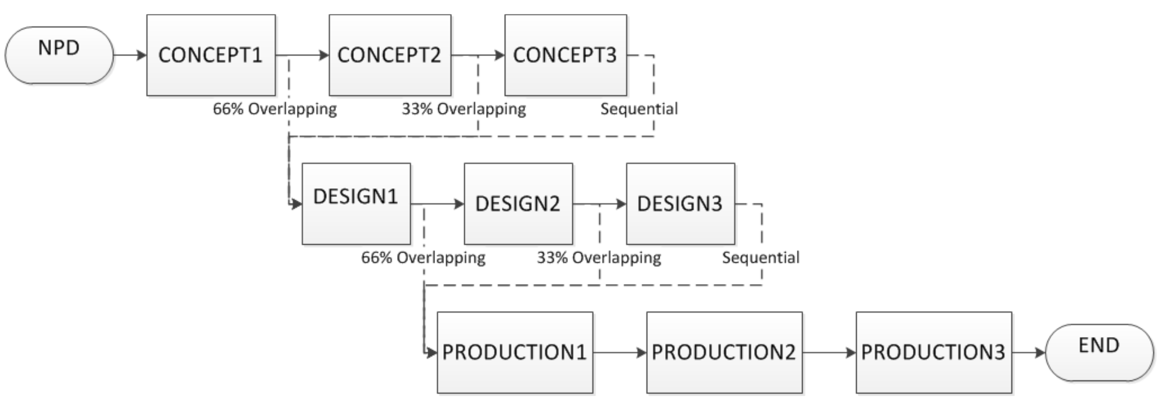


Figure 31: 3–Phase & 3–Activity NPD Framework

Through this 3–phase and 3–activity framework, various overlapping ratios of an NPD process: 0%, 33%, 66%, or mixed (e.g., 0% overlap between Concept and Design and 33% overlap between Design and Production), can be constructed by connecting intra–phase activities via different combinations of dashed arrows.

6.2.2 Overlapping Strategy: Sequential vs. Concurrent

An NPD process with 0% overlapping is also called a ***Sequential*** process, in which the downstream phase is allowed to start only after receiving the output information from the upstream phase in its finalized form. That is, different phases comprising an NPD process are connected in a completely linear fashion.

Besides its capability of representing a sequential process, this framework can also be assembled into ***Concurrent*** processes by allowing the parallelization of upstream and downstream activities as shown in *Figure 32.* For a 33% overlapped process, the first activity of downstream phase begins simultaneously with the last activity of upstream phase. Obviously, as compared to its counterpart in a sequential process, the solution uncertainty of the downstream activity increases due to the fact that it begins before the completion of all upstream activities using only preliminary output information, while the solution uncertainty of the upstream activity remains unchanged. That is to say, only the solution uncertainty of overlapped activities in succeeding phases (e.g., D1 and P1 under 33% overlapping strategy; D1, D2, P1, and P2 under 66% overlapping strategy) will be affected under the current model assumptions.

Also, this research presupposes that the integration of design effort from two or more overlapped and independently processed activities (i.e., the ones without information independency) that have already passed rework review could be obtained simply by adding them together in an exact way how the resultant effort accumulation is calculated for a set of sequentially processed activities. However, by doing this, the fact that overlapped downstream activity (or activities) starts in the absence of information output from the overlapped upstream activity has been ignored. A potential reduction in the cumulative effort when integrating overlapped activities should be considered in future work to better reflect the reality.

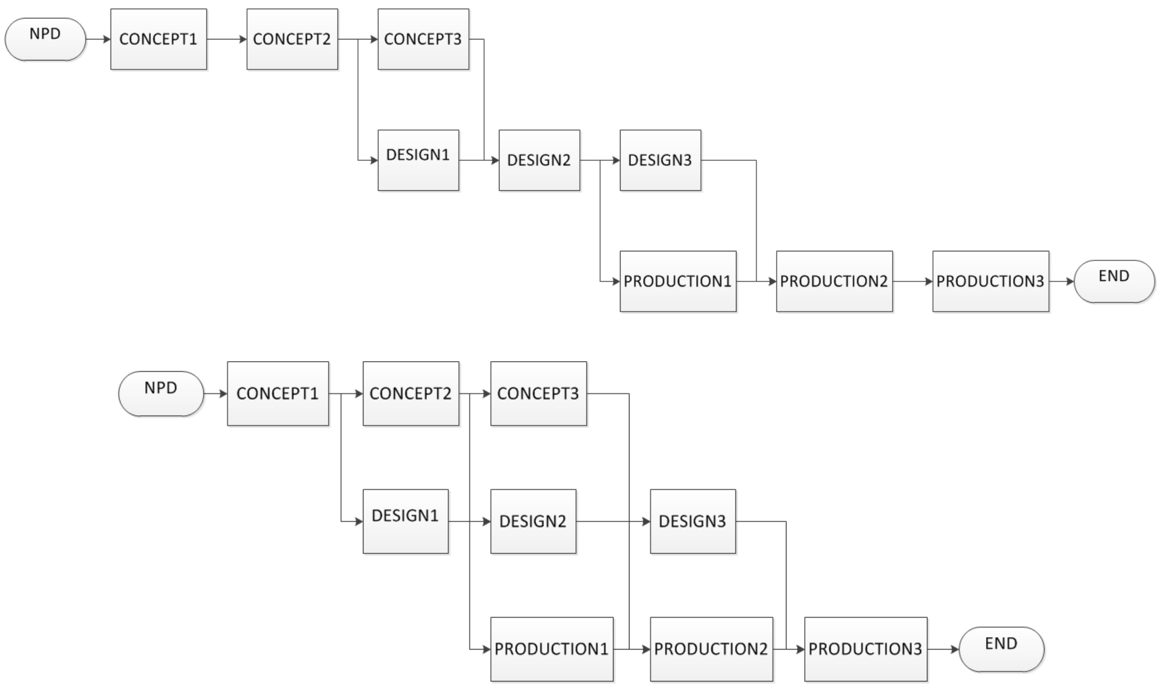


Figure 32: NPD Process with 33% & 66% Overlapping

Similarly, for a 66% overlapped NPD process represented by this 3–phase and 3–activity framework, the first activity of the following phase starts simultaneously with the second activity of the preceding phase. *APPENDIX B* shows how the overlapped upstream and downstream activities are actually modeled in Arena.

6.2.3 NPD Process Parameters

When considering the activity duration estimates, it is further assumed that the mutually independent and exponentially distributed duration has a mean of $\beta = 2$ days for activities in all three phases. Furthermore, the number of tasks that compose activities within one phase remains the same, but increases from phase to phase to represent the increasing content and complexity of design and development activities as the NPD project unfolds: $k = 4$ for activities in Concept phase; $k = 6$ for Design phase; and $k = 10$ for Production phase. Note that when the ***Learning Curve Effects*** are taken into account, random variables described by the Erlang distribution $ERLANG(\beta, k)$ only represent processing intervals of NPD basework. Rework duration is also subject to N_{ij}, the number of times that an activity is attempted, in the form of $LCE = max\left(\left(\frac{1}{2}\right)^{N_{ij}}, 0.1\right)$.

To match the three major phases of the illustrated NPD process, it is assumed that there exist three different functional areas: *marketing*, *engineering*, and *manufacturing*, that participate in the overall NPD process through integrated ***Departmental Interaction***. Based on the model assumption that each activity consumes a total number of 100 resources units to complete, departmental interaction is defined as follows: 60 units (i.e. individual servers) requested from major department and 20 units requested from each of the other two minor departments. To estimate the final project cost, the busy usage cost rates are set as *$25/hour* and idle cost as *$10/hour* for all resources. The impacts of ***Resource Constraints*** ranging from 70 – 200 units per department will be examined in *SS 16.3.5*.

Different rigidities of rework review, which are represented by various rework criteria ratios (i.e., relationships between rework criteria and the evolving functional design solution scope

$(S_m)_t$) will be explored more in depth through "what–if" analysis presented in *SS 16.3.2*. Details concerning numerical implementation are given in *APPENDIX C*.

6.2.4 NPD Rework: Iterations and EECs

Differentiation between NPD iterations (indicated by connectors in green) and EECs (indicated by connectors in red) is illustrated in *Figure 33*. This work flow chart presents all the possible intra– and inter–phase rework loops within and between Concept phase and Design phase.

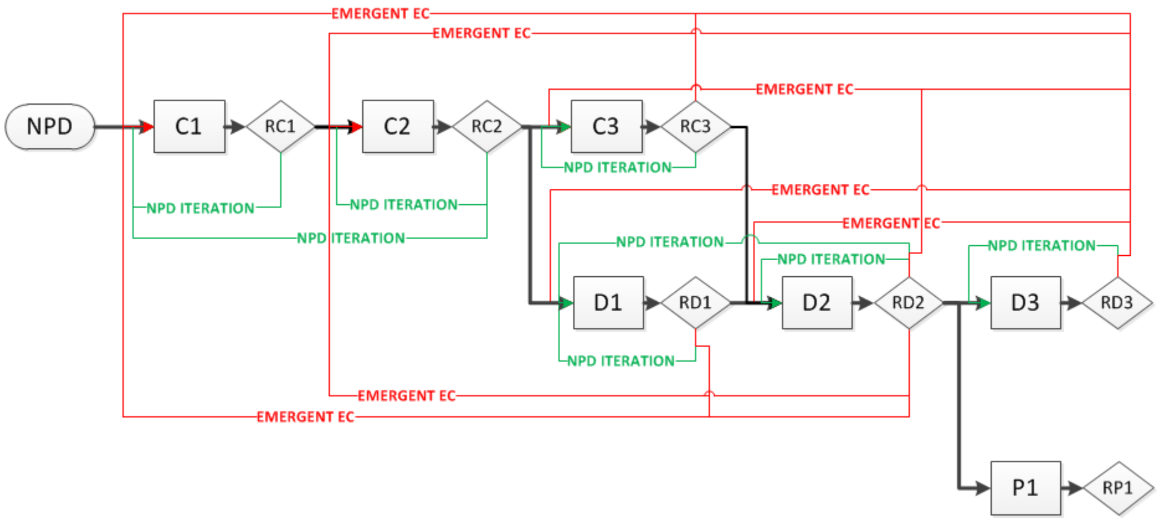


Figure 33: Model Section (Concept & Design phase) of NPD Iteration & EEC

Once the outcome of an activity has been released to activity (/activities) in its downstream phase (i.e., concept design information released to Design phase, or detailed design information released to Production phase), any rework to this activity is defined as an EEC. On the other hand, modifications to activities whose outcome has not been finalized and received by activities in downstream phase are called NPD iterations. The activity to which the NPD project is looped back is the starting point of the rework loop.

A complete list of all possible “starting points” of NPD iteration loops or EEC loops identified by the rework review following the completion of each activity is given in *APPENDIX D*. Recall that iteration and EEC probabilities are determined by the real–time value of solution uncertainty. It is further assumed that each possible starting point displayed in *APPENDIX D* has an equal chance of being selected.

The four curves shown in *Figure 34* illustrate how functional iteration probability and functional EEC probability vary with the cumulative committed functional effort under two rework likelihood levels characterized by different sets of rework probability constants.[27]

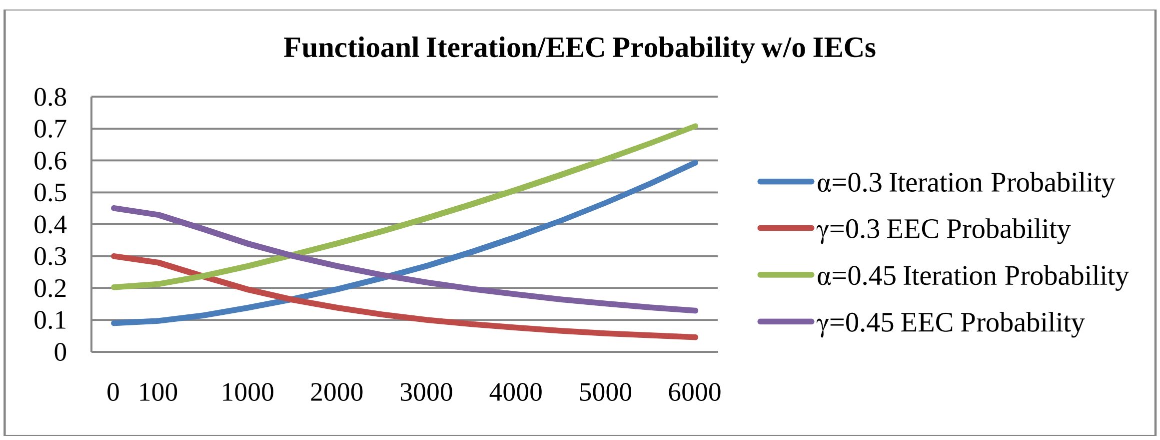


Figure 34: Functional Rework Probability vs. Cumulative Committed Functional Effort

The two convex increasing (increasing with increasing rates) curves over the cumulative committed functional effort reflect the functional iteration probabilities for activities with rework likelihood characterized by IPCs $\alpha = 0.3$ and $\alpha = 0.45$, while the other two convex decreasing

[27] Although *Figure 34* is illustrated in a continuous way, the model actually deals with rework probabilities only at discrete points. Also, these four lines reflect rework probability values of an NPD process without counting for the occurrence of IECs in which $(S_m)_t$ remains unchanged.

(decreasing with decreasing rates) curves characterized by EPCs $\gamma = 0.3$ and $\gamma = 0.45$ are on behalf of EEC probabilities. Note that *Figure 34* only shows functional rework probability. As a result of the cross–functional integration, the actual rework probability assigned to a particular activity during the rework review is the weighted average (by resource requirement) of the functional rework probabilities from all participating departments.

6.2.5 IEC Section

Figure **35** gives an overview of the IEC model section applying 33% overlapping strategy. It is assumed that an IEC will propagate to one of its downstream activities in the current or next phase with equal chances, and this propagation will continue in the same manner until the end of IEC propagation loop when no more change is identified.

For the purpose of demonstration, a full list of potential downstream change propagations of each IEC is provided on the right side of the IEC Propagation decision point. In the actual simulation model, verbal description is replaced by connectors between the IEC propagation decision point and the corresponding IEC process modules (i.e., the rectangular blocks in

Figure **35**).

Take the IEC to activity Concept1 as an example, change propagation will result in a maximum of six follow–up IECs (i.e., IECs to C2, C3/D1, D2, D3/P1, P2, and P3) and a minimum of two (i.e., IECs to C3/D1/D2 and D3/P1/P2/P3). For simplicity, it is also assumed that each IEC, no matter in which activity it is occurred, equally consumes 10 resource units from each of the three departments to get processed.

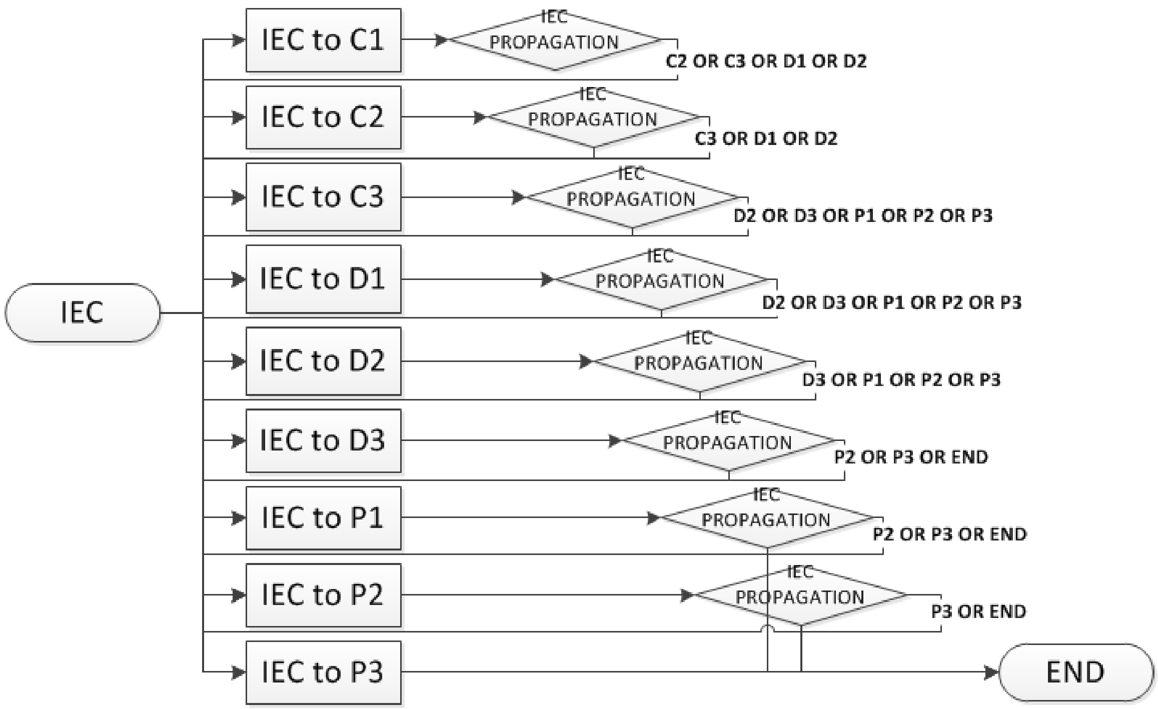


Figure 35: Overview of IEC Section for 33% Overlapping (Coupled PD Activities only)

Definition expression of key model variables/attributes and the information required to complete major Arena modules, such as *Process* Module and *Assign* Module are provided in *APPENDIX E*.

6.2.6 Summary of Model Inputs and Outputs

Table 11 summarizes a complete list of model input data. It is important to know that all the model parameter values are set in a way to facilitate relative comparison of project performance among various scenarios using "what–if" analysis instead of aiming to reproduce the real behavior patterns of an NPD project of any kind. To successfully implementation of the proposed simulation model for a specific use or situation, these inputs should be appropriately adapted depending on different circumstances.

There are altogether 14 model inputs that represent key NPD and ECM decision parameters, among which 7 (highlighted rows in gray) are chosen as *design factors* or *constraints* and their effects on the project *performance measures* will be tested at specific levels (highlighted text in bold), while others will be held constant when the design of experiment is conducted.[28]

Table 11: Model Inputs

Input Data	Value
List of phases and activities comprising process	$I = 3\ (Concept - Design - Production)$; $J_i = 3\ (e.g. Concept1 - Concept2 - Concept3)$
List of involving departments	$M = 3$ $(Marketing;\ Engineering;\ Manufacturing)$
Overlapping Strategy ***(OS)***	$\boldsymbol{Low}$: 0%; $\boldsymbol{Medium}$: 33%; $\boldsymbol{High}$: 66%
NPD Activity Duration *(days)*	$d_{1j} = ERLANG(2,4), D_{1j} = 8$; $d_{2j} = ERLANG(2,6), D_{2j} = 12$; $d_{3j} = ERLANG(2,10), D_{3j} = 20; j = 1,2,3$
Learning Curve Effects ***(LCE)***	$\boldsymbol{no\ LCE}; \boldsymbol{LCE} = max\left(\left(\frac{1}{2}\right)^{N_{ij}}, 0.1\right)$
NPD Activity Functional Resource Consumption	$r_{1j1} = 60, r_{1j2} = r_{1j3} = 20$; $r_{2j1} = 20, r_{2j2} = 60, r_{2j3} = 20$; $r_{3j1} = r_{3j2} = 20, r_{3j3} = 60; j = 1,2,3$
Functional Resources Constraints ***(FRC)***	$\boldsymbol{R_m} = 70, 80, \ldots, 190, 200;\ m = 1,2,3$
Cost of Resource	$Busy/Hour = \$25; Idle/Hour = \$10$
Rework Likelihood ***(RL)***	$\boldsymbol{Low}: \alpha = \gamma = 0.3;\ \boldsymbol{High}: \alpha = \gamma = 0.45$
Rework Criteria ***(RC)***	***Stepped Linear; Linear; Convex–Up; Concave–Up***
IEC Arrival Frequency (Inter–arrival Times) *(days)*	$\boldsymbol{Low}: Random\ (Expo)20$; $\boldsymbol{Medium}: Random\ (Expo)10$; $\boldsymbol{High}: Random\ (Expo)5$
IEC Duration Estimates *(days)*	$w_{lg} = TRIA(1.6, 2, 3.2), g = 1,2,3$; $w_{lg} = TRIA(2.4, 3, 4.8), g = 4,5,6$; $w_{lg} = TRIA(4,5,8), g = 7,8,9; l = 1,2,\ldots,L_t$
IEC Functional Resource Consumption	$\boldsymbol{s_{lgm}} = 10\ and\ 20, l = 1,2,\ldots,L_t$; $g = 1,2,3;\ m = 1,2,3$

[28] These held-constant factors, such as number of phases and activities comprising the process, number of involving departments, duration estimates of NPD activities and IECs, etc., are peculiar to specific development project as . For purposes of the present experiment these factors are not of interest.

At the end of each simulation run, Arena automatically generates a variety of both default and user specified model output statistics, which include time, cost, Work in Process (WIP), count, etc. Information is displayed under different category sections (e.g., Entity, Process, Queue, Resource, and User Specified). Some of the key model *responses* are listed in the table below.

Note that summary data shown in an Arena report are statistics (e.g., sample mean, sample standard deviation, 95% confidence interval half width, minimum output value, maximum output value, etc.) over the replications. To compare and evaluate all different alternative system configurations, the Arena built–in function Process Analyzer (PAN) is adopted to more effectively and efficiently collect results of running all scenarios. Outputs of each model replication run are written to an Excel worksheet and displayed in a scatter plot using the ReadWrite function.

Table 12: Model Outputs

Output Data	Definition
NPD Project Lead Time	The total time of an NPD entity accumulated in process activities and delays (time elapsed between start of Concept phase and end of Production phase).
Project Cost	The total of busy costs (i.e., costs while seize) for all staffing and resources for both NPD and IEC entities.
Total Cost	The total expenditure on both busy and idle (i.e., costs while scheduled, but not busy) resources for NPD and IEC entities.
Cumulative Functional Effort	The accumulated departmental workload (in units of person–days) accounted for both NPD and IEC entities.
Cumulative Total Effort	The accumulated total effort accounted for both NPD and IEC entities (i.e., the sum of all the cumulative functional efforts).
Quality	The ratio of the final design solution scope over the original design solution scope.

6.3 Experimental Design, Simulation Results and Discussions

6.3.1 Impact of Process Characterizations

First of all, only the NPD section of the model framework is examined to investigate quantitatively[29] how the three major factors that characterize a development process: i) ***Overlapping Strategies (OS)***, ii) ***Rework Likelihood (RL)*** which is represented by either iteration probability constant α or EEC probability constant γ, and iii) ***Learning Curve Effects (LCE)***, and also the interactions between them actually impact the occurrence and magnitude of rework, and thus affect the two response variables: NPD *lead time* and final *project cost*. Note that project cost is referring to the busy cost of resource usage. It should be differentiated from the *total cost* (sum of busy and idle cost), which will be measured and compared in later analyses.

Specifically, three factor levels of *OS*: **(a)** Low 0%, **(b)** Medium 33%, and **(c)** High 66%; two factor levels of *RL*: **(1)** Low $\alpha = \gamma = 0.3$ and **(2)** High $\alpha = \gamma = 0.45$; and two factor levels of *LCE*: **(A)** $no\ LCE$ and **(B)** $LCE = max\left(\left(\frac{1}{2}\right)^{N_{ij}}, 0.1\right)$ are selected in the experimental design to measure how these process variables result in different values for the model response. Functional resource availability is fixed at $R_k = 100, k = 1, 2, 3$. Also, the rework criteria of review decisions after the completion of each activity follow a "Stepped Linear" strategy, which will be discussed in detail in the following subsection.

Since we haven't taken IECs into account yet, there is no change of the design solution scope as the project unfolds (i.e., $(S_m)_t = EN_m$). Therefore, the final project quality remains unchanged. Running results of the ideal but unrealistic case, an NPD project that proceeds

[29] As opposed to the qualitative analysis through the construction of causal loop diagrams in Chapter 3.

exactly according to the pre–determined development schedule without any rework, are used as the baseline **(BL1)** to compare the impact of rework on the responses of interest under different scenarios.

6.3.1.1 Total Efforts – Mean Values

200 *replicates* are generated under each combination of *LCE*, *RL,* and *OS*, and thus result in altogether ($200 \times 3 \times 2 \times 2 =$) 2400 simulation runs each using separate input random numbers. Performance data generated by the model are then exported to a Microsoft Excel worksheet, in which individual project performance measures are recorded and various experiments are generated.

Mean values of the experiment outcomes are displayed in *Table 13*. Columns **(i)** and **(ii)** record in an absolute sense the mean values of the observed lead time and project cost from 200 replications of each scenario, while columns **(I)** and **(II)** show the percentage change of **(i)** and **(ii)** relative to the baseline case results (**BL1**), respectively.

Besides simply obtaining mean values of the responses for each performance measure and its percentage change from baseline scenario, a three–factor *Analysis of Variance (ANOVA)* for this 2×2×3 factorial design is further conducted to test hypotheses about the significance of factors' main effects, and to determine whether factors interact using the statistical software package Minitab 16.0.1 developed by Minitab, Inc. (State College, Pennsylvania). ANOVA results for NPD lead time and project cost are summarized in *Table 14*.

Table 13: Project Performance under the Impact of *OS*, *RL* and *LCE*

LCE	***RL* (α, γ)**	***OS***	**(i) Lead Time (Days)**	**(I) Time %Change c/w BL1**	**(ii) Project Cost ($\$ \times 1000$)**	**(II) PC %Change c/w BL1**
(BL1) Baseline	No Rework	**(a)** 0%	119		7,168	
		(b) 33%	101		7,168	
		(c) 66%	81		7,169	
(A) $No\ LCE$	**(1) *Low*** $\alpha = \gamma =$ 0.3	**(a)** 0%	158	32.0%	**10,781**	48.2%
		(b) 33%	160	58.9%	11,778	61.9%
		(c) 66%	**131**	62.6%	12,107	66.6%
	(2) *High* $\alpha = \gamma =$ 0.45	**(a)** 0%	176	47.2%	**11,948**	64.2%
		(b) 33%	192	90.4%	14,542	99.8%
		(c) 66%	**162**	100.1%	14,927	105.4%
(B) $LCE = max\left(\left(\frac{1}{2}\right)^{N_{ij}-1}, 0.1\right)$	**(1) *Low*** $\alpha = \gamma =$ 0.3	**(a)** 0%	141	17.6%	9,542	33.1%
		(b) 33%	129	28.1%	9,436	31.6%
		(c) 66%	**106**	31.0%	**9,185**	28.1%
	(2) *High* $\alpha = \gamma =$ 0.45	**(a)** 0%	152	27.2%	**10,370**	44.7%
		(b) 33%	158	56.6%	12,044	68.0%
		(c) 66%	**121**	49.2%	11,037	54.0%

It is important to note that we are not making managerial suggestions merely based on the final output performance measures (i.e., columns **(i)** & **(ii)**) obtained for each scenario. Rather our attention is also focused on the comparison of these numbers to their corresponding baseline results (i.e., columns **(I)** & **(II)**), which helps to provide us intuitive understanding of the impacts of reworks on project performance under different process features and parameter settings. Through the interpretation of results presented in *Table 13* and *Table 14*, several concluding observations can be issued:

1. When rework is not involved, the project performance stays consistent: the higher the activity overlapping ratio, the less the lead time. It can be obtained by summing up the durations of activities along the critical path. At the same time, since total person–days effort required for completing the project remains unchanged no matter which *OS* is

applied, final project cost for all levels of *OS* (i.e., **(a)**, **(b)**, and **(c)**) in the baseline case should be very much similar, which is confirmed by the running results. This can be considered as a simple model verification check[30].

2. *P*–values for the test statistics in both ANOVAs indicate that all three factors, *LCE*, *RL*, and *OS*, affect both lead time and project cost significantly. In addition to these main effects, the interaction between *RL* and *OS* is significant to NPD lead time. Also, interactions *LCE* – *RL* and *RL* – *OS* have *P*–values around 0.05, indicating some influence between them. Furthermore, *LCE* – *OS* and *RL* – *OS* interactions are identified to be significant to project cost.

Table 14: ANOVA for NPD Lead Time and Project Cost

Factors and Levels of the Experiment					
Factor	Type	Levels	Values		
LCE	Fixed	2	$No\ LCE, LCE = max\left(\left(\frac{1}{2}\right)^{N_{ij}-1}, 0.1\right)$		
RL	Fixed	2	$\alpha = \gamma = 0.3, \alpha = \gamma = 0.45$		
OS	Fixed	3	0%, 33%, 66%		
Analysis of Variance for NPD Lead Time					
Source of Variation	Sum of Squares	Degrees of Freedom	Mean Square	F_0	P–Value
LCE	132105	1	132105	248.63	0.000[31]
RL	77826	1	77826	146.48	0.000
OS	116772	2	58386	109.89	0.000
LCE×*RL*	2011	1	2011	3.78	0.052
LCE×*OS*	3139	2	1569	2.95	0.053
RL×*OS*	7948	2	3974	7.48	0.001
LCE×*RL*×*OS*	1654	2	827	1.56	0.212
Error	312418	2388	531		
Total	653872	2399			

[30] Model is continuously verified by the reading through and examining the outputs for reasonableness and justification under a variety of scenarios and settings of parameters.

[31] If the P-value is <0.05 (those of which are highlighted in red), we can conclude that the single factor or the interaction between two factors is a significant effect at 95% level of significance.

Analysis of Variance for NPD Project Cost					
Source of Variation	Sum of Squares	Degrees of Freedom	Mean Square	F_0	P–Value
LCE	9.05414E+14	1	9.05414E+14	277.00	0.000
RL	6.10608E+14	1	6.10608E+14	186.81	0.000
OS	2.06364E+14	2	1.03182E+14	31.57	0.000
LCE×*RL*	5.38479E+12	1	5.38479E+12	1.65	0.200
LCE×*OS*	9.60643E+13	2	4.80322E+13	14.69	0.000
RL×*OS*	8.39527E+13	2	4.19763E+13	12.84	0.000
LCE×*RL*×*OS*	8.76979E+12	2	4.38489E+12	1.34	0.262
Error	1.92197E+15	2388	3.26866E+12		
Total	3.83853E+15	2399			

3. ***Effects of LCE:*** by comparing the mean values of lead time and project cost of scenarios **(A)** with scenarios **(B)** under different combinations of *RL* and *OS* levels, it can be concluded that the evaluation of learning curve effects unambiguously results in a remarkable decrease in both NPD lead time and cost.

4. ***Effects of RL:*** by comparing **(i)** and **(ii)** of scenarios **(1)** with scenarios **(2)** under different combinations of *LCE* and *OS* levels, it can be concluded that a higher likelihood of rework in NPD activity undoubtedly causes an increase in both lead time and cost.

5. ***Effects of OS w/o LCE:*** by comparing lead time and project cost of scenarios **(A)** in a relative sense (i.e., columns **(I)** and **(II)**), we find that an increasing overlapping ratio aggravates the impact of NPD rework on both responses. That is, when NPD rework is included in the model but no *LCE* is considered, the greater the overlapping ratio, the higher the percentages of increase in both lead time and project cost as compared to baseline case. In addition, we notice the time–cost tradeoffs between a sequential process and a 66% overlapped process from columns **(i)** and **(ii)**. This observation agrees to the general acknowledgement that overlapping may save time but is more costly.

6. ***Effects of OS w/ LCE:*** Situation is not that predictable when *LCE* is taken into account and formulated as $LCE = max\left(\left(\frac{1}{2}\right)^{N_{ij}-1}, 0.1\right)$ in the model. Significant increase of both time and cost due to rework is alleviated by the evaluation of *LCE*. Under low *RL* circumstances ($\alpha = \gamma = 0.3$), a highly overlapped process excels in both response variables in an absolute sense. However, there is not clear trend shown in the comparative values (i.e., columns **(I)** and **(II))**. Particularly, at high level of *RL* ($\alpha = \gamma = 0.45$), we observe that a 33% overlapped process leads to both absolute (compared with the results of 0% and 66% in scenario **(B)–(2)**) and relative (compared with the 33% baseline results **(BL1)–(b)**) maximum values for lead time and project cost.
7. By comparing columns **(I)** and **(II)**, we observe a project behavioral pattern that the percentage increase of project cost is always higher than that of lead time at the occurrence of rework. That is to say, compared with lead time, project cost is more sensitive to rework. And the difference between the two percentages of increase is largest when a sequential NPD process is adopted. The only exception is scenario **(B)–(1)–(c)** with the percentage increase of project cost 0.9% lower than that of lead time.

The above numerical results should only be used to gain qualitative insights. By no means can we conclude that by pursuing a higher overlapping ratio we will always end up with shorter development time and lower cost because of the following reasons:

a) These results are based on a particular set of model inputs as shown in *Table 11*.
b) The presented model is not feasible to examine any arbitrary overlapping strategy due to limitations of the model structure. Only three levels (i.e., 0%, 33% and 66%) can be constructed given this 3 – phase and 3 – activity NPD framework.

c) The model assumption that "downstream action can start with information in a preliminary form before all activities in upstream phase are completed" may not always be true in reality. Extremely high concurrency of activities can be very risky or even non–applicable for those NPD processes with strong informational dependencies among activities.

6.3.1.2 Functional Efforts – Mean Values

After investigating project cost performance that reflects the overall effort devoted to the NPD project, how the amount of functional effort contributed by each participating department is affected by different *LCE*, *RL*, and *OS* levels is further examined. Running results are recorded in *Table 15*. Following the same presentation format as previous, columns **(v)** to **(viii)** display the committed functional effort from Marketing, Engineering, and Manufacturing Departments and the overall total effort, respectively, measured in person–days; and columns **(V)** to **(VIII)**, on the other hand, exhibit the percentage change of these numbers versus baseline case behavior of the model.

Table 15: Functional Efforts under the Impacts of *LCE, OS,* and *RL*

LCE	***RL*** **(α, γ)**	***OS***	**(v) Mkt Effort** *(person–days)*	**(V) MktE % Change c/w BL1**	**(vi) Eng Effort** *(person–days)*	**(VI) EngE %Change c/w BL1**	**(vii) Mfg Effort** *(person–days)*	**(VII) MfgE %Change c/w BL1**	**(viii) Tot Effort** *(person–days)*	**(VIII) TE %Change c/w BL1**
(BL1) Baseline	No Rework	**(a)** 0%	3,298		3,851		4,799		11,948	
		(b) 33%	3,298		3,851		4,799		11,948	
		(c) 66%	3,296		3,861		4,791		11,948	
(A) $No\ LCE$	**(1)** ***Low*** $\alpha = \gamma = 0.3$	**(a)** 0%	4,749	44.0%	5,348	38.9%	7,872	64.0%	17,969	50.4%
		(b) 33%	6,073	84.1%	6,743	75.1%	6,676	39.1%	19,492	63.1%
		(c) 66%	6,198	88.0%	6,843	77.2%	7,112	48.4%	20,153	68.7%
	(2) ***High*** $\alpha = \gamma = 0.45$	**(a)** 0%	5,792	75.6%	5,845	51.8%	8,277	72.5%	19,914	66.7%
		(b) 33%	8,407	154.9%	8,350	116.8%	7,331	52.8%	24,088	101.6%
		(c) 66%	8,286	151.4%	8,617	123.2%	8,009	67.2%	24,912	108.5%
(B) $LCE = max\left(\left(\frac{1}{2}\right)^{N_{ij}-1}, 0.1\right)$	**(1)** ***Low*** $\alpha = \gamma = 0.3$	**(a)** 0%	4,067	23.3%	4,662	21.1%	7,175	49.5%	15,904	33.1%
		(b) 33%	4,543	37.8%	5,185	34.6%	5,886	22.7%	15,614	30.7%
		(c) 66%	4,629	40.4%	4,899	26.9%	6,067	26.6%	15,595	30.5%
	(2) ***High*** $\alpha = \gamma = 0.45$	**(a)** 0%	4,586	39.1%	5,127	33.1%	7,571	57.8%	17,284	44.7%
		(b) 33%	6,364	93.0%	6,766	75.7%	6,714	39.9%	19,844	66.1%
		(c) 66%	5,954	80.6%	6,077	57.4%	6,701	39.9%	18,732	56.8%

At the same time, *Figure 36* and *Figure 37* represent these two sets of data by simple line charts. Three major conclusions can be drawn by breaking down the overall committed effort into functional effort contributed by each department:

1. From *Figure 36*, we observe that differences between the committed effort from the major department (i.e., Mfg Effort) of downstream phase (i.e., Production phase), and the efforts devoted by the other two departments (i.e., Mkt Effort & Eng Effort) drop dramatically from a sequential process (i.e., **(a)**) to concurrent processes (i.e., **(b)** and **(c)**) regardless of *LCE* or *RL* levels.

2. Moreover, from a relative perspective, the percentage increase of Mfg Effort versus baseline is higher than those of Mkt and Eng Efforts in all sequential processes but **(A)–(2)–(a)**, in which Mfg Effort %Change = *72.5%* and is slightly lower than Mkt Effort %Change = *75.6%*. However, in concurrent processes, an inverse relationship but of a much greater magnitude (especially at high *RL* level) is observed. That is to say, by starting downstream activities early with only preliminary information, concurrent engineering tends to alleviate the impacts of rework on activities in Production phase while intensifying those on activities in the two upstream phases. Although the concept of *cross–functional integration* has already been applied to the sequential process that allows engineers from Mfg Dept to be engaged early in both Concept and Design phases, which differentiates it from a traditional waterfall process, the impact of rework mostly occur in Mfg Dept. A concurrent process tends to shift rework risks and even out committed efforts among various functional areas owing to another critical characterization of concurrent engineering: *parallelization of activities*.

3. Mkt Effort undergoes the highest percentage of increase when *RL* changes from low to high, regardless of *LCE* or *OS* levels. Then is the Eng Effort. Mft Effort has the least amount of fluctuation across different scenarios.

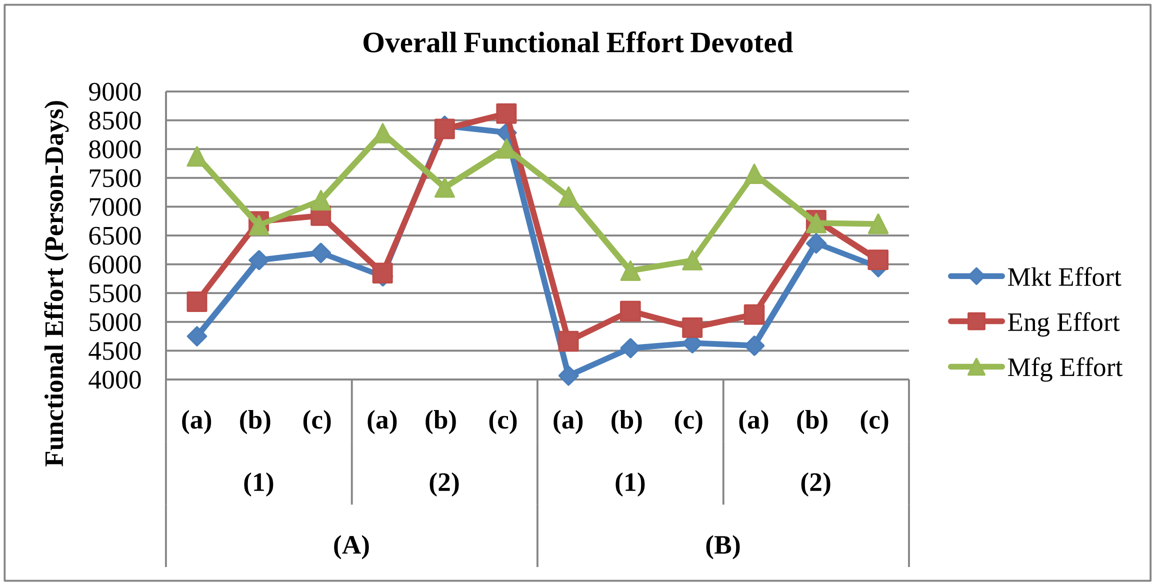


Figure 36: Overall Functional Effort Devoted

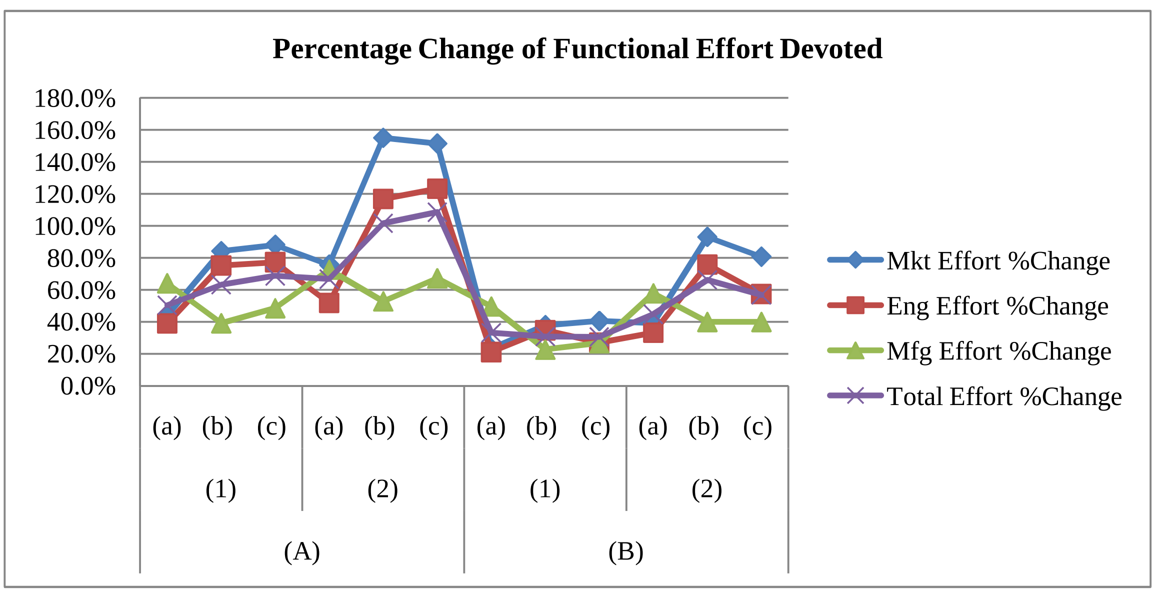


Figure 37: Percentage Change of Functional Effort Devoted

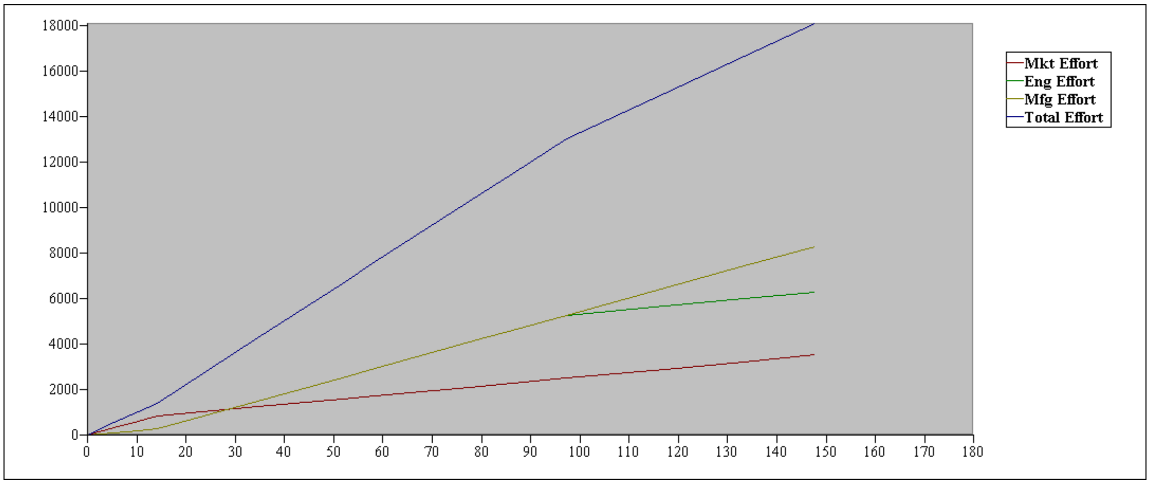


Figure 38: Cumulative Functional Effort and Total Effort w/o IECs (0%)

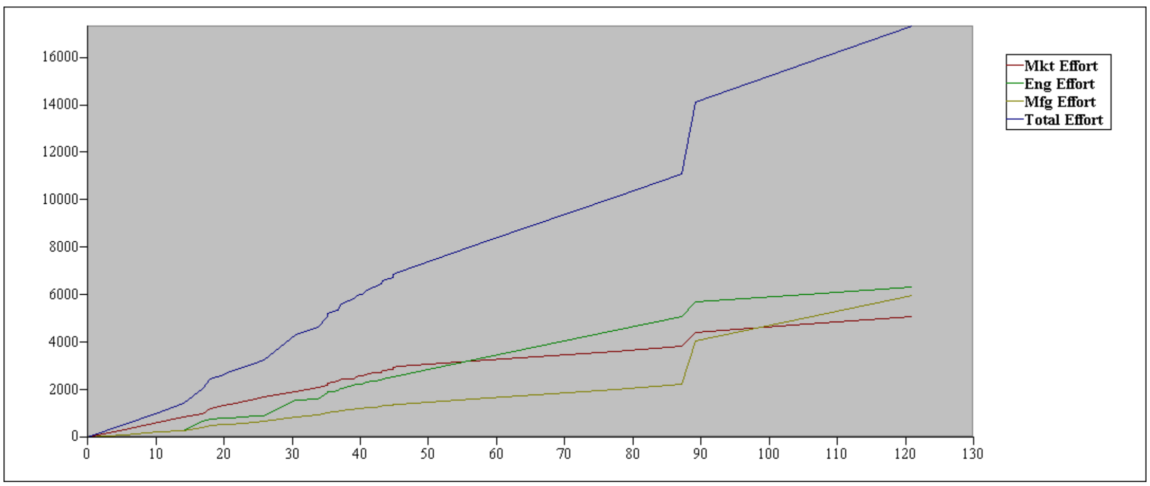


Figure 39: Cumulative Functional Effort and Total Effort w/o IECs (33%)

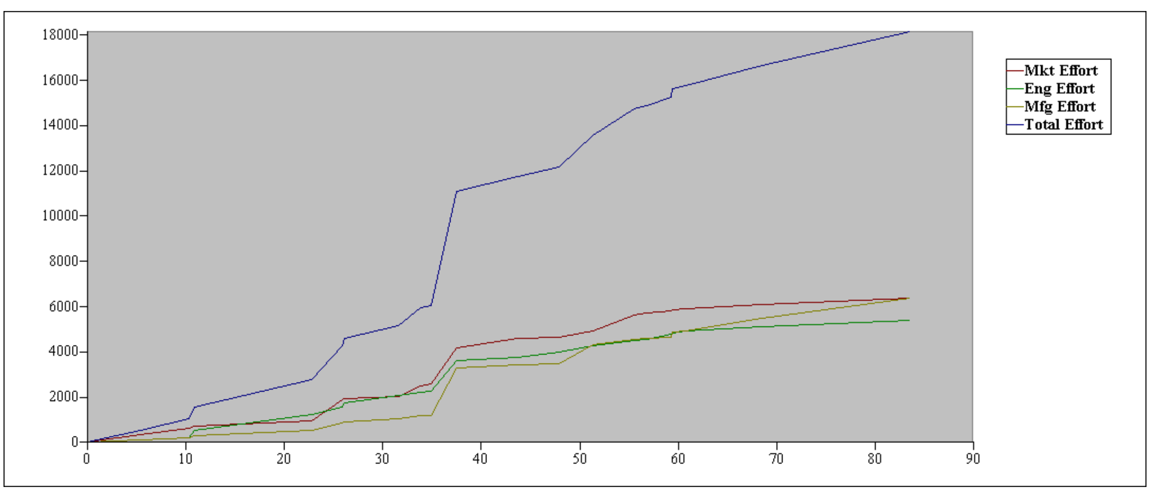


Figure 40: Cumulative Functional Effort and Total Effort w/o IECs (66%)

Figure 38 through *Figure 40* are observational plots of the cumulative functional and total effort over time for three levels of *OS*. Note that angled lines are drawn between “stepped” observed values.

6.3.1.3 Total Efforts – Scatter Plots

To better visualize the correlations between lead time and effort, scatter plots of 200 model replicates’ lead time and total effort outcomes under different levels of *OS* and *RL* are demonstrated in *Figure 41*. Red lines in the plots indicate the lead time and total effort required for **BL1** baseline cases (an “ideally executed” project without accounting for rework).

We can clearly observe that a majority of replications exceed the lead time and effort of **BL1** by a considerable amount because of rework. Furthermore, as overlapping ratio and rework probability constants (α for IPC and γ for EPC) increase, there is also a notable increase in the number of replicates that are off the trend line. This phenomenon reveals that a high overlap ratio of upstream and downstream activities, combined with a high likelihood of unanticipated activity rework that requires additional resources will result in a strong tendency for NPD projects to behave in an unstable and unpredictable manner and lead to unforeseen departures from the predetermined baseline plan.

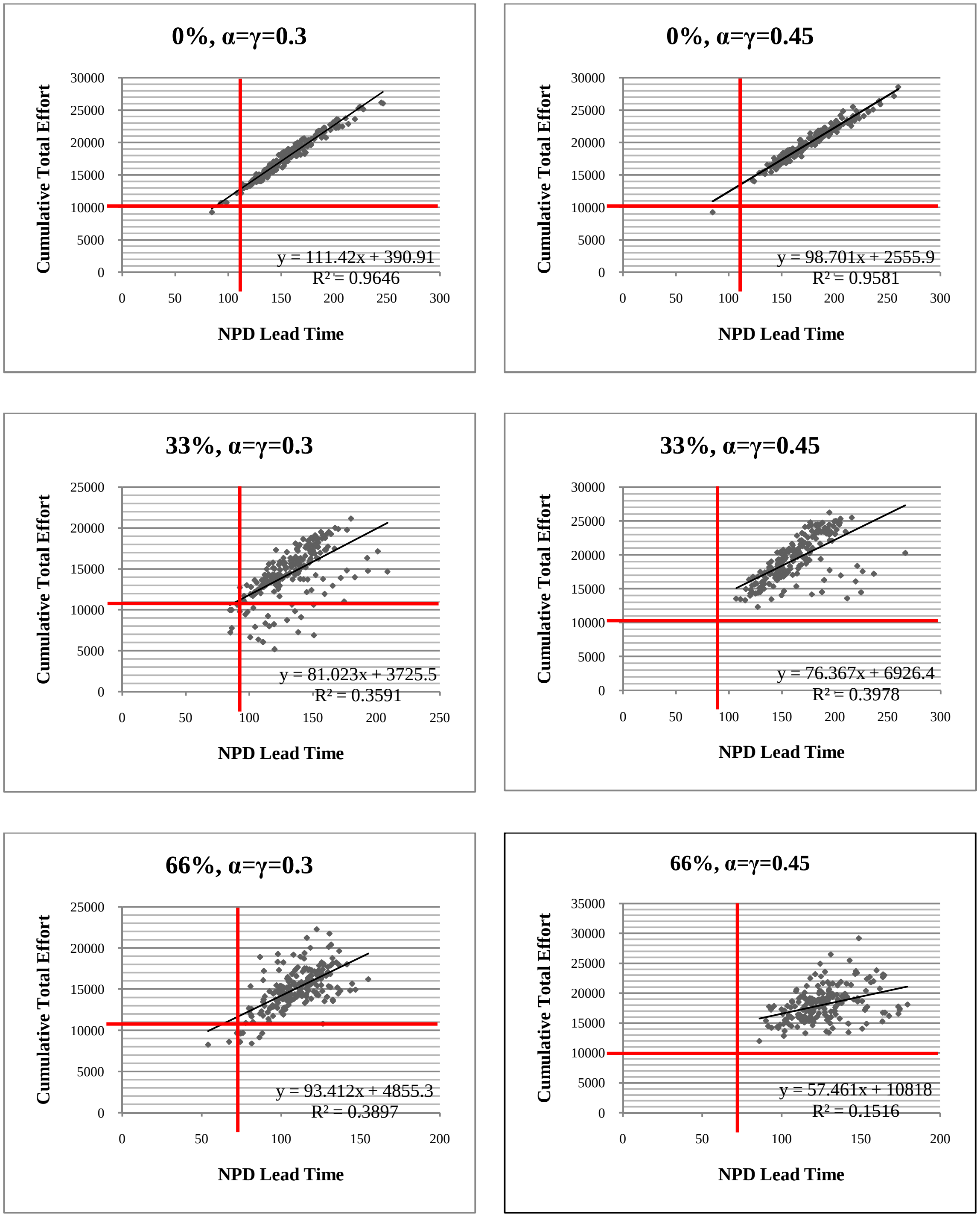


Figure 41: Scatter Plots of the *RL* Impact on Different *OS*

6.3.2 Impact of Rework Review Strategy

In this subsection, different types of ***Rework Review Strategies (RRS)*** which are applied at decision points of rework review after the completion of each activity are investigated for their effects on lead time and project cost. *RRSs* are characterized in this research by ***Rework Criteria (RC)***: model variables in the form of a certain percentage (i.e., *RC* ratio) of the design solution scope $(S_m)_t$ (or EN_m when IECs are not counted). *RC* represents the minimal expectation for an activity in terms of the cumulative functional effort devoted, above which the activity outcome will be accepted by engineers and project managers without conducting the third step "rework evaluation" as shown in *Figure 30*. When the cumulative functional effort up to date fails to meet (i.e., is less than) the *RC*, the NPD project will need to proceed with a rework evaluation. It may either continue to perform next activity/activities (depending on the *OS* used), or start an intra–phase iteration rework loop or an inter–phase EEC rework loop according to the weighted rework probability calculated based on the current value of functional solution uncertainty.

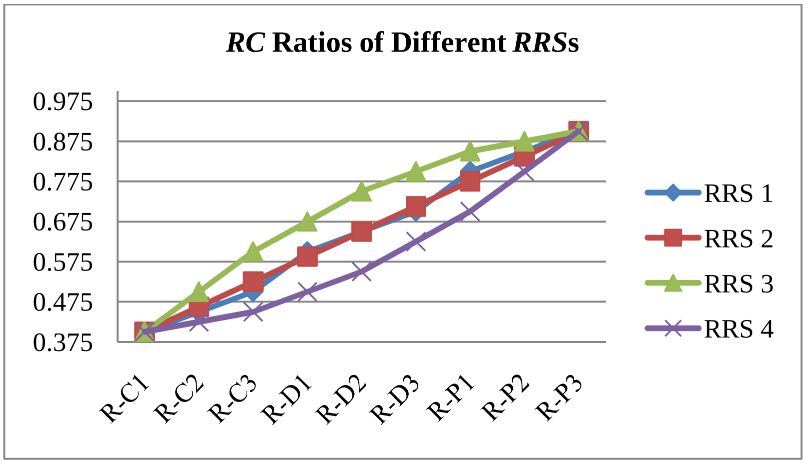


Figure 42: Rework Criteria Ratios of Different Rework Review Strategies

As shown in *Figure 42*, while having *RC* ratios fixed for both reviews following activity Concept 1 (R–C1) and Production 3 (R–P3), four *RRS*s with different increasing patterns of *RC* ratios along the course of an NPD process are examined: **(BL2: Stepped Linear)** increasing linearly within each phase in 5% increments and across phases in a 10% increment; **(C: Linear)** increasing linearly in 6.25% increments; **(D: Convex–Up)** increasing at a decreasing rate; and **(E: Concave–Up)** increasing at an increasing rate.

Note that the first type “stepped linear” is served as the baseline case to which the model behavior under different *RRSs* is compared. It is used as the default *RRS* in later analysis unless otherwise specified.

Running results for all combinations of *RRS*, *LCE*, *RL*, and *OS* levels are displayed in *Table 16*, from which the following three major conclusions can be drawn:

1. ***Effects of RRS:*** there is no obvious distinction in lead time or project cost observed between **(BL2: Stepped Linear)** and **(C: Linear)** *RRS*s. Adoption of the **(D: Convex–Up)** *RRS*, which is a more restrictive policy compared to others, leads to a longer NPD lead time and higher project cost. Adversely, adoption of the **(E: Concave–Up)** *RRS* (a less restrictive policy) leads to a shorter NPD lead time and lower project cost.
2. ***Effects of LCE:*** by comparing results of **(I)** and **(II)** under *No LCE* and $LCE = max\left(\left(\frac{1}{2}\right)^{N_{ij}-1}, 0.1\right)$, especially for scenarios **(D)** and **(E)**, we observe constant higher absolute values in *No LCE* cases, from which we can conclude that the inclusion of *LCE* reduces the impacts of *RRS*.

3. One thing worth noting is that **(i)** and **(ii)** values of scenarios **(D)–(b)** in *No LCE* cases and scenario **(D)–(2)–(b)** in $LCE = max\left(\left(\frac{1}{2}\right)^{N_{ij}-1}, 0.1\right)$ case (see the numbers highlighted in bold) are much higher than results of the corresponding baseline cases while **(I)** and **(II)** values are much higher than results of scenarios under same *RRS* and *RL* but different *OS*.

Table 16: Project Performance under the Impact of *RRS*

RRS	***RL*** **(α, γ)**	***OS***	**(i) Lead Time (Days)**	**(I) Time %Change c/w BL2**	**(ii) Project Cost ($\$ \times 1000$)**	**(II) PC %Change c/w BL2**
No LCE						
(BL2A) *RRS1*: Stepped Linear	**(1) *Low*** $\alpha = \gamma = 0.3$	**(a)** 0%	158		10,781	
		(b) 33%	160		11,778	
		(c) 66%	131		12,107	
	(2) *High* $\alpha = \gamma = 0.45$	**(a)** 0%	176		11,948	
		(b) 33%	192		14,542	
		(c) 66%	162		14,927	
(C) *RRS2*: Linear	**(1) *Low*** $\alpha = \gamma = 0.3$	**(a)** 0%	158	–0.10%	10,774	–0.06%
		(b) 33%	159	–0.39%	11,735	–0.37%
		(c) 66%	130	–1.32%	11,975	–1.09%
	(2) *High* $\alpha = \gamma = 0.45$	**(a)** 0%	177	0.76%	12,033	0.71%
		(b) 33%	192	0.22%	14,573	0.21%
		(c) 66%	160	–0.84%	14,880	–0.31%
(D) *RRS3*: Convex–Up	**(1) *Low*** $\alpha = \gamma = 0.3$	**(a)** 0%	165	4.66%	11,317	4.97%
		(b) 33%	**189**	**18.10%**	**14,270**	**21.16%**
		(c) 66%	137	4.32%	12,651	4.49%
	(2) *High* $\alpha = \gamma = 0.45$	**(a)** 0%	190	7.81%	12,912	8.07%
		(b) 33%	**219**	**13.94%**	**17,228**	**18.47%**
		(c) 66%	170	4.86%	15,776	5.69%
(E) *RRS4*: Concave–Up	**(1) *Low*** $\alpha = \gamma = 0.3$	**(a)** 0%	153	–2.88%	10,412	–3.42%
		(b) 33%	153	–4.33%	11,276	–4.26%
		(c) 66%	125	–4.74%	11,502	–5.00%
	(2) *High* $\alpha = \gamma = 0.45$	**(a)** 0%	166	–5.61%	11,231	–6.00%
		(b) 33%	180	–6.19%	13,608	–6.43%
		(c) 66%	150	–7.45%	13,825	–7.38%

Table 16: Project Performance under the Impact of *RRS* (cont'd)

RRS	***RL*** **(α, γ)**	***OS***	**(i) Lead Time (Days)**	**(I) Time %Change c/w BL2**	**(ii) Project Cost ($\$ \times 1000$)**	**(II) PC %Change c/w BL2**
$LCE = max\left(\left(\frac{1}{2}\right)^{N_{ij}-1}, 0.1\right)$						
(BL2B) ***RRS1*****: Stepped Linear**	**(1)** ***Low*** $\alpha = \gamma = 0.3$	**(a)** 0%	141		9,542	
		(b) 33%	129		9,436	
		(c) 66%	106		9,185	
	(2) ***High*** $\alpha = \gamma = 0.45$	**(a)** 0%	152		10,370	
		(b) 33%	158		12,044	
		(c) 66%	121		11,037	
(C) ***RRS2*****: Linear**	**(1)** ***Low*** $\alpha = \gamma = 0.3$	**(a)** 0%	141	0.00%	9,540	–0.03%
		(b) 33%	128	–0.62%	9,382	–0.57%
		(c) 66%	106	–0.28%	9,167	–0.20%
	(2) ***High*** $\alpha = \gamma = 0.45$	**(a)** 0%	152	–0.13%	10,365	–0.05%
		(b) 33%	158	–0.06%	12,068	0.20%
		(c) 66%	120	–0.33%	11,011	–0.23%
(D) ***RRS3*****: Convex–Up**	**(1)** ***Low*** $\alpha = \gamma = 0.3$	**(a)** 0%	144	2.21%	9,763	2.32%
		(b) 33%	130	0.93%	9,544	1.14%
		(c) 66%	107	0.95%	9,314	1.40%
	(2) ***High*** $\alpha = \gamma = 0.45$	**(a)** 0%	157	3.36%	10,757	3.73%
		(b) 33%	**169**	**6.98%**	**13,215**	**9.73%**
		(c) 66%	124	2.90%	11,485	4.06%
(E) ***RRS4*****: Concave–Up**	**(1)** ***Low*** $\alpha = \gamma = 0.3$	**(a)** 0%	138	–1.78%	9,352	–1.99%
		(b) 33%	126	–2.25%	9,219	–2.29%
		(c) 66%	102	–3.78%	8,873	–3.40%
	(2) ***High*** $\alpha = \gamma = 0.45$	**(a)** 0%	145	–4.47%	9,860	–4.92%
		(b) 33%	150	–5.07%	11,414	–5.23%
		(c) 66%	116	–4.15%	10,505	–4.81%

6.3.3 Impact of IEC Arrival Frequency

After analyzing the NPD section of the proposed model framework only, a separate IEC part is added to evaluate how handling of IECs that arise from outside sources will affect the design solution scope and solution uncertainty, and thus impact the overall lead time, cost, and quality[32] of the NPD project.

This subsection investigates the impact of IEC arrival frequency on the three responses while assuming the same duration estimates by associated NPD phase (as appeared in *Table 11*) and resource consumption ($s_{lgm} = 10$) for all incoming IECs. *FRC* still remains as $R_k = 100, k = 1, 2, 3$, and $LCE = max\left(\left(\frac{1}{2}\right)^{N_{ij}}, 0.1\right)$is undertaken.

Three levels of IEC arrival rate will be tested through the design of experiments: **(C)** random monthly ($Random\ (Expo)20$), **(D)** random bi–weekly ($Random\ (Expo)10$), and **(E)** random weekly ($Random\ (Expo)5$). The entire set of scenarios **(B)** from the previous section is served as baseline **(BL3)**, to which the impacts of IEC arrivals will be compared.

Running results of the experiment are displayed in *Table 17*. Note that *quality*, which is served as the third experiment response, appears in column **(iii)**. It is expressed in a relative magnitude by comparing the absolute value of design solution scope to 12,000 of the baseline case which has no IECs accounted for. A resulting number greater than 1 indicates improvement in quality in comparison with the baseline scenario. The percentages of change versus baseline results are shown in column **(III)**.

[32] Design solution scope, an indicator of project quality, is now included to be the third response variable since it becomes a dynamic process variable by the consideration of IECs.

Table 17: Project Performance under the Impacts of IEC Arrival Frequency

IEC ARR	***RL*** **(α,γ)**	***OS***	**(i) Lead Time** *(Days)*	**(I) Time %Change c/w BL3**	**(ii) Project Cost** ($\$ \times 1000$)	**(II) Cost %Change c/w BL3**	**(iii) Quality**	**(III) Quality %Change c/w BL3**
(BL3) (B) No IECs	**(1)** Low $\alpha = \gamma = 0.3$	**(a)** 0%	141		9,542		1	
		(b) 33%	129		9,436		1	
		(c) 66%	**106**		**9,185**		1	
	(2) High $\alpha = \gamma = 0.45$	**(a)** 0%	152		**10,370**		1	
		(b) 33%	158		12,044		1	
		(c) 66%	**123**		11,037		1	
(C) ***Monthly*** Random IECs	**(1)** Low $\alpha = \gamma = 0.3$	**(a)** 0%	145	3.5%	10,952	14.8%	**1.20**	19.9%
		(b) 33%	134	3.5%	10,808	14.5%	1.19	19.0%
		(c) 66%	**108**	2.1%	**10,204**	11.1%	1.15	15.0%
	(2) High $\alpha = \gamma = 0.45$	**(a)** 0%	156	2.7%	**11,877**	14.5%	1.22	22.4%
		(b) 33%	162	2.9%	13,548	12.5%	**1.23**	23.1%
		(c) 66%	**124**	2.7%	12,045	9.1%	1.17	16.6%
(D) ***Bi–Weekly*** Random IECs	**(1)** Low $\alpha = \gamma = 0.3$	**(a)** 0%	152	7.9%	12,343	29.4%	**1.39**	39.1%
		(b) 33%	138	7.2%	11,847	25.6%	1.36	35.7%
		(c) 66%	**114**	7.9%	**11,164**	21.5%	1.28	28.0%
	(2) High $\alpha = \gamma = 0.45$	**(a)** 0%	163	7.5%	13,307	28.3%	1.42	42.3%
		(b) 33%	169	7.1%	15,094	25.3%	**1.45**	44.5%
		(c) 66%	**130**	7.9%	**13,247**	20.0%	1.33	33.5%
(E) ***Weekly*** Random IECs	**(1)** Low $\alpha = \gamma = 0.3$	**(a)** 0%	172	22.5%	15,565	63.1%	**1.83**	82.6%
		(b) 33%	150	15.9%	14,012	48.5%	1.67	67.4%
		(c) 66%	**125**	18.1%	**13,057**	42.1%	1.57	56.6%
	(2) High $\alpha = \gamma = 0.45$	**(a)** 0%	181	19.4%	16,556	59.6%	1.76	75.5%
		(b) 33%	193	22.3%	18,746	55.6%	**1.95**	95.2%
		(c) 66%	**148**	22.6%	**16,076**	45.7%	1.71	70.8%

There are several conclusions can be drawn from the running results shown in *Table 17*:

1. Generally, handling of randomly arriving IECs will cause an increase in both NPD lead time and project cost, which is indicated by the positive values appear in columns **(I)** and **(II)**. It also expands the design solution scope by meeting additional customer requirements that emerge along the process, thus enhances the final product quality. This is reflected by the values in column **(iii)** that are greater than 1.
2. Agreeing with the observation obtained from the previous subsection that project cost is more sensitive to rework than lead time is, project cost is again more responsive to the occurrences of IECs, which is indicated by a larger percentage shown in **(II)** than the one in **(I)**. Also, the differences between these two columns in *Table 17* are much greater on average than the ones in *Table 15*. This is due to the fact that handling of majority of IECs is not on the critical path while most rework is undertaken on the critical path (expect those ones executed concurrently for the overlapped activities with shorter durations), thus IECs have less impact on lead time than rework do.
3. By comparing columns **(I)** through **(III)** to evaluate the impact of IEC arrival frequency, we will find that lead time is subject to an increase at a higher rate compared with cost. Specifically, the results indicate a nearly proportional increase rate in quality and project cost and an exponential growth rate of lead time as more IECs are handled.
4. There are high correlations between the different responses:

   CORREL (Lead Time, Cost) = r_{LC} = 0.873, *CORREL (Lead Time, Quality)* = r_{LQ} = 0.941,

   CORREL (Cost, Quality) = r_{CQ} = 0.900

   Since we model the random IEC arrivals by assigning the Exponential distribution with a specified mean, an NPD process with longer lead time consequently receive more IECs as the project unfolds. This observed fact, in turn, causes repetitive resource congestion phenomenon, and therefore delays the NPD project. Maximum IEC arrivals should be assigned in future research to limit the growth from such a reinforcing loop of IEC occurrences so that various scenarios can be compared more equitably.

5. Even though the handling of IECs results in an overall increase in project cost, when we take a close look at the project cost by separating it into NPD cost and IEC cost, there is no distinct change observed in NPD cost resulting from the IEC arrivals or the frequency of IECs. Actually, NPD cost in fact decreases, on average, by a very slight amount (1.14%) when compared with the baseline case **(BL3)**. That is to say, under current parameter settings, regular NPD activities are not influenced remarkably by the net effect of resource congestion and evolving design solution scope brought about by IECs even in the weekly IEC arrival case.

Plots of functional and total effort committed to the project over time following three overlapping strategies are shown in *Figure 43 – 45*. We can observe that compared with *Figure 38 – 37* there are more sudden stepped functional effort increase as the project evolves over time (lines are more rugged). These frequent changes in resource demand will certainly impose difficulties or hardship to demand management and also increase the non value–added coordinating effort which is not captured by the model presently.

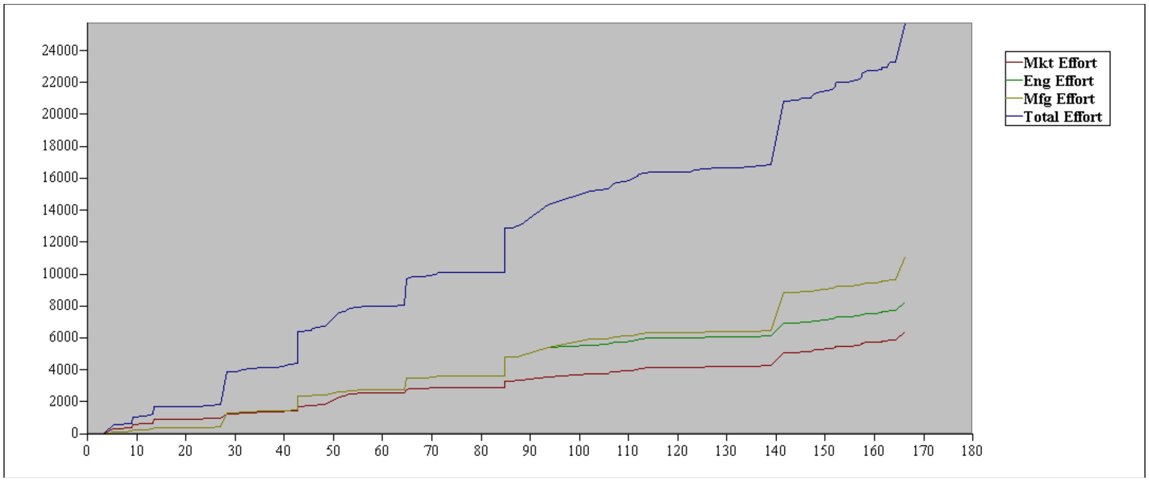


Figure 43: Cumulative Functional Effort and Total Effort w/ IECs(0%)

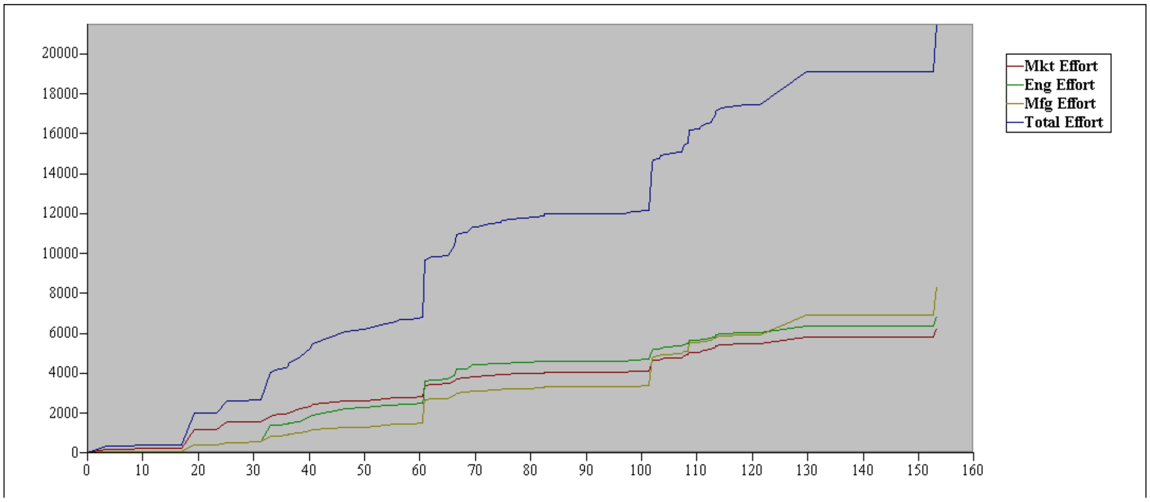


Figure 44: Cumulative Functional Effort and Total Effort w/ IECs (33%)

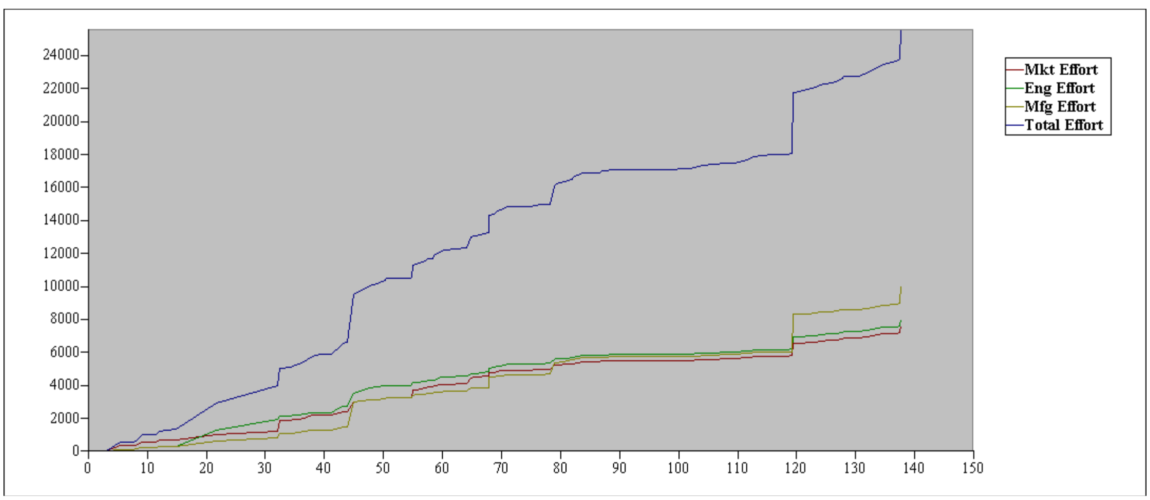


Figure 45: Cumulative Functional Effort and Total Effort w/ IECs (66%)

6.3.4 Combined Impact of IEC Frequency and Size

Experimental design presented in this subsection seeks to explore how different are the impacts of **(F)** half–less frequent (random bi–weekly) but double–size ($s_{lgm} = 20$) IECs on the overall performance as compared with **(E)** random weekly IECs with regular size ($s_{lgm} = 10$). *Table 18* lists the results of baseline case **(BL3: No IECs)**, and then summarizes the absolute and comparative results of two scenarios **(E)** and **(F)**.

From *Table 18*, we observe that **(F)** possesses a "clear" advantage over **(E)** in lead time under each combination of *RL* and *OS* levels. Specifically, **(F)**, on average, leads to 7.15 less days of lead time at low *RL* level ($\alpha = \gamma = 0.3$) and 9.71 less days at high *RL* level ($\alpha = \gamma = 0.45$) compared with **(E)**. Also, **(F)**, on average, leads to 10.15 less days of lead time at low *OS* level (0%), 9.17 less days at medium *OS* level (33%), and 5.99 less days at high *OS* level (66%) compared with **(E)**. We can conclude that the competitive advantage in lead time reduction resulted from batching of IECs is the greatest for a sequential process. And it reduces as overlapping ratio of the PD process increases.

However, neither **(E)** nor **(F)** shows "dominant" advantage in project cost or quality. Differences between results of **(E)** and **(F)** are not as significant as those for lead time. The managerial suggestion behind these numbers is that we may intentionally batch the incoming IECs instead of process them individually to avoid too frequent interruptions to regular NPD activities.

Table 18: Project Performance under the Impact of IEC Size

IEC ARR	*RL* (α, γ)	*OS*	(i) Lead Time *(Days)*	(I) Time %Change c/w BL3	(ii) Project Cost $(\$ \times 1000)$	(II) Cost %Change c/w BL3	(iii) Quality	(III) Quality %Change c/w BL3
(BL3) **(B)** **No IECs**	**(1)** Low $\alpha = \gamma = 0.3$	**(a)** 0%	141		9,542		1	
		(b) 33%	129		9,436		1	
		(c) 66%	**106**		**9,185**		1	
	(2) High $\alpha = \gamma = 0.45$	**(a)** 0%	152		**10,370**		1	
		(b) 33%	158		12,044		1	
		(c) 66%	**123**		11,037		1	
(E) ***Weekly*** Random IECs	**(1)** Low $\alpha = \gamma = 0.3$	**(a)** 0%	172	22.5%	15,565	63.1%	**1.83**	82.6%
		(b) 33%	150	15.9%	14,012	48.5%	1.67	67.4%
		(c) 66%	**125**	18.1%	**13,057**	42.1%	1.57	56.6%
	(2) High $\alpha = \gamma = 0.45$	**(a)** 0%	181	19.4%	16,556	59.6%	1.76	75.5%
		(b) 33%	193	22.3%	18,746	55.6%	**1.95**	95.2%
		(c) 66%	**148**	22.6%	**16,076**	45.7%	1.71	70.8%
(F) ***Bi–Weekly*** Random double–sized IECs	**(1)** Low $\alpha = \gamma = 0.3$	**(a)** 0%	161	14.8%	15,303	60.4%	**1.80**	79.6%
		(b) 33%	144	11.9%	14,177	50.3%	1.70	69.7%
		(c) 66%	**119**	12.9%	**13,069**	42.3%	1.56	56.4%
	(2) High $\alpha = \gamma = 0.45$	**(a)** 0%	172	13.1%	16,302	57.2%	1.84	83.8%
		(b) 33%	180	14.0%	18,166	50.8%	**1.88**	88.4%
		(c) 66%	**141**	17.2%	**15,966**	44.7%	1.70	69.6%

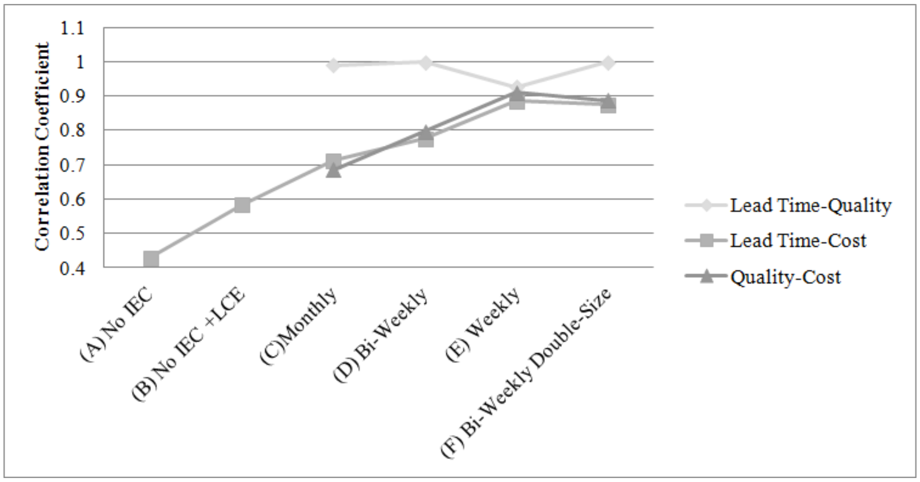


Figure 46: Correlation Coefficient under Different IEC Arrival Frequency

Figure 46 depicts the average correlation coefficients between responses (r_{LQ}, r_{LC}, and r_{QC}) for scenario sets **(A) – (F)**. Since there is no change of design solution scope in **(A)** & **(B)**, correlation coefficients related with Quality (i.e., r_{LQ} and r_{QC}) will not be available from the chart. From the trend lines we can conclude that the correlation coefficients between Cost and the other two responses (i.e., r_{LC} and r_{QC}) are very similar and increase as the random IECs arrive more frequently in **(C) – (E)**. For **(F)**, these two coefficients decrease by a slight amount. On the other hand, the correlation coefficient between Lead Time and Quality r_{LQ} follows a comparatively opposite trend in **(D) – (F)**.

6.3.5 Impact of Resource Constraints

The statistical design presented in this subsection compare the effects of *Functional Resource Constraints (FRC)* on project performance under various combinations of *OS* and *RL* levels. At the same time, the NPD project is influenced by a high level of environmental uncertainty (i.e., weekly random IEC arrivals).

Since there are examination conditions that more resources than the amount required by regular NPD activities are set aside just to handle rework and random IECs, a shorter lead time is achieved in such occasions at the expense of high resource idle cost incurred at the time they are in use. Therefore, when examining the impact of resource constraints (especially in the case of allocating additional resources following a time–driven NPD strategy), it is important to recognize these idle resource costs for the purpose of project planning and control.

In addition to column **(ii)** *Project Cost (PC)*, which is served as the main cost indicator in previous analyses, column **(iv)** *Total Cost (TC)* is captured here to represent the total expenditure on both busy and idle resources. 10 levels of FRC are set up for each scenario, in which the lowest level is chosen to be the sum of:

1) The maximum functional resource demand for a specific *OS* process structure (e.g., 60 for 0% overlapped process; 80 for 33% overlapped process; 100 for 66% overlapped process) which is required when the overlapped activities (e.g., C3/D1 and D3/P1 for 33% overlapped process; C3/D2/P1 for 66% overlapped process) are processed simultaneously;

2) Additional 10 units of resources from each department to handle rework and IECs.

By doing so, the lowest levels of *FRC* for 0%, 33% and 66% *OS* levels are 70, 90, and 110 units of resources, respectively. They are used as the baseline cases **(BL4: Minimum FRC)** for comparison with the performance of scenarios in which more resources will be allocated. Starting from **BL4**, next levels are set by 10– unit increments. Columns **(I) – (IV)** represent the percentages of change (either increase as indicated by a positive number or decrease as indicated by a negative number) of the four model responses, Lead Time, Project Cost, Total Cost, and Quality, as compared to the associated baseline case results.

6.3.5.1 Low Rework Likelihood

Table 19 summarizes the running results that consist of two major parts: 1) mean values of response variables in **(i) – (iv)**, and 2) their percentages of change versus baseline case results in **(I) – (IV)**, under three levels of *OS* (as shown in group **(a)**, **(b)**, and **(c)**) and low *RL* ($\alpha = \gamma = 0.3$).

Detailed analysis will be provided next by interpreting the scatter plots for each combination of every two responses. It is then followed by a brief presentation of results, plots, and observations for scenarios under high *RL* ($\alpha = \gamma = 0.45$) scenarios.

Figure 47 displays three scatter plots grouped by *OS*, showing the relationships between lead time and total cost of the NPD project under various *FRC* levels. *Figure 48*, on the other hand, shows the relationships between the percentages of change in these two responses compared with the baseline case under various *FRC* levels. It provides a convenient and straightforward way of analyzing the trade–offs between time and cost when making the decision of how many resources to allocate. Decision makers could find the optimal *FRC* level by allowing *x–value* in the graph (reduction in lead time) to be as big as possible and *y–value* (increase in total cost) to be as low as possible according to the schedule target, available budget, and overall organizational strategy. Dots from lower right **(BL4)** to upper left in both plots represent an increasing level of *FRC.* Direction of the increase of *FRC* is indicated by an arrow that appears in the lower right corner within each plot.

Table 19: Project Performance under the Impact of *FRC* (Low *RL*)

OS	***FRC*** *(Units of Resource/Dept)*	**(i) Lead Time** *(Days)*	**(I) Time % Change c/w BL4**	**(ii) Project Cost** (*$ × 1000*)	**(II) PC % Change c/w BL4**	**(iv) Total Cost** (*$ × 1000*)	**(IV) TC % Change c/w BL4**	**(iii) Quality**	**(III) Quality % Change c/w BL4**
(a) 0%	**(BL4a)** 70	202		16,179		19,873		**1.9305**	
	80	181	–10.1%	15,571	–3.8%	**19,785**	–0.4%	1.8511	–4.1%
	90	178	–11.8%	15,666	–3.2%	20,913	5.2%	1.8335	–5.0%
	100	172	–14.7%	15,565	–3.8%	21,722	9.3%	1.8260	–5.4%
	110	168	–17.0%	15,400	–4.8%	22,494	13.2%	1.8147	–6.0%
	120	165	–18.1%	15,324	–5.3%	23,458	18.0%	1.8090	–6.3%
	130	163	–19.1%	15,312	–5.4%	24,446	23.0%	1.8052	–6.5%
	140	162	–19.5%	15,358	–5.1%	25,579	28.7%	1.8139	–6.0%
	150	162	–19.6%	15,347	–5.1%	26,725	34.5%	1.8122	–6.1%
	160	**162**	–19.9%	**15,341**	–5.2%	27,819	40.0%	1.8146	–6.0%
(b) 33%	**(BL4b)** 90	160		14,608		**19,125**		**1.7471**	
	100	150	–6.6%	**14,012**	–4.1%	19,160	0.2%	1.6741	–4.2%
	110	151	–5.8%	14,399	–1.4%	20,564	7.5%	1.7087	–2.2%
	120	150	–6.5%	14,466	–1.0%	21,595	12.9%	1.7011	–2.6%
	130	145	–9.7%	14,210	–2.7%	22,043	15.3%	1.6878	–3.4%
	140	144	–10.3%	14,199	–2.8%	22,982	20.2%	1.6787	–3.9%
	150	143	–10.5%	14,178	–2.9%	23,966	25.3%	1.6844	–3.6%
	160	142	–11.0%	14,144	–3.2%	24,884	30.1%	1.6714	–4.3%
	170	**141**	–12.1%	14,076	–3.6%	25,649	34.1%	1.6704	–4.4%
	180	142	–11.4%	14,169	–3.0%	26,877	40.5%	1.6814	–3.8%
(c) 66%	**(BL4c)** 110	122		**13,151**		**17,531**		**1.5638**	
	120	118	–2.9%	13,152	0.0%	18,110	3.3%	1.5376	–1.7%
	130	120	–1.9%	13,423	2.1%	19,234	9.7%	1.5586	–0.3%
	140	116	–4.9%	13,161	0.1%	19,567	11.6%	1.5298	–2.2%
	150	115	–5.4%	13,274	0.9%	20,399	16.4%	1.5285	–2.3%
	160	116	–4.8%	13,465	2.4%	21,433	22.3%	1.5412	–1.4%
	170	115	–5.6%	13,505	2.7%	22,183	26.5%	1.5368	–1.7%
	180	114	–6.1%	13,459	2.3%	22,902	30.6%	1.5324	–2.0%
	190	**114**	–6.7%	13,454	2.3%	23,615	34.7%	1.5336	–1.9%
	200	114	–6.6%	13,424	2.1%	24,447	39.5%	1.5324	–2.0%

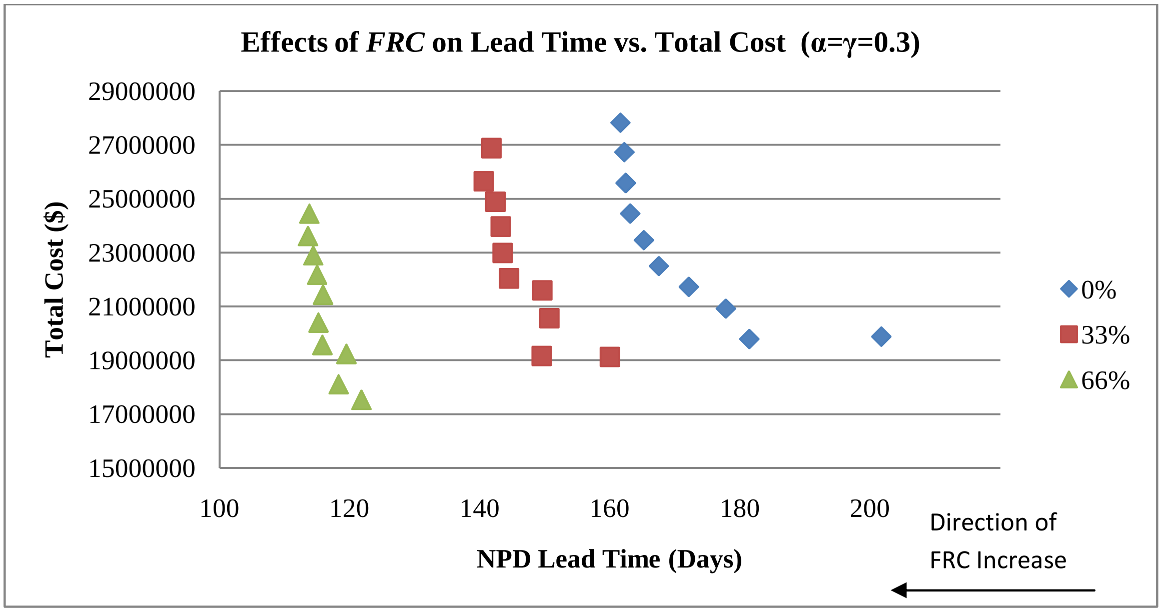


Figure 47: Effects of *FRC* on Lead Time and Total Cost (Low *RL*)

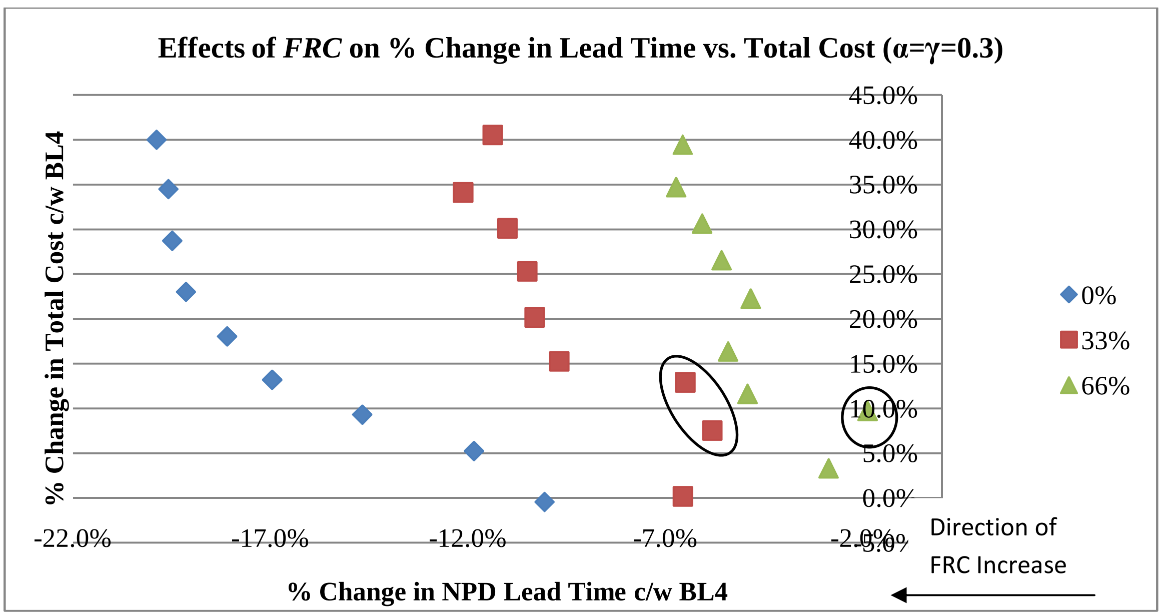


Figure 48: Effects of *FRC* on % Change in Lead Time and Total Cost (Low *RL*)

Several conclusions can be drawn from the above two plots:

1. A higher level of *OS* leads to a shorter NPD lead time and less total cost given the same amount of functional resource allocation, which is illustrated by the shifting lines of data points to the lower left as the *OS* increases in *Figure 47*.
2. However, the percentage of reduction in NPD lead time resulted from an increasing level of *FRC* decreases as the overlapping ratio increases. That is to say, the benefits of lead time reduction by assigning more resources are the most obvious in a sequential process, and activity overlap reduces the degree of obviousness the benefits have. The higher the *OS*, the less the benefits. This is demonstrated by the shifting lines of data points to the right as the *OS* increases in *Figure 48*.
3. For scenarios within group **(a)** (i.e., sequential NPD process), the degree of obviousness the benefits have diminishes as *FRC* increases, which is shown by the decreasing negative slopes between every two adjacent points.
4. Although the running results of the other two groups **(b)** and **(c)** (i.e., 33% and 66% overlapped NPD processes) generally follow a similar time–cost tradeoff trend line as the sequential process, there exist exceptions which are circled out in *Figure 48* that actually shift to the right of the trend lines. For example, the *FRC* level of 110 (units of resource/dept) unexpectedly yields a slight higher NPD lead time than the situation where 10 less resources per department are allocated in a 33% overlapped process.

Figure **49** and *Figure 50* illustrate the relationships between lead time and quality, and between the percentages of change versus baseline of the two responses under various levels of *FRC*, respectively.

Linearity between lead time and quality is observed in all three *OS* levels: the higher the functional resource availability, the shorter the lead time, and the lower the quality. Such linearity has already been stated in the previous two subsections. Recall that we use design solution scope, which is the total amount of person–day effort required to meet the whole set of product goals, to reflect the quality of the final product. And also, design solution scope is evolving along the course of the project. Under this definition, the observation has a straightforward explanation: the longer the lead time, the more random IECs will occur and to be processed, and therefore resulting in a higher product quality. Again, the definition of quality requires further examination and refinement in future work, especially by linking with the solution uncertainty of the final product.

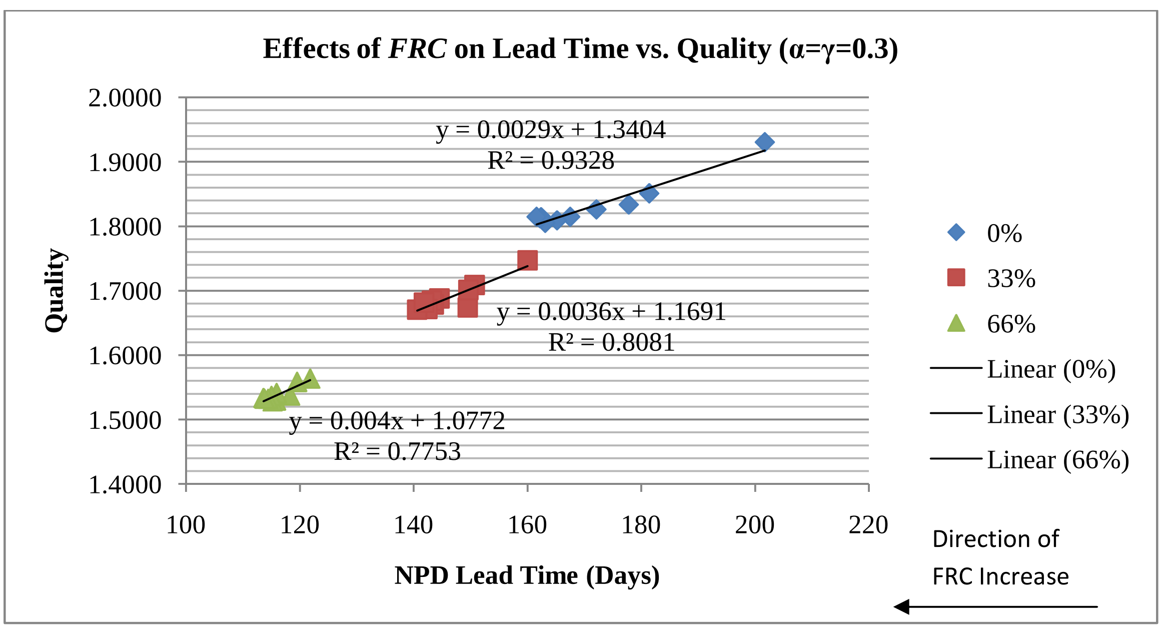


Figure 49: Effects of *FRC* on Lead Time and Quality (Low RL)

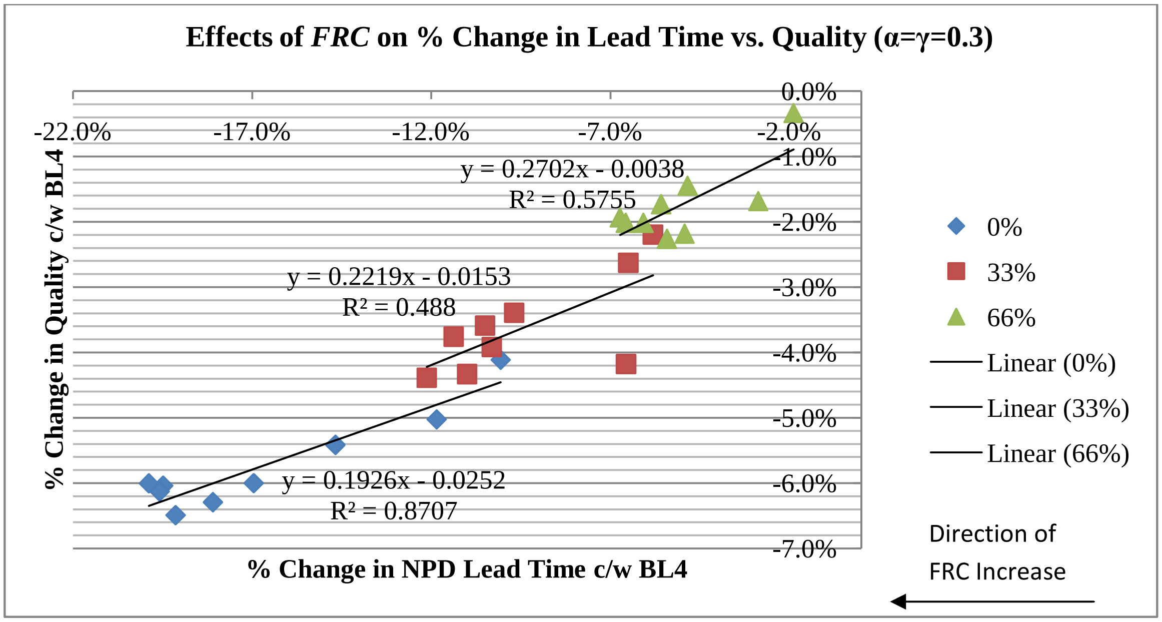


Figure 50: Effects of *FRC* on % Change in Lead Time and Quality (Low *RL*)

Figure **49** reveals the fact that the linearity slope (a) between lead time and quality increases as the *OS* increases. That is to say, the reduction in NPD lead time achieved by assigning more resources will lead to a quality decrease, and the decrease runs at a slower rate under a lower *OS*. On the other hand, as illustrated in *Figure 50*, the percentage of decrease in quality versus baseline case is the largest in a sequential process and decreases as *OS* increases. But again, the rate of the percentage decrease in quality as the NPD lead time reduces declines at a slow pace under lower level of *OS*.

Figure 51 and *Figure 52*, similarly, illustrate the two relationships between total cost and quality. Since the analysis of *FRC* (i.e., to reduce the NPD lead time by allocating more resources) is basically time–driven instead of quality–driven, not much insight can be drawn from these two plots except the fact that degree of quality drop decreases as the *OS* increases revealed in *Figure 51*, which agrees with the trend shown in

Figure **49**.

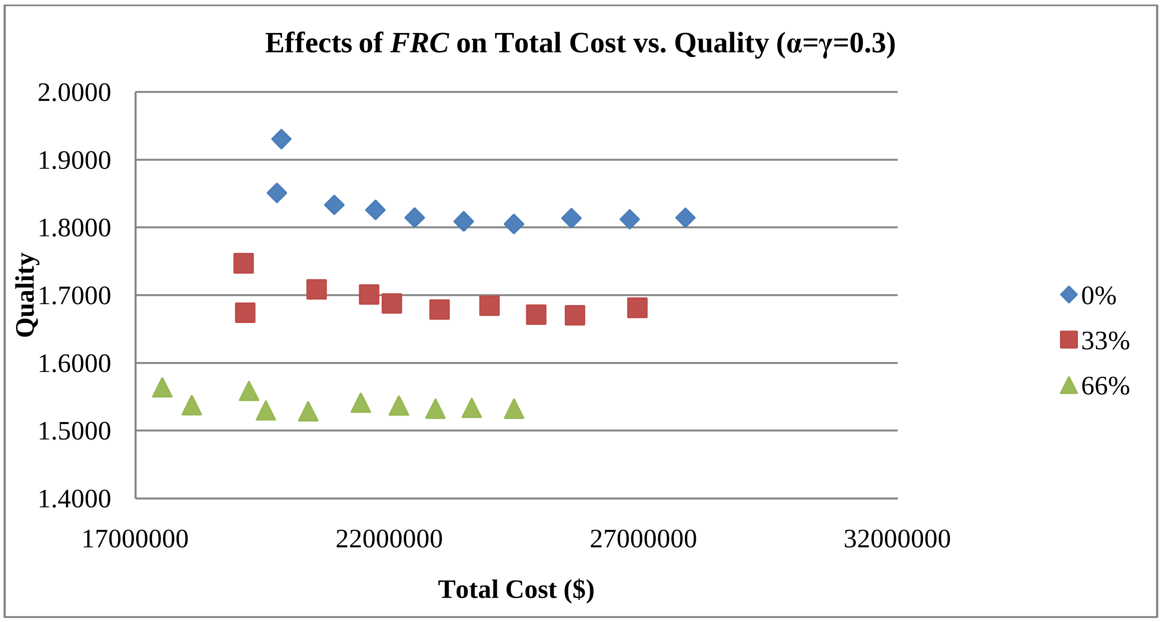


Figure 51: Effects of *FRC* on Total Cost and Quality (Low *RL*)

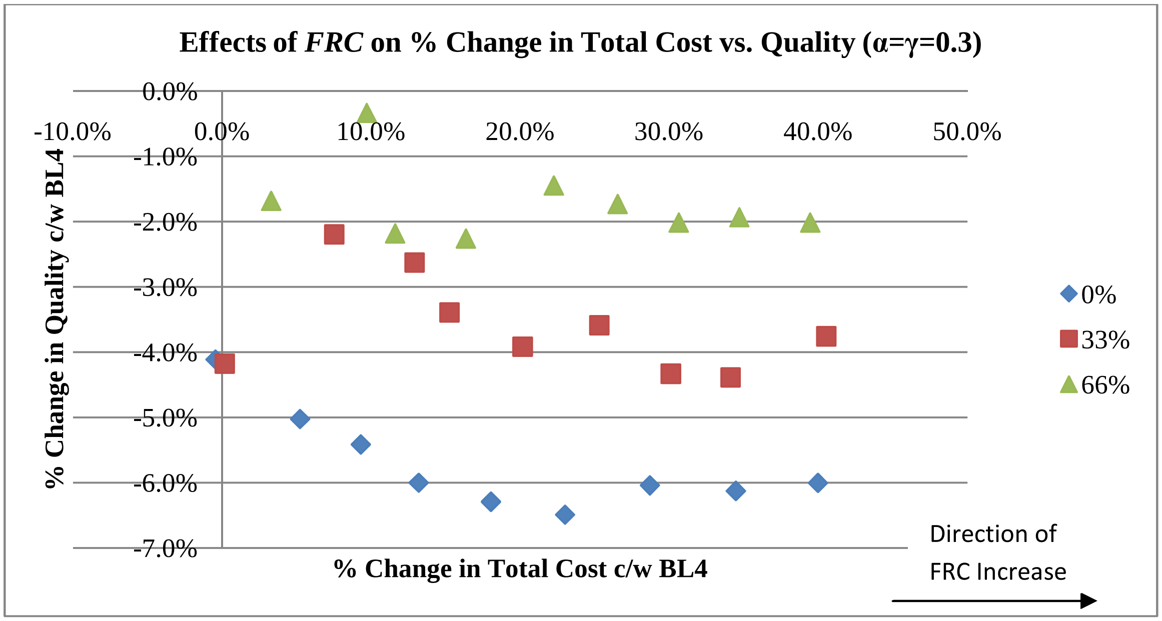


Figure 52: Effects of *FRC* on % Change in Total Cost and Quality (Low *RL*)

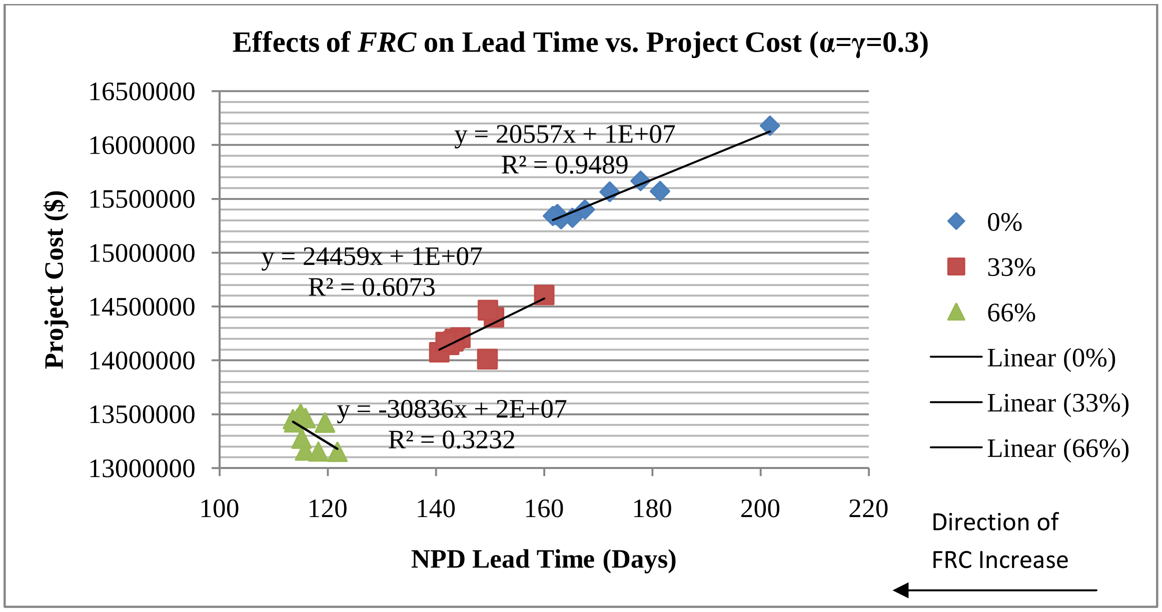


Figure 53: Effects of *FRC* on Lead Time and Project Cost (Low *RL*)

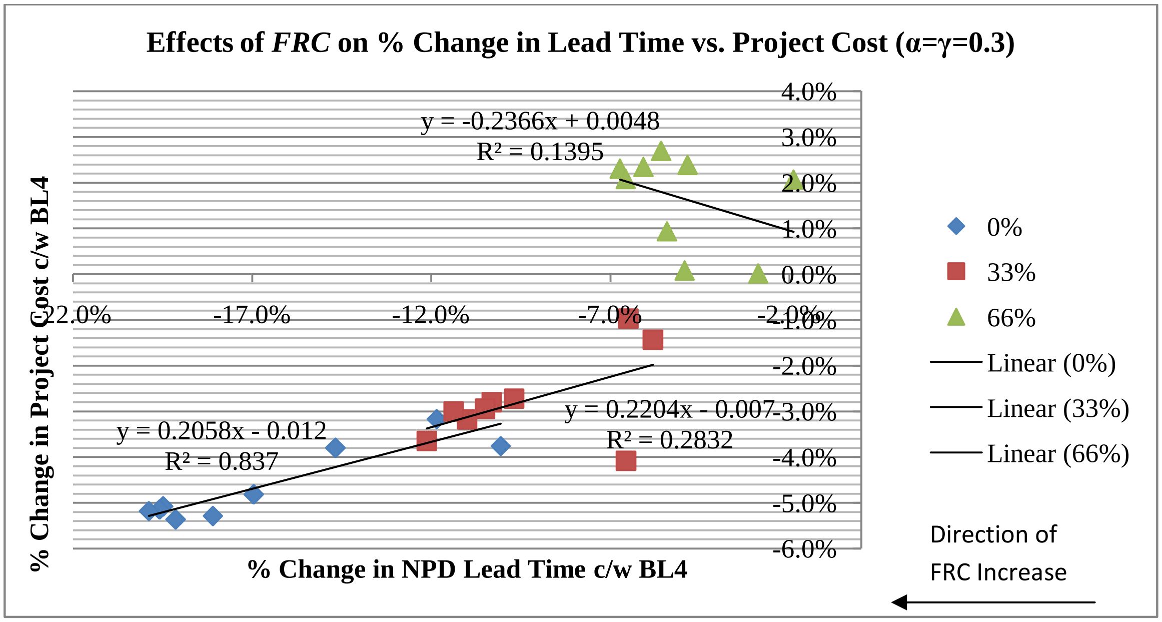


Figure 54: Effects of *FRC* on % Change in Lead Time and Project Cost (Low *RL*)

Figure 53 and *Figure 54* illustrate the relationships between lead time and project cost, and between the percentages of change versus baseline of the two responses, respectively. We observed an unexpected low *negative* correlation between NPD lead time and project cost ($(a_{L(PC)})_c = -30836, (r_{L(PC)})_c = 0.323$) and also a low negative correlation between the percentage of change in these two responses as compared with baseline ($(a_{\%L(\%PC)})_c = -0.237, (r_{\%L(\%PC)})_c = 0.140$) in 66% OS, while in 0% and 33% scenarios high *positive* correlations are displayed ($(r_{L(PC)})_a = 0.949, (r_{L(PC)})_b = 0.607, (r_{\%L(\%PC)})_a = 0.837$) with a few exceptions. That is to say, in 0% and 33% processes, an increase of functional resource availability leads to a reducing lead time indicated by the negative percentage of change in column **(VII)**, and a corresponding reducing project cost indicated by the negative number in **(VIII)**. In a 66% process, an increase of functional resource availability similarly leads to a reducing lead time, and, on the contrary, an increasing project cost indicated by the positive percentage of change in **(VIII)**.

In order to find out reasons behind this unexpected increase in project cost, we further examined both committed NPD effort and IEC effort of each scenario. The results are shown in *Table 20*. Columns **(V)** and **(VI)** are the percentage change of the NPD effort and the IEC effort compared with **BL4**, respectively. Note that NPD effort includes effort spent in both NPD base work (around 12,000 person–days and subject to activity uncertainty), and rework in terms of iterations and EECs. An obvious increase, which is represented in bold in *Table 20*, can be observed within **(c)–(V)** as compared with the other two *OS* levels. That is, an increase of *FRC* in a 66% overlapped process tends to bring about more NPD rework while there is no apparent relationship shown between *FRC* and overall NPD effort in a sequential or a 33% overlapped process. On the other hand, the decreasing trend within each OS level and the increasing trend from **(a)** to **(c)** in column **(VI)** can be explained by the high positive correlation between lead time and occurrence of IECs.

Table 20: NPD and IEC Effort under the Impact of *FRC* (Low *RL*)

OS	***FRC*** *(Units of Resource/Dept)*	**Mkt Effort** *(person–days)*	**Eng Effort** *(person–days)*	**Mfg Effort** *(person–days)*	**NPD Effort** *(person–days)*	**(V) NPD Effort % Change c/w BL4**	**IEC Mkt Effort** *(person–days)*	**IEC Eng Effort** *(person–days)*	**IEC Mfg Effort** *(person–days)*	**IEC Effort** *(person–days)*	**(VI) IEC Effort % Change c/w BL4**
(a) 0%	**(BL4a)** 70	4,063	4,615	7,121	15,799		3,722	3,722	3,722	11,166	
	80	4,030	4,631	7,077	15,739	–0.4%	3,404	3,404	3,404	10,213	–8.5%
	90	4,118	4,705	7,285	16,109	2.0%	3,334	3,334	3,334	10,002	–10.4%
	100	4,088	4,707	7,234	16,030	1.5%	3,304	3,304	3,304	9,912	–11.2%
	110	4,058	4,651	7,181	15,891	0.6%	3,259	3,259	3,259	9,776	–12.5%
	120	4,052	4,637	7,143	15,831	0.2%	3,236	3,236	3,236	9,708	–13.1%
	130	4,057	4,649	7,151	15,857	0.4%	3,221	3,221	3,221	9,663	–13.5%
	140	4,046	4,645	7,139	15,830	0.2%	3,255	3,255	3,255	9,766	–12.5%
	150	4,050	4,636	7,145	15,831	0.2%	3,249	3,249	3,249	9,747	–12.7%
	160	4,046	4,626	7,121	15,793	0.0%	3,258	3,258	3,258	9,775	–12.5%
(b) 33%	**(BL4b)** 90	4,471	5,082	5,743	15,295		2,988	2,988	2,988	8,965	
	100	4,380	5,048	5,768	15,197	–0.6%	2,696	2,696	2,696	8,089	–9.8%
	110	4,446	5,123	5,824	15,393	0.6%	2,835	2,835	2,835	8,504	–5.1%
	120	4,482	5,169	5,914	15,565	1.8%	2,805	2,805	2,805	8,414	–6.1%
	130	4,420	5,077	5,791	15,289	0.0%	2,751	2,751	2,751	8,254	–7.9%
	140	4,427	5,141	5,811	15,379	0.5%	2,715	2,715	2,715	8,144	–9.2%
	150	4,395	5,086	5,809	15,291	0.0%	2,738	2,738	2,738	8,213	–8.4%
	160	4,419	5,122	5,836	15,377	0.5%	2,686	2,686	2,686	8,057	–10.1%
	170	4,390	5,099	5,780	15,270	–0.2%	2,682	2,682	2,682	8,045	–10.3%
	180	4,405	5,119	5,783	15,306	0.1%	2,726	2,726	2,726	8,177	–8.8%
(c) 66%	**(BL4c)** 110	4,599	4,879	5,992	15,471		2,255	2,255	2,255	6,766	
	120	4,688	4,949	6,033	15,670	**1.3%**	2,150	2,150	2,150	6,451	–4.7%
	130	4,668	4,996	6,137	15,801	**2.1%**	2,234	2,234	2,234	6,703	–0.9%
	140	4,643	4,965	6,119	15,727	**1.7%**	2,119	2,119	2,119	6,357	–6.0%
	150	4,713	5,041	6,243	15,996	**3.4%**	2,114	2,114	2,114	6,343	–6.3%
	160	4,767	5,095	6,318	16,180	**4.6%**	2,165	2,165	2,165	6,495	–4.0%
	170	4,809	5,156	6,357	16,322	**5.5%**	2,147	2,147	2,147	6,442	–4.8%
	180	4,801	5,143	6,343	16,287	**5.3%**	2,130	2,130	2,130	6,389	–5.6%
	190	4,811	5,155	6,317	16,283	**5.2%**	2,135	2,135	2,135	6,404	–5.4%
	200	4,804	5,153	6,329	16,286	**5.3%**	2,129	2,129	2,129	6,388	–5.6%

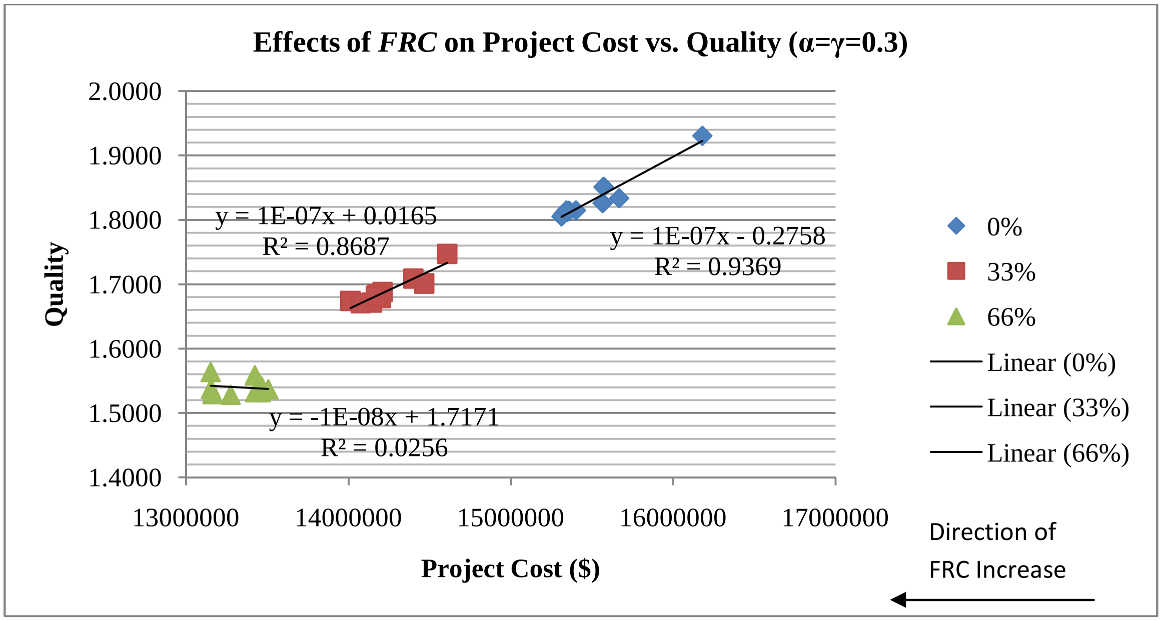


Figure 55: Effects of *FRC* on Project Cost and Quality (Low *RL*)

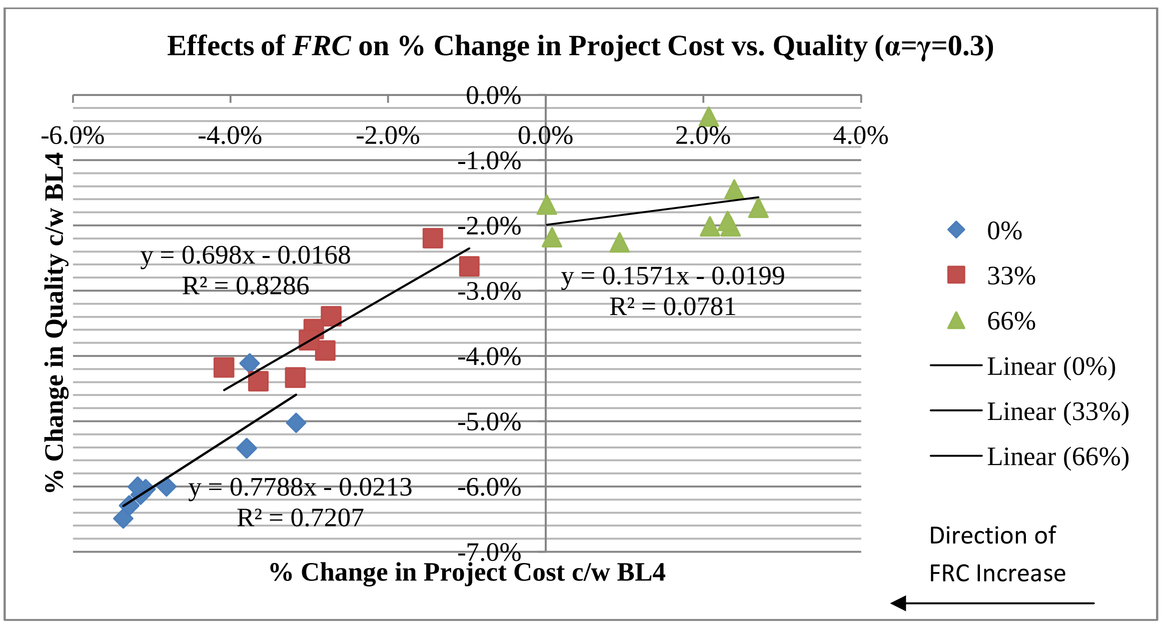


Figure 56: Effects of *FRC* on % Change in Project Cost and Quality (Low *RL*)

Figure 55 and *56* illustrate the relationships between project cost and quality, and the percentages of change versus baseline of the two responses, respectively. Almost the same pattern as appear in *Figure 53* and *54* is observed here.

High linearity between project cost and quality in both absolute ($(r_{(PC)Q})_a = 0.937, (r_{(PC)Q})_b = 0.869$) and relative ($(r_{(\%PC)\%Q})_a = 0.721, (r_{(\%PC)\%Q})_b = 0.829$) values for 0% and 33% overlapped processes. However, there is less of a relationship between project cost and quality observed in neither absolute ($(r_{(PC)Q})_c = 0.026$) nor relative ($(r_{(\%PC)\%Q})_c = 0.078$) values for the 66% overlapped process.

6.3.5.2 High Rework Likelihood

Running results (*Table 21*) and the corresponding pairs of scatter plots (*Figure 57* through *66*) between model responses at high *RL* level are presented in this subsection. Major differences between the results under the two *RL* levels are concluded as follows:

1. At high RL level, an upper–right shift of data points of the 33% overlapped process is observed in *Figure 57*. To be more specific, while still holds a slight advantage in NPD lead time, the total cost of a 33% overlapped process surpasses that of a sequential one.
2. At the high RL level, a left shift of data points of the 66% overlapped process is observed in *Figure 58*. The high level of OS shows an improved reduction in lead time as compared with the baseline case.
3. At the high RL level, the correlation coefficient between lead time and quality increases for all OS levels. The correlation coefficient between the percentage changes of the two responses also increases, especially for 33% and 66% OS levels by a significant amount.
4. At high RL level, the value of column **(c)–(VIII)** is no longer positive. That is, the project cost also decreases when more functional resources are allocated in the same way of what happens in scenarios under the other two OS levels in both RL levels.

5. At the high RL level, the relationships between project cost and lead time, and between project cost and quality, remains almost the same as at the low RL level.

To conclude, there is no unique resource allocation policy that optimizes all three performance indicators. Through a full comprehension of the importance of each performance indicator and its relation with the overall goal, further tradeoff studies should be conducted to ultimately make robust decisions.

By looking at the two extreme levels we can find that allocating only the lowest resource level yields a fairly long NPD lead time due to the resource congestion phenomenon especially occurred during overlaps. However, allocating much more resources than needed by regular NPD activities will alleviate resource congestion when IECs arise but lead to a much higher total cost owing to the high idle cost when resources are not in use. In reality, companies typically execute several NPD projects in parallel and share the same resources across projects. This situation of high idle cost will be mitigated but at the expense of a changing rate of learning when resources are being switched among different projects.

Table 21: Project Performance under the Impact of *FRC* (High *RL*)

OS	***FRC*** *(Units of Resource/Dept)*	**(i) Lead Time** *(Days)*	**(I) Time % Change c/w BL4**	**(ii) Project Cost** *($ × 1000)*	**(II) PC % Change c/w BL4**	**(iv) Total Cost** *($ × 1000)*	**(IV) TC % Change c/w BL4**	**(iii) Quality**	**(III) Quality % Change c/w BL4**
(a) 0%	**(BL4a)** 70	219		17,702		21,663		**2.0416**	
	80	195	–11.1%	16,771	–5.3%	**21,279**	–1.8%	1.9256	–5.7%
	90	187	–14.9%	16,544	–6.5%	22,005	1.6%	1.8818	–7.8%
	100	181	–17.2%	16,556	–6.5%	22,987	6.1%	1.8853	–7.7%
	110	177	–19.4%	16,333	–7.7%	23,778	9.8%	1.8650	–8.6%
	120	176	–19.6%	16,462	–7.0%	25,077	15.8%	1.8820	–7.8%
	130	174	–20.8%	16,333	–7.7%	26,028	20.2%	1.8611	–8.8%
	140	173	–21.2%	**16,321**	–7.8%	27,198	25.5%	1.8563	–9.1%
	150	**172**	–21.4%	16,345	–7.7%	28,405	31.1%	1.8613	–8.8%
	160	173	–21.3%	16,451	–7.1%	29,745	37.3%	1.8687	–8.5%
(b) 33%	**(BL4b)** 90	195		18,334		**23,641**		**1.9312**	
	100	193	–1.2%	18,746	–2.2%	25,126	6.3%	1.9518	–1.1%
	110	185	–5.0%	18,309	–0.1%	25,650	8.5%	1.9180	–0.7%
	120	178	–8.8%	18,038	–1.6%	26,203	10.8%	1.8788	–2.7%
	130	173	–11.3%	**17,816**	–2.8%	26,891	13.7%	1.8504	–4.2%
	140	171	–12.3%	17,888	–2.4%	27,970	18.3%	1.8396	–4.7%
	150	170	–12.9%	18,021	–1.7%	29,168	23.4%	1.8498	–4.2%
	160	167	–14.4%	17,842	–2.7%	29,938	26.6%	1.8368	–4.9%
	170	**167**	–14.7%	17,754	–3.2%	31,024	31.2%	1.8243	–5.5%
	180	167	–14.7%	17,931	–2.2%	32,351	36.8%	1.8440	–4.5%
(c) 66%	**(BL4c)** 110	144		16,253		**21,168**		**1.7057**	
	120	139	–3.8%	16,015	–1.5%	21,590	2.0%	1.6723	–2.0%
	130	134	–7.4%	**15,703**	–3.4%	21,918	3.5%	1.6418	–3.7%
	140	132	–8.3%	15,826	–2.6%	22,818	7.8%	1.6356	–4.1%
	150	131	–9.5%	16,021	–1.4%	23,708	12.0%	1.6281	–4.5%
	160	128	–11.4%	15,973	–1.7%	21,433	14.8%	1.6246	–4.8%
	170	127	–12.1%	15,888	–2.2%	24,309	18.3%	1.6197	–5.0%
	180	**125**	–13.5%	15,750	–3.1%	25,040	21.0%	1.5941	–6.5%
	190	125	–13.2%	15,867	–2.4%	26,631	25.8%	1.6096	–5.6%
	200	126	–12.9%	16,067	–1.1%	27,731	31.0%	1.6117	–5.5%

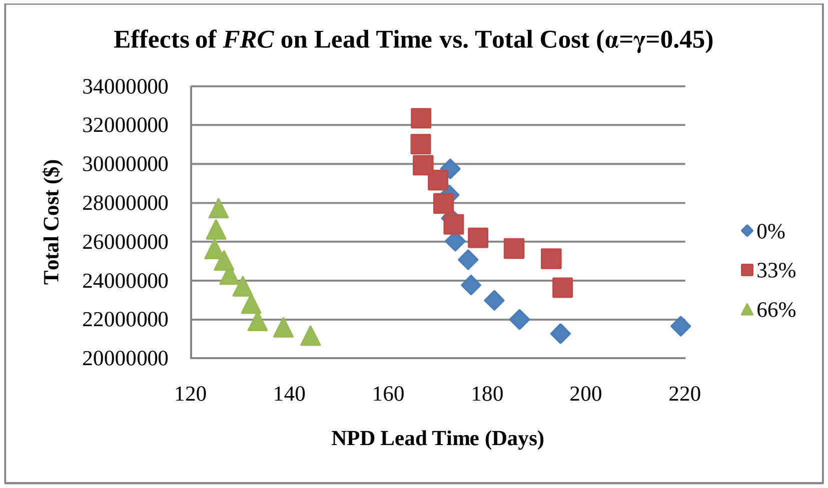


Figure 57: Effects of *FRC* on Lead Time and Total Cost (High *RL*)

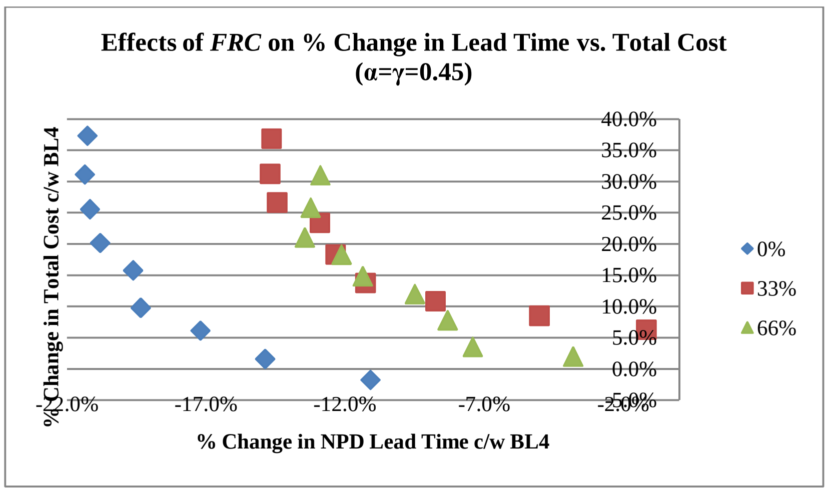


Figure 58: Effects of *FRC* on % Change in LT and TC (High *RL*)

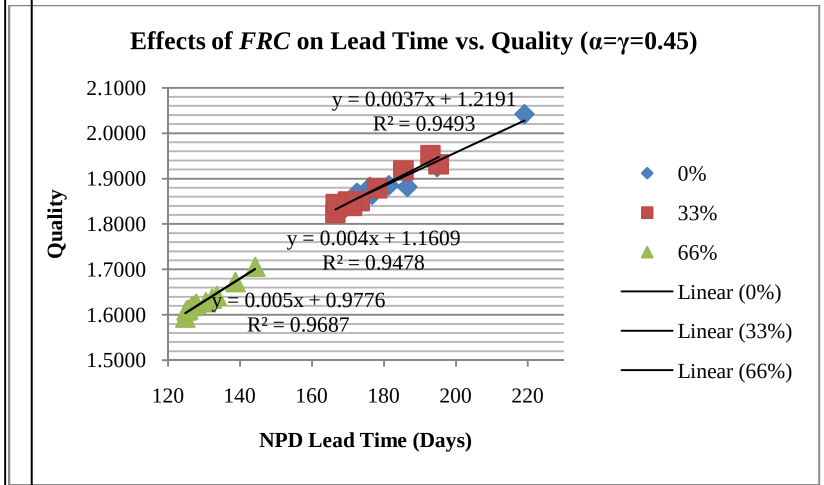


Figure 59: Effects of *FRC* on Lead Time and Quality (High *RL*)

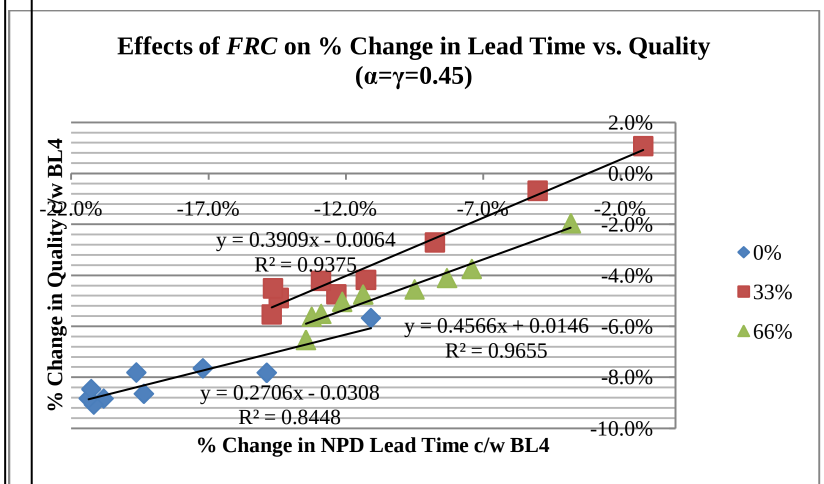


Figure 60: Effects of *FRC* on % Change in PC and Quality (High *RL*)

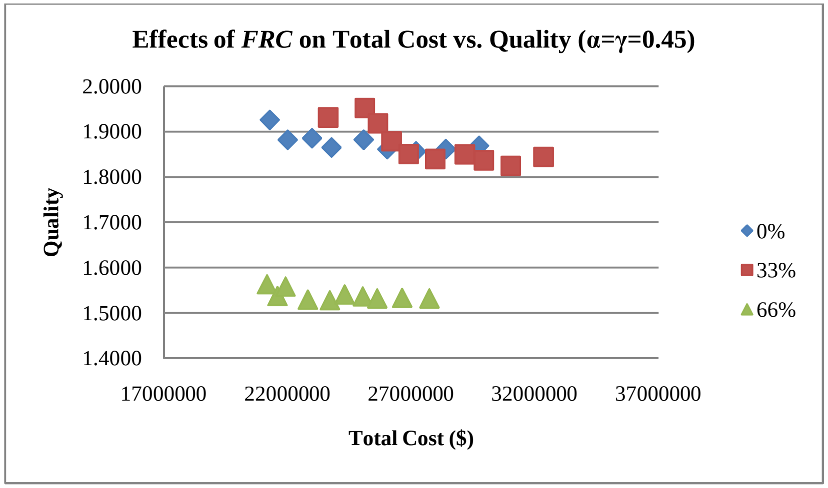


Figure 61: Effects of *FRC* on Total Cost and Quality (High *RL*)

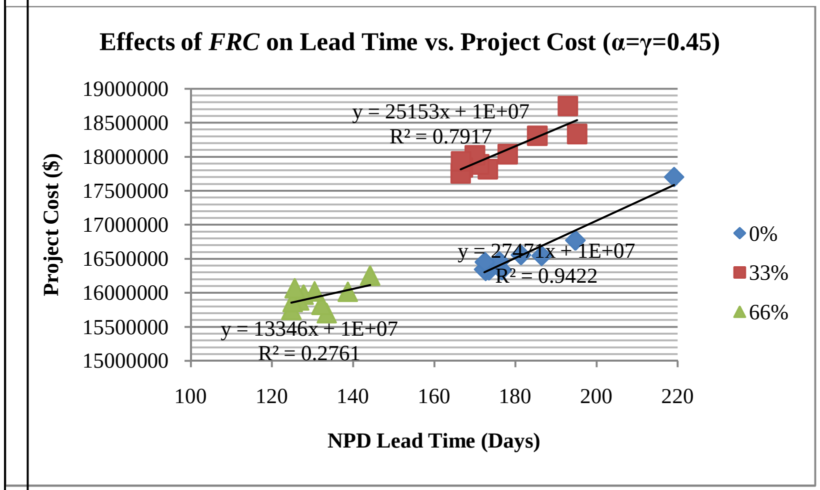


Figure 63: Effects of *FRC* on Lead Time and Project Cost (High *RL*)

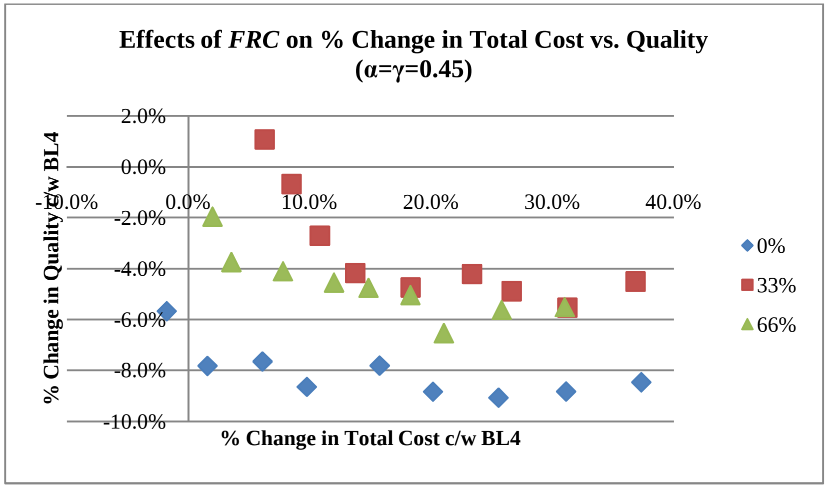


Figure 62: Effects of *FRC* on % Change in TC and Quality (High *RL*)

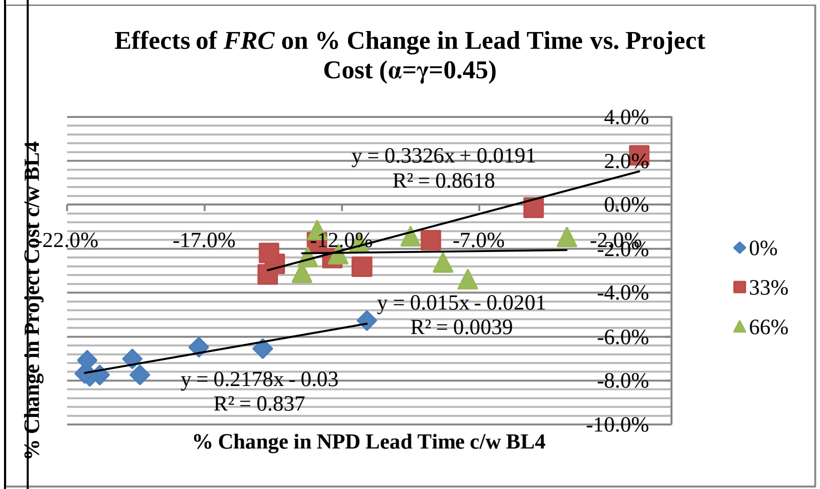


Figure 64: Effects of *FRC* on % Change in LT and PC (High *RL*)

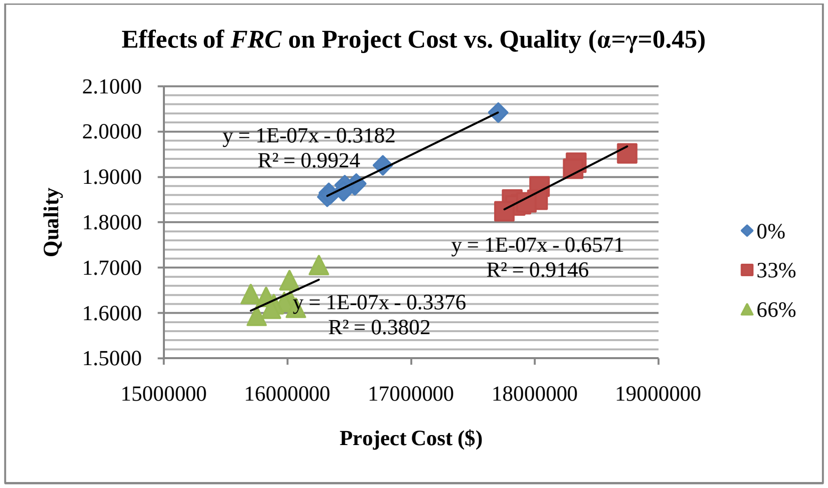


Figure 65: Effects of *FRC* on Project Cost and Quality (High *RL*)

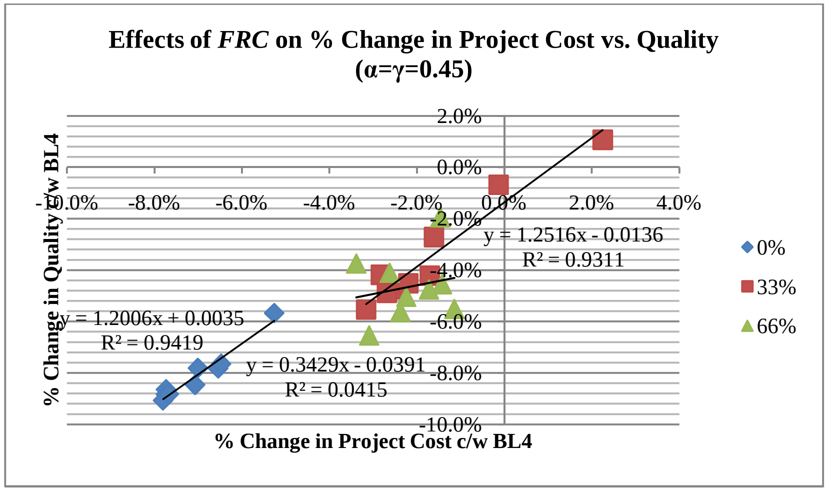


Figure 66: Effects of *FRC* on % Change in PC and Quality (High *RL*)

6.3.6 Impact of Change Propagation due to Product Configuration

Not only the PD process activities but the product architecture as well are closely dependent and may highly likely trigger change propagation from one to another (Eckert, Clarkson, and Zanker 2004; Rutka, et al. 2006; Koh and Clarkson 2009; Krishna and Moon, 2009). In the second set of simulation experiments, in addition to change propagation phenomenon of IECs due to the couplings between PD activities that has already been captured in previous policy analyses, the nature and extent of change propagation due to a high degree of coupling among constituent product components and systems will be discussed by specifying the accurate dependency information of a product configuration and integrating it into the IEC section of the model.

6.3.6.1 Additional Model Inputs

Figure 67 shows two simple product architecture examples that will be used to demonstrate simulation procedure and logic in analyzing the impacts of ***Product Configuration (PC)*** on change propagation.

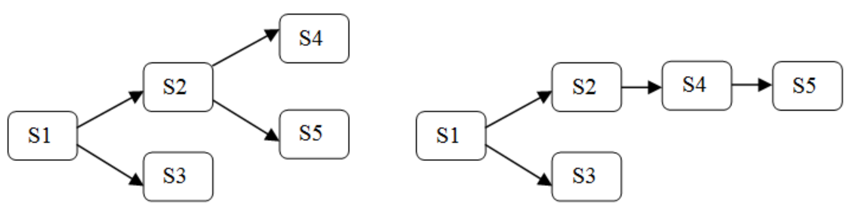


Figure 67a/b: Two Different 5–System Product Configuration

Both of them have five interrelated systems but different numbers of levels in their product breakdown structures. For the product configuration that consists 3 levels of system (abbreviated as *LevelOfSys*) shown on the left, system *S1* on the top level is interrelated with two other systems, *S2* and *S3*. It goes down only one level for *S3* while systems *S4* and *S5* simultaneously and independently interact with *S2*.

It is assumed that IECs to parent system will uni–directionally propagate to its children system(s) – but not the other way around. That is to say, IECs to *S1* will propagate to *S2* and *S3*, and IECs to *S2* will propagate to *S4* and *S5*. Since *S3*, *S4* and *S5* are at the bottom level, changes to them will not cause any propagation. It is also assumed that during the IEC propagation, changes to the children system(s) are triggered simultaneously by the completion of their parent. For instance, the propagation of IEC from *S2* to *S4* and *S5* will occur at the same time if there are enough resources available.

The product architecture shown on the right illustrates a 4 *LevelOfSys* configuration. System *S4* of this product configuration is interrelated with *S5*. Any change to *S4* will further propagate to *S5*.

6.3.6.2 Modification of Model Logic

The following two process flow diagrams illustrate the enhanced model logic of IEC section (diagram above reflects the 3 LevelofSys *PC* and the one below reflects the 4 LevelofSys *PC*) by taking into account the change propagation phenomenon due to complex product configuration of interconnected components and systems. Note that the simulation procedure of *IEC process propagation* shown in

Figure **35** is now nested inside an outer loop of *IEC product propagation* according to the dependency properties among product systems that have been visualized in *Figure 67*.

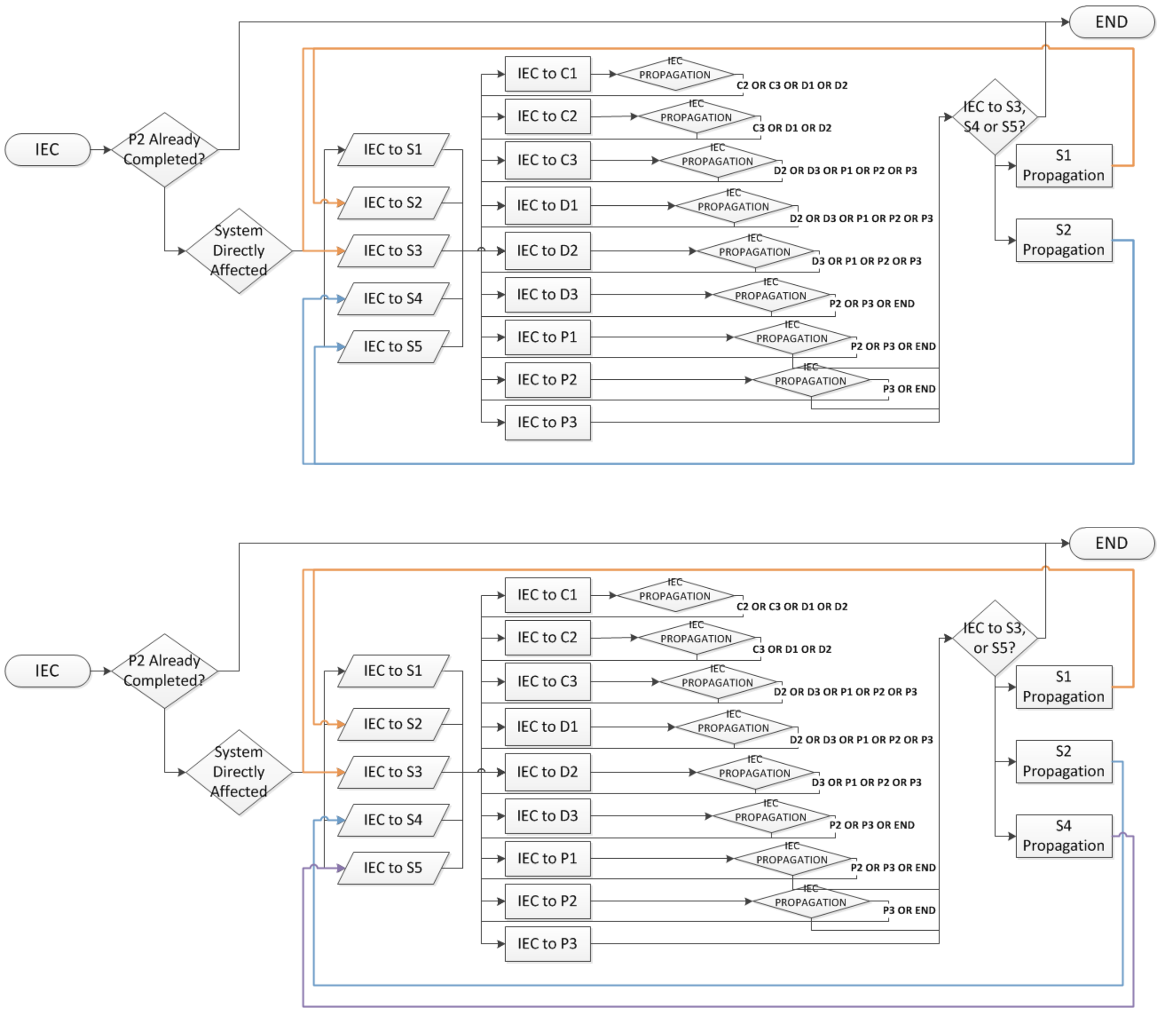


Figure 68a/b: Overview of IEC Section for 33% Overlapping (Coupled PD Activities and Coupled Product Configuration)

APPENDIX F shows the process flow of how IEC propagation among interdependent product systems is actually modeled in Arena.

6.3.6.3 Result Analysis

Model input settings are chosen as follows. *FRC* is kept at the level of $R_k = 100, k = 1, 2, 3$ and $LCE = max\left(\left(\frac{1}{2}\right)^{N_{ij}}, 0.1\right)$ is examined. The NPD project is under various levels of environmental uncertainty with regular size ($s_{lgm} = 10$).

Concerning both space and time limits, only 33% overlapping strategy is examined and analyzed here to illustrate how the proposed model is applicable in the impact analysis of IEC propagation due to interconnected product configuration. Running results and their percentage changes from baseline (**BL5**), in which product architecture couplings are not considered, are shown in *Table 22*. There are several observations as well as preliminary conclusion statements can be drawn from the results obtained from the 33% overlapped process:

1. When the effects of change propagation due to the interconnected product configuration are taken into account, results show a general increasing trend, which is indicated by positive values appeared in columns **(I)**, **(II)**, and **(III)**, in the multiple dimensions of NPD project performance (i.e., lead time, project cost, and quality) from baseline case.
2. However, there is one noticeable exception to the common increase: unexpected decreases in NPD project lead time are observed for the 3 LevelofSys product configuration (i.e., **(*PC1*) (C) & (E)**). In particular, dramatic decreases are caught in the scenario of weekly random IEC arrivals, especially at a high level of RL: **(*PC1*) (E) (2)**.
3. When the effects of change propagation due to *PC* are taken into account, some of the high correlations between model responses that have been observed in previous analyses diminish. Specifically, while the correlation between Project Cost and Quality still remains high (r_{CQ} = 0.962), the correlation between Lead Time and Project Cost (r_{LC} = 0.536) and the one between Lead Time and Quality (r_{LQ} = 0.386) drops significantly as compared to the results shown in *Figure 46*.

4. ***Effects of IEC ARR:*** the influence of change propagation due to *PC* increases as the environmental uncertainty increases. In general, we observe that the more frequent the IEC arrivals, the larger percentage changes of the three model responses from **BL5**.
5. ***Effects of RL:*** there is no clear trend in the impacts of *RL* on the three model responses when effects of change propagation due to *PC* are taken into account. By comparing the differences between data of columns **(I)**, **(II)**, and **(III)** in rows **(1)** and those in rows **(2)**, we observe both increases and decreases in model responses when *RL* goes from Low **(1)** to High **(2)**.
6. ***Combined effects of RL & IEC ARR:*** however, by comparing data of columns **(I)**, **(II)**, and **(III),** from **(1)** to **(2)** and through scenarios **(C)** to **(E)**, another counterintuitive project behavior can be perceived: for weekly IEC arrivals **(E)**, a high *RL* reversely leads to a lower percentage of increase in all three responses as compared to results of a low *RL*, except in two cases: **(PC1) (E) (I)** and **(PC2) (E) (III)**.

Table 22: Project Performance under the Impacts of Product Configuration (33% Overlapping Strategy)

Product Configuration	**IEC ARR**	***RL* (α, γ)**	**(i) Lead Time** *(Days)*	**(I) Time % Change c/w BL5**	**(ii) Project Cost** ($\$ \times 1000$)	**(II) PC % Change c/w BL5**	**(iii) Quality**	**(III) Quality % Change c/w BL5**
(BL5)	**(C)** ***Monthly*** Random IECs	**(1)** Low	134		10,808		1.19	
		(2) High	162		13,548		1.23	
	(D) ***Bi–Weekly*** Random IECs	**(1)** Low	138		11,847		1.36	
		(2) High	169		15,094		1.45	
	(E)***Weekly*** Random IECs	**(1)** Low	150		14,012		1.67	
		(2) High	193		18,746		1.95	
(PC1) **3** ***LevelOfSys***	**(C)** ***Monthly*** Random IECs	**(1)** Low	131	–2.25%	13,047	20.7%	1.54	29.6%
		(2) High	157	–3.02%	16,498	21.8%	1.66	34.8%
	(D) ***Bi–Weekly*** Random IECs	**(1)** Low	139	0.52%	16,017	35.2%	1.98	46.1%
		(2) High	170	0.53%	20,147	33.5%	2.19	51.3%
	(E)***Weekly*** Random IECs	**(1)** Low	127	**–14.80%**	21,968	56.8%	2.91	73.6%
		(2) High	141	**–27.02%**	26,905	43.5%	3.25	66.6%
(PC2) **4** ***LevelOfSys***	**(C)** ***Monthly*** Random IECs	**(1)** Low	137	2.84%	12,532	16.0%	1.46	22.7%
		(2) High	169	4.07%	16,018	18.2%	1.56	27.0%
	(D) ***Bi–Weekly*** Random IECs	**(1)** Low	153	10.49%	15,113	27.6%	1.85	36.1%
		(2) High	185	9.64%	19,623	30.0%	2.09	44.9%
	(E)***Weekly*** Random IECs	**(1)** Low	178	19.03%	22,569	61.1%	2.90	**73.4%**
		(2) High	224	16.26%	28,584	52.5%	3.40	**74.2%**

6.4 Summary

This chapter describes and presents a simple numerical example to show how the proposed simulation model works, and to study the impacts of different product, process, team, and environment characteristics on project performance measures and how various NPD and ECM policy decisions could be systematically evaluated.

The NPD section of model framework is first implemented to analyze the impact of NPD process features and *rework review strategy*. Then, the IEC section is included to explore the impact of IEC arrival frequency, *IEC batching policy*, and *resource assignment strategy*. Finally, the IEC section is extended to account for change propagation phenomenon resulted from interconnected product configuration.

Model outputs are presented in both absolute value and relative value (as compared to the results of baseline case), based upon which general observations are made, and conclusions regarding different managerial strategies and coordination policies together with root causes of interesting running phenomenon, especially those counterintuitive ones, are discussed in great details.

CHAPTER 7

CONCLUSIONS

7.1 Summary of Contributions

The principle contributions of this dissertation to the existing body of PD process modeling and ECM modeling literature, from theoretical implications to practical applications, are threefold.

First of all, *Chapter 3* presents *a conceptual exploratory study of four major ECM issues*: i) occurrence of ECs, ii) long EC lead time, iii) high EC cost, and iv) occurrence frequency and magnitude of iterations and ECs. From a systems view, main contribution factors and cause–and–effect relationships between them are identified by creating both causal links and causal feedback loops. This proposed conceptual causal framework is presumably the first systematic investigation of ECM risk drivers at project–level, reflecting common understanding between industry and academia. In particular, occurrence frequency of iterations and ECs in a resource–constrained environment was explicitly explored by building and interpreting a full list of both closed causal feedback loops and causal links considering interrelated system variables such as design solution scope, solution uncertainty, learning curve effects, iteration/EC size, and resource availability, among others. Moreover, *three field survey studies conducted in automotive and information technology industries* are documented at length in *Chapter 4*.

The conceptual causal relationships among factors that have been identified, data collected and evidence observed of the actual NPD/ECM processes and corresponding decision making procedures, together with other validated PD process modeling methodologies and research findings in the existing literature, lay down the foundation to support essential underlying assumptions of the simulation model described in *Chapter 5*.

Secondly, this research proposes a comprehensive Discrete–Event Simulation (DES) model that captures different aspects of PD project–related (i.e., product, process, team, and environment) complexity to investigate their resultant impacts on the occurrence and magnitude of iterations and ECs that stochastically arise during the course of an NPD project, and how the multiple dimensions of project performance, including lead time, cost, and quality, are consequently affected. In addition to the integration of several critical characteristics of PD projects that have been previously developed and tested, (e.g., concurrent and collaborative development process, learning curve effects, resources constraints), this research introduces the following new features and dynamic structures that are explicitly modeled, verified, and validated for the first time:

1) It *explicitly distinguishes between two different types of rework by the time of occurrence*: intra–phase iterations and inter–phase EECs. Moreover, *engineering changes are further categorized into two groups by their causes of occurrence*, emergent ECs “that are necessary to reach an initially defined standard in the product” (Eckert, Clarkson, and Zanker 2004), and initiated ECs in response to new customer requirements or technology advances.
2) *Uncertainty is differentiated and conceptualized into three categories*: low–level activity uncertainty, medium–level solution uncertainty, and high–level environmental

uncertainty. Activity uncertainty is reflected in the stochastic activity duration using probability distributions and environmental uncertainty is primarily modeled by the arrival frequency and magnitude of IECs. In particular, solution uncertainty is an important model variable that dynamically determines the rework probability which will be discussed next.

3) This research provides presumably the first attempt to integrate cause–and–effect relationships among project variables into a DES model of development projects. Traditional DES model deals with only static project features in "open–loop, single–link" causal relationship format (Ford 1995) that remain constant as the model evolves[33]. *Rework probability is no longer pre–determined* and remains fixed over the entire time frame of the NPD process as appeared in most of previous studies. Instead, it is calculated in real time by the model itself. That is to say, rework probability is now included in a *feedback structure* that changes over time in response to the project's evolving uncertainty levels.

4) The specific three–step *rework review process structure*, together with the *rigidity of rework reviews*, allows more explicit and detailed modeling of this critical aspect of ECM, which is not attempted by previous studies. Decision points are used with rules to conditionally process ECs. They also give the users flexibility to define one or more rules in priority evaluation order.

5) *The traditional restrictive assumption of a stable development process with no environmental disturbance is also relaxed* by introducing the random occurrence of IECs,

[33] For example, previous studies of complex product development processes commonly leave rework probabilities unchanged as project progresses.

which leads to an enlarged design solution scope of the final product and thus affecting the project solution uncertainty.

Last but not least, the proposed model framework can be calibrated and used as a decision–support tool to assist ECM practitioners in quantifying the impacts of various managerial strategy and coordination policy alternatives, such as overlapping strategies, rework review policies, IEC batching policies, impacts of coupled product architecture, etc., on the project lead time, cost, and quality from a systematic perspective. This dissertation illustrates in detail how such trade–off studies can be conducted and how simulation results can be interpreted for better NPD and ECM decisions.

7.2 Major Findings

7.2.1 Current Practice of ECM

The qualitative and quantitative observations and findings from the three field survey studies reported in this dissertation reveal important issues for consideration regarding the current practice of ECM:

1. Despite the fact that ECM is confirmed to be of great importance within the surveyed domains and ECs are recognized as main sources that significantly impact the dynamic behavior of NPD projects, companies involved in complex product development and operations are still lacking on systematic formal approach in tracking and managing EC information.

2. Furthermore, even for those companies that utilize computer–aided documentation systems to record, track, communicate and maintain NPD/ECM data on a regular basis, different process participants collect and organize project–related data such as, work breakdown structure, work effort estimates, resource demand and capacity, status reporting (i.e., completion of the WBS tasks), in various forms with different levels of accuracy, and make corresponding updates at different time schedules. This results in varied data quality in terms of data accuracy, completeness, consistency, and timeliness. In addition, the integration of available data residing in various sources with different formats and quality into valid model inputs at project level remains challenging.
3. Risk and impact assessment information is typically provided only to support approval decision about requested changes. However, a series of records of how the EC (or CR) is actually implemented in terms of resource consumption, cycle time, and cost are usually not accumulated.

7.2.2 Combining Process Feedbacks with Discrete Event Simulation to Support NPD & ECM Decisions

This research demonstrates how closed–loop feedback relationships among model variables can be incorporated into a DES model to improve PD project behavior and performance predictions, and thus support NPD and ECM decision makings. Results show under different conditions of uncertainty (i.e., activity uncertainty in terms of deviations from average activity duration, solution uncertainty in terms of learning curve effects and rework likelihood of a particular NPD process, and environmental uncertainty in terms of IEC arrival frequency and magnitude), how we should apply various kinds of strategies and policies, including process overlapping, rework review, IEC batching, resource allocation, to not only achieve benefits but also recognize potential tradeoffs among lead time, cost and quality. Specific conclusions drawn from the research will be discussed further in next subsection.

Figure 69 illustrates, from a higher level, how the simulation model design presented in this dissertation can be possibly implemented. Guidelines of application consist of three main steps: i) data acquisition in terms of product, process, team, and environment information about NDP projects, ii) simulation model construction and selection of design factors, constraints and response variables together with their corresponding levels and range, and iii) decision support analysis & optimization.

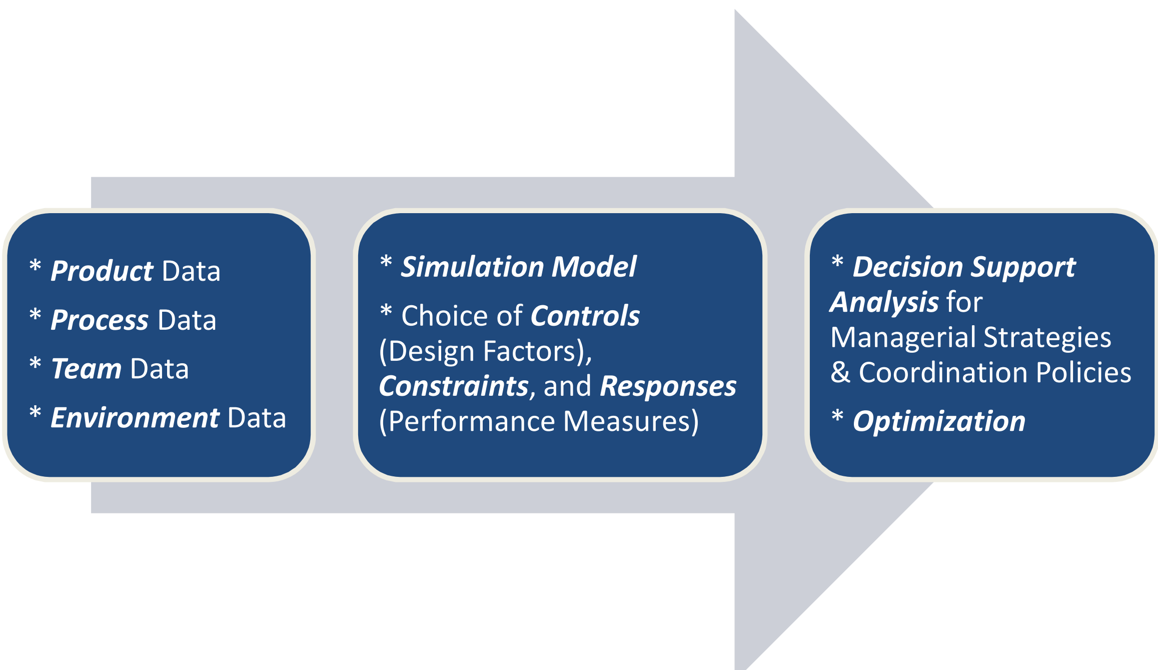


Figure 69: Application Method of ECM Decision Support System

7.2.3 Major Findings from the Simulation Study

The research concludes with the following findings or understandings that either have been identified previously in the existing literature or disclosed for the first time with the help of newly added and verified model features:

1. Significant increase of both time and cost due to rework is alleviated by the evaluation of *LCE*.
2. The percentage increase of project cost is always higher than that of lead time at the occurrence of rework and IECs. That is, compared with lead time, project cost is more sensitive to rework/IECs.
3. By starting downstream activities early with only preliminary information, concurrent engineering tends to alleviate the impacts of rework on activities in downstream phases while intensifying those on activities in the upstream phases. It also tends to shift rework risks and even out committed efforts among various functional areas. In addition, departments that are majorly involved in upstream phases undergo higher fluctuation in effort.
4. A high overlap ratio of upstream and downstream activities, combined with a high likelihood of unanticipated activity rework that requires additional resources will result in a strong tendency for NPD projects to behave in an unstable and unpredictable manner and lead to unforeseen departures from the predetermined baseline plan.
5. Adopting a more restrictive *RRS* (Convex–Up) leads to a longer NPD lead time and higher project cost. There is no obvious distinction between Stepped Linear and Linear *RRSs*. Also, the evaluation of *LCE* reduces the impacts of *RRS*.

6. When only the IEC process propagation among development activities is examined, high correlations between lead time, cost, and quality are observed. However, when the effects of IEC product propagation among dependent product components/systems, the correlation between lead time and project cost, and the one between lead time and quality drop significantly.
7. Batching of IECs possesses a competitive advantage in lead time over handling IECs individually. This superiority is the greatest when a sequential PD process is adopted, and reduces as overlapping ratio increases. However, there is neither IEC policy shows “dominant” advantage in project cost or quality.
8. Potential tradeoffs among NPD lead time and total cost are clearly identified when resource assignment decision is to be made. A higher level of *OS* leads to a shorter NPD lead time and less total cost given the same amount of functional resource allocation. However, the benefits of lead time reduction by assigning more resources are the most obvious in a sequential process, and activity overlap reduces the degree of obviousness the benefits have. The higher the *OS*, the less the benefits.
9. Linearity between lead time and quality is observed in all three *OS* levels: the higher the functional resource availability, the shorter the lead time, and the lower the quality. The linearity slope increases as the *OS* increases. The percentage of decrease in quality versus baseline case is the largest in a sequential process and decreases as *OS* increases.
10. The evaluation of IEC product propagation leads to a general increase of the multiple dimensions of NPD project performance from baseline case, except a counterintuitive decrease in NPD project lead time for a less coupled product configuration under a high environmental uncertainty and a high *RL*.

7.3 Limitations

There are many limitations this dissertation has faced that could potentially lead to some considerable impacts on its ability to effectively answer the research questions raised in *Chapter 1* and the quality of the findings listed in *Chapter 6*. By exploring the nature of these major limitations, suggestions of how such limitations could be overcome in future work will also be discussed.

1. **Limitations of Model Assumptions:** This research suffers from potential weaknesses because of the following simplified and restrictive model assumptions, which should be used with caution:

   1) Exponential relationship between solution uncertainty and rework probability,
   2) Add–ability of solution completeness for overlapped activities,
   3) Complete predictability of NPD activities and fixed activity precedence relationships,
   4) Mandatory Rework and Sequential Rework Process, and
   5) Static Rework Criteria.

2. **Lack of Flexibility in Model Extensions:** The proposed model is illustrated by an "abstractly simplified" numerical example of a three–phase and three–activity NPD process, and then is further expended into a two–level change propagation loop to use the full model capacity. Without any doubt, just the illustration presented in *Chapter 6* is far from enough in dealing with real world NPD and ECM issues of considerable size and complexity. Extensions of the rigid model structure, including i) *project size* (e.g., number of comprising phases and activities of a process, number of comprising systems, subsystems, and components of a product, etc.), ii) *concurrency of projects*, and iii)

precedence relationships among activities and couplings of product structure, require considerable additional modeling construction effort. These problems are rooted in the pitfalls specific to simulation studies due to the structural constraints of simulation models, especially for those that are built by off–the–shelf software packages. One of the most valuable explorations of the current work is to incorporate programming and scripting to make the model easier to build, edit and manage.

3. **Complexity and Difficulty in Model Parameterization**: This research aims at providing a model–based decision support tool to evaluate the mutual influence of NPD and ECM. In order to effectively implement and use this proposed tool, companies have to parameterize the simulation model using actual data under various levels of granularity to reflect their own NPD/ECM processes and project complexity. However, the acquisition of data within or across organization(s) relating to product, process, team, and environment information in order to appropriately parameterize the model is extremely challenging. Field observations (*Chapter 4*) have revealed at least three major obstacles toward model parameterization: i) *not enough data*, ii) *inaccurate or outdated* (practically meaningless) *data*, and iii) *data integration* due to the fact that different pieces of NPD and ECM data are usually collected and maintained by different departments or process participants from their own perspective in numerous *data formats* and various *granularity levels* (within a range from project level, cross functional team level, department level, to organizational level and /or inter–organizational network level). A key element of solving this problem and accomplishing large scale simulation is to achieve organizational level

of data acquisition, integration, and maintenance, and to link the simulation model with input data that are extracted automatically from corporate databases.

4. **Difficulty in Model Validation:** Lack of model validation via comparison to a corresponding real system is an important, obvious limitation of this research. Similar to the previously mentioned limitation in model parameterization, this one is also originated from the inability to attain extensive industrial data in author's individual capacity. However, this dissertation uses many other types of validation methods (e.g., construction of simulation model using validated model structures and features, close examination of model assumptions and parameter settings by NPD/ECM practitioners, qualitative comparison of model results to related NPD/ECM literature, published case studies and empirical research, and actual project performance observed in dissertation field survey studies).

5. **Descriptive Simulation Model to Support Decision Making**: It is important to note that the proposed model is only a descriptive simulation model instead of a prescriptive optimization one. It yields distributions of performance outputs (i.e., lead time, cost, and quality) when characteristics of product, process, team, and environment (e.g., overlapping ratio of process, IEC arrival patterns, cross functional integration, resource availability, etc.) are provided. It is not able to offer a set of characteristics to give the optimal development performance. However, as a what–if tool, it is competent to "*be foresight* (predicting how systems might behave in the future under assumed conditions) and *policy design* (designing new decision–making strategies and organizational structures and evaluating their effects on the behavior of the system)" (Sterman 1991).

Furthermore, this research work provides foundational findings identified by manipulating the descriptive components, on which a prescriptive optimization model can be built by specifying the objective function, the decision variables, and the constraints.

7.4 Future Work

Based upon the above discussions of essential research limitations, possible directions of future work can be summarized as follows:

1. In *SS 6.3.6*, only 33% *OS* is explored for preliminary conclusion statements of the effects of product configuration on IEC propagation, and project performance indicators accordingly. Other two process overlapping ratios should also be analyzed in depth to provide sufficient evidence and generate all of the conclusions that need to be drawn. Moreover, comprising items of a complex engineering product are typically interdependent. Model assumption of unidirectional IEC propagation path (i.e., IECs can only propagate from parent product item to child item, not the other way around) is a departure from reality and should be broadened in future work to capture those more complicated bidirectional change propagations.
2. As already mentioned in the limitations of model assumption, the following model features: i) different relationships between solution uncertainty and rework probability, ii) more detailed modeling of dynamic rework review criteria (in replace of the current static one), and iii) parallel rework policy should be tested to assess their impacts on project performance measures.

3. Our reading of the literature has indicated a lack of development process models that are capable to be extended and implemented into a multi–project environment while still keeping detailed aspects of project complexity. Building blocks of the model framework presented in this dissertation can be reconfigured and applied at various detail levels. From a single project level to the entire organizational level, it opens possibilities for further analyses of multi–project management, such as work force planning strategies, coordination policies of interdependent parallel projects, etc.
4. This model can also be further extended across organizations. By relaxing the single organization restriction of the current model and including inter–organizational influences, how engineering changes propagate along supply chain and affect NPD project performance can be explored.

BIBLIOGRAPHY

Aurich, Jan C., M. Rößing. 2007. Engineering Change Impact Analysis in Production Using VR. *Digital Enterprise Technology*, Springer US. SESSION1:75–82

Balakrishnan, N., A. K. Chakravarty. 1996. Managing Engineering Change: Market Opportunities and Manufacturing Costs. *Production and Operations Management*. 5 (4):335–356.

Balcerak, K. J., B. G. Dale. 1992. Engineering Change Administration: The Key Issues. *Computer–Integrated Manufacturing Systems*. 5 (2):125–132.

Barzizza, R., M. Caridi, R. Cigolini. 2001. Engineering Change: A Theoretical Assessment and A Case Study. *Production Planning & Control*. 12 (7):717–726.

Bhuiyan, N. 1998. Dynamic Models of Concurrent Engineering Processes and Performance. Doctoral thesis. McGrill University. Montreal, Canada.

Bhuiyan, N., D. Gerwin, V. Thomson. 2004. Simulation of the New Product Development Process for Performance Improvement. *Management Science*. 50(12):1690–1703.

Bhuiyan, N., G. Gregory, V. Thomson. 2006. Engineering Change Request Management in a New Product Development Process. European Journal of Innovation *Management*. 9(1):5–19.

Black, L. J., N. P. Repenning. 2001. Why Firefighting Is Never Enough: Preserving High–Quality Product Development. *System Dynamic Review*. 17(1):33–62.

Bouikni, N., A. Desrochers. 2006. A Product Feature Evolution Validation Model for Engineering Change Management. *Journal of Computing and Information Science in Engineering*. 6(2):188–195.

Braha, D., Y. Bar–Yam. 2007. The Statistical Mechanics of Complex Product Development: Empirical and Analytical Results. *Management Science*. 53(7):1127–1145.

Brown, S. L. 1995. Product Development: Past Research, Present Findings, and Future Directions. *Academy of Management Review.* 20(2): 343–378.

Browning, T. R. 1998. Modeling and Analyzing Cost, Schedule, and Performance in Complex System Product Development. Doctoral thesis. Massachusetts Institute of Technology. Cambridge, MA.

Browning, T. R., S. D. Eppinger. 2002. Modeling Impacts of Process Architecture on Cost and Schedule Risk in Product Development. *IEEE Transactions on Engineering Management.* 49(4): 428–442.

Browning, T. R., E. Fricke, H. Negele. 2006. Key Concepts in Modeling Product Development Processes. *System Engineering*. 9(2):104–128.

Browning, T. R., R. V. Ramasesh, 2007. A Survey of Activity Network–Based Process Models for Managing Product Development Projects. *Journal of Product Innovation Management*. 16(2):160–172.

Chen, Y. M., W. S., Shir, C. Y. Shen, 2002. Distributed Engineering Change Management for Allied Concurrent Engineering. *International Journal of Computer Integrated Manufacturing*. 15(2):127–151.

Cho, S. H., S. D. Eppinger, 2005. A Simulation–Based Process Model for Managing Complex Design Projects. *IEEE Transactions on Engineering Management*. 52(3):316–328.

Clark, K. B., T. Fujimoto. 1991. *Product Development Perfornmance: Strategy, Organization, and Management in the World Auto Industry*. Boston, Mass.: Harvard Business School Press.

Clarkson P. J., C. Simons, C. Eckert, 2001. Predicting Change Propagation in Complex Design. In *Proceedings of 13th International Conference on Design Theory and Methodology,* 9–12. Pittsburgh, Pennsylvania: American Society of Mechanical Engineers.

Clarkson P. J. and C. Eckert. 2004. Design and Process Improvement: A Review of Current Practice. Springer. 1st edition.

Diprima, M. 1982. Engineering Change Control and Implementation Considerations. *Production and Inventory Management.* 23: 81–87

Do, N., I. J. Choi, M. K. Jang. 2002. A Structure–Oriented Product Data Representation of Engineering Changes for Supporting Integrity Constraints. *The International Journal of Advanced Manufacturing Technology*. 20(8):564–570.

Do, N., I. J. Choi, M. Song. 2008. Propagation of Engineering Changes to Multiple Product Data Views Using History of Product Structure Changes. *International Journal of Computer Integrated Manufacturing*. 21(1):19–32.

Dragut, A. B., and J. W. M. Bertrand. 2008. A Representation Model for the Solving–Time. *European Journal of Operational Research.* 189(3):1217–1233.

Earl C., J. H. Johnson and C. Eckert. (2005). "Complexity" in *Design Process Improvement – A Review of Current Practice*. 174-196, Springer, ISBN 1-85233-701-X, 2005

Eckert, C., P. J. Clarkson, W. Zanker. 2004. Change and Customisation in Complex Engineering Domains. *Research in Engineering Design*. 15(1):1–21.

Ford, D. N. 1995. The Dynamics of Project Management: An Investigation of the Impacts of Project Process and Coordination on Performance. Doctoral thesis. Sloan School of Management. Massachusetts Institute of Technology. Cambridge, MA.

Ford, D. N., J. D. Sterman. 1998. Dynamic Modeling of Product Development Processes. *System Dynamics Review*. 14(1): 31–68.

Ford, D. N., J. D. Sterman. 2003. Overcoming the 90% Syndrome: Iteration Management in Concurrent Development Projects. *Concurrent Engineering: Research and Applications.* 11(3):177–186

Ford, D. N., J. D. Sterman. 2003. The Liar's Club: Concealing Rework in Concurrent Development. *Concurrent Engineering: Research and Applications.* 11(3):211–219

Ford, D. N., D. K. Sobek. 2005. Adapting Real Options to New Product Development by Modeling the Second Toyota Paradox. *IEEE Transactions on Engineering Management.* 52(2): 175–185

Ford Motor Company. 2003. Advanced Product Quality Planning (APQP) Status Reporting Guideline. Ford APQP Guideline. Release Level 3.2. http://www.takeuchi–iso.com/isots16949_2009/ts_apqp_doc/Ford_APQPGuideline.pdf

Ha, A. Y., E. L. Porteus. 1995. Optimal Timing of Reviews in Concurrent Design for Manufacturability. Management Science. 41(9):1431–1447.

Harhalakis, G. 1986. Engineering Changes for Made–to–order Products: How An MRP II System Should Handle Them. *Engineering Management International*. 4:19–36.

Hegde, G. G., Sham Kekre, Sunder Kekre. 1992. Engineering Changes and Time Delays: A Field Investigation. *International Journal of Production Economics*. 28(3):341–352.

Hillier, F. S., G. J. Lieberman. 2001. *Simulation Modeling & Analysis*. New York: McGraw–Hill, Inc. 4th edition.

Ho, C. J. 1994. Evaluating the Impact of Frequent Engineering Changes on MRP System Performance. *International Journal of Production Research*. 32(3):619–641.

Ho, C. J., J. Li. 1997. Progressive Engineering Changes in Multi–level Product Structures. *Omega: International Journal for Management Science*. 25(5):585–594.

Hobday, M. 1998. Product Complexity, Innovation, and Industrial Organization. *Research Policy*. 26(6):689–710.

Huang, G. Q., K. L. Mak. 1998. Computer Aids for Engineering Change Control. *Journal of Materials Processing Technology*. 76(1):187–191.

Huang, G. Q., K. L. Mak. 1999. Current Practices of Engineering Change Management in Hong Kong Manufacturing Industries. *Journal of Materials Processing Technology*. 19(1):21–37.

Huang, G. Q., W. Y. Yee, K. L. Mak. 2001. Development of a Web–based System for Engineering Change Management. *Robotics and Computer Integrated Manufacturing*. 17(3):255–267.

Huang, G. Q., W. Y. Yee, K. L. Mak. 2003. Current Practices of Engineering Change Management in UK Manufacturing Industries. *International Journal of Operations & Production Management*. 139(1–3):481–487.

Huge, E. C. 1977. Engineering Change Control. *International Journal of Operations & Production Management*. 139(1–3):481–487.

Keller, R., C. M. Eckert, P. J. Clarkson. 2005. Multiple Views to Support Engineering Change Management for Complex Products. In *Proceedings of the 3rd International Conference on Coordinated & Multiple Views in Exploratory Visualization*, 33–51.

Kelton, W. David, R. P. Sadowski, D. T. Sturrock. 2006. ***Simulation with Arena***, The McGraw–Hill Companies. London, 4th edition.

Klein, M., S. Poltrock, M. Handel. 2007. A Coordination–Theoretic Approach to Understanding Process Differences. *MIT Sloan School of Management Working Paper*. 4637–07.

Koh, E. CY., P. J. Clarkson. 2009. A Modelling Method to Manage Change Propagation. In *Proceedings of the 18th International Conference on Engineering Design.* Stanford, California.

Krishnan, V., S. D. Eppinger, D. E. Whitney. 1997. A Model–Based Framework to Overlap Product Development Activities. *Management Science*. 43(4):437–451.

Krishnan, V., K. T. Ulrich. 2001. Product Development Decisions : A Review of the Literature. *Management Science*. 47(1):1-21.

Law, Averill M. 2007. *Simulation Modeling & Analysis*. New York: McGraw–Hill, Inc. 4th edition.

Lee, H. J., H. J. Ahn, J. W. Kim 2006. Capturing and Reusing Knowledge in Engineering Change Management: A Case of Automobile Development. *Information Systems Frontiers*. 8(5):375–394.

Lévárdy, V. and T. R. Browning. 2009. An Adaptive Process Model to Support Product Development Project Management. *IEEE Transaction on Engineering Management.* 56(4): 600–620.

Lin, J., K. H. Chai, Y. S. Wong, A. C. Brombacher. 2007. A Dynamic Model for Managing Overlapped Iterative Product Development. *European Journal of Operational Research*. 185:378-392

Loch, C. H., C. Terwiesch. 1999. Accelerating the Process of Engineering Change Orders: Capacity and Congestion Effects. *Journal of Product Innovation Management*. 16(2):145–159.

Loch, C. H., J. Mihm, A. Huchzermeier. 2003. Concurrent Engineering and Design Oscillations in Complex Engineering Projects. *Concurrent Engineering: Research and Applications*. 11(3):187–199.

Lyneis, J. M., D. N. Ford. 2007. System Dynamics Applied to Project Management: a Survey, Assessment, and Directions for Future Research. *System Dynamics Review*. 23(2/3): 157–189.

Maull, R., D. Hughes, J. Bennett. 1992. The Role of Bill–of–materials as a CAD/CAPM Interface and the Key Importance of Engineering Change Control. *Computing & Control Engineering Journal*. 3(2):63–70.

Moon, Y. B. 2007, Enterprise Resource Planning (ERP): a review of the literature. *International Journal of Management and Enterprise Development*. 4(3): 235–264.

Morgan, J. M, J. K. Liker. 2006. *The Toyota Product Development System – Integrating People, Process, and Technology*. Productivity Press.

Park, M., F. Peña–Mora. 2003. Dynamic Change Management for Construction: Introducing the Change Cycle into Model-Based Project Management. *System Dynamics Review*. 19(3):213-242.

Peng, T. K., Amy J. C. Trappey. 1998, A Step Toward STEP–compatible Engineering Data Management: the Data Models of Product Structure and Engineering Changes. *Robotics and Computer–Integrated Manufacturing*. 14(2): 89–109.

Pimmler, T. U., and S. D. Eppinger. 1994. “Integration Analysis of Product Decompositions.” In *Proceedings of the ASME Design Theory and Methodology Conference*, 343–351. Minneapolis, Minnesota: American Society of Mechanical Engineers.

Reddi, R. K., and Y. Moon. 2009. A Framework for Managing Engineering Change Propagation. *International Journal of Innovation and Learning*. 6(5): 461–476.

Reichelt, K., J. Lyneis. 1999. The Dynamics of Project Performance: Benchmarking the Drivers of Cost and Schedule Overrun. *European Management Journal*. 17(2):135-150.

Reidelbach, M. A. 1991, Engineering Change Management for Long–lead–time Production Environments. *Production and Inventory Management Journal*. 32(2): 84–88.

Roemer, T. A., R. Ahmadi. 2004. Concurrent Crashing and Overlapping in Product Development. *Opertaion Research*. 52(4): 606–622.

Rouibah, K., K. R. Caskey. 2003, Change Management in Concurrent Engineering from a Parameter Perspective. *Computers in Industry*. 50(1): 15–34.

Rutka, A. et al. 2006. Methods for Engineering Change Propagation Analysis. In Proceedings of 25th International Congress of the Aeronautical Sciences. Hamburg, Germany.

Saeed, B. I., D. M. Bowen, V. S. Sohoni. 1993. Avoiding Engineering Changes Through Focused Manufacturing Knowledge. *IEEE Transactions on Engineering Management*. 40(1):54–59.

Scholz–Reiter, B., F. Krohne, B. Leng, H. Hohns. 2007. Technical Product Change Teams: An Organizational Concept for Increasing the Efficiency and Effectiveness of Technical Product Changes during Ramp–up Phases. *International Journal of Production Research*. 45(7):1631–1642.

Smith, R. P., S. D. Eppinger. 1997. A Predictive Model of Sequential Iteration in Engineering Design. *Management Science*. 43(8):1104–1120.

Sweetser, A. 1999. A Comparison of System Dynamics and Discrete Event Simulation. In Proceedings of 17th International Conference of the System Dynamics Society and 5th Australian & New Zealand Systems Conference. Wellington, New Zealand.

Tavcar, J., J. Duhovnik. 2005. Engineering Change Management in Individual and Mass Production. *Robotics and Computer–Integrated Manufacturing*. 21(3):205–215.

Tako, A. A., S. Robinson. 2009. Comparing Discrete–Event Simulation and System Dynamics: Users' Perceptions. *Journal of the Operational Research Society*. 60: 296–312.

Terwiesch, C., C. H. Loch. 1999. Managing the Process of Engineering Change Orders: The Case of the Climate Control System in Automobile Development. *Journal of Product Innovation Management*. 16(2):160–172.

Thomas, H. R., C. L. Napolitan. 1994. The Effects of Changes on Labor Productivity: Why and How Much. CII Document 99. Construction Industry Institute. Austin, TX.

Unger, D. W., S. D. Eppinger. 2009. Comparing Product Development Processes and Managing Risk. *International Journal of Product Development*. 8(4):382–401.

W□nstr□m, C., P. Jonsson. 2006. The Impact of Engineering Changes on Materials Planning. *Journal of Manufacturing Technology Management*. 17(5):561–584.

Watts, F. 1984. Engineering Changes: A Case Study. *Production and Inventory Management Journal*. 25(4):55–62.

Wright, I. C. 1997. A Review of Research into Engineering Change Management: Implications for Product Design. *Design Studies*. 18(1):33–42.

Wynn, D. C., C. M. Eckert, P. J. Clarkson. 2007. Modelling Iteration in Engieering Design. In *Proceedings of the 17th International Conference on Engineering Design.* Paris, France.

Wynn, D. C., K. Grebici, P. J. Clarkson. 2011. Modelling the Evolution of Uncertainty Levels during Design. *International Journal on Interactive Design and Manufacturing.* 5:187–202.

APPENDICES

APPENDIX A:

Questionnaire of Change Request Approval Process

Interviewees (Number of People Interviewed):

Project Managers (1); Quality Assurance Leader (2); Product Manager (2); Release Manager (2); Demand Administrator (1)

Interview Question:

1) How well do you think the Change Request process flow chart describes the nature of how a CR is being handled?

____

2) Could you briefly describe your approval process of a CR in terms of:
    a. How long does it usually take you to approve/disapprove a CR? Please provide both actual length and calendar length.

    ____

    b. Do you make decision based on any reports/metrics/models from other people?

    ____

    c. What are the commonly experienced road–blocks that keep you from decision making?

    ____

3) Could you briefly explain the causes behind an extremely long approval process (with extreme cases prepared according to different participants)?

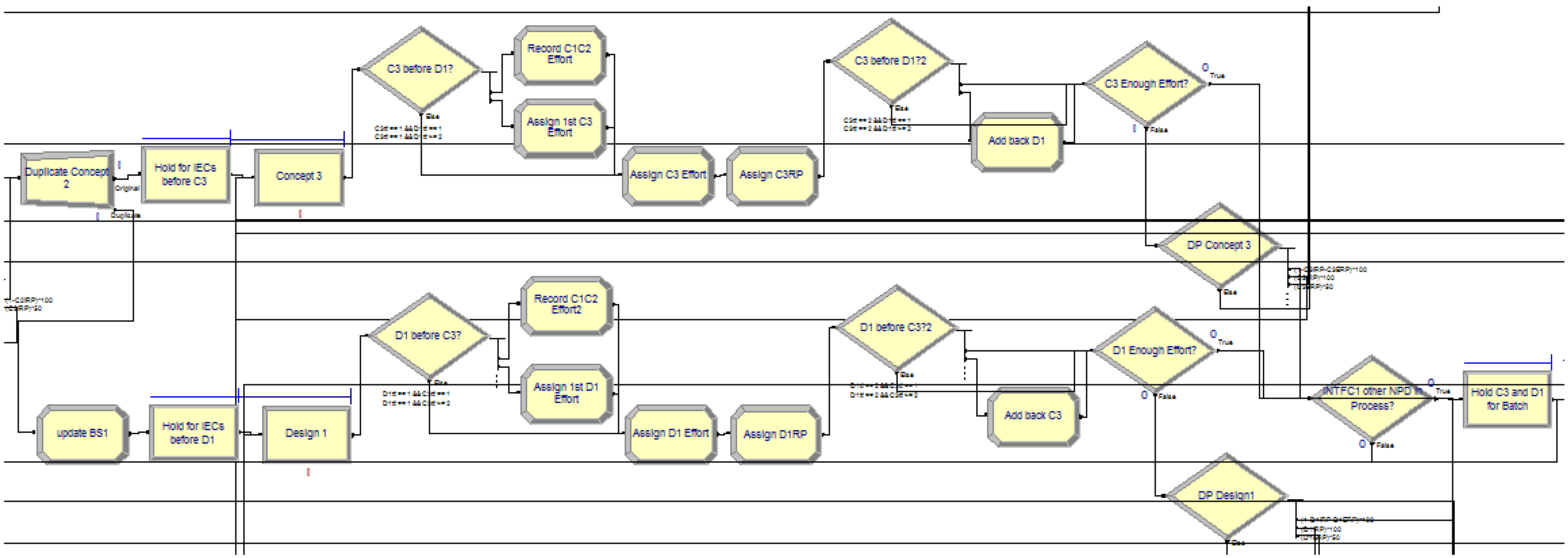
____

4) What do you think can be done to improve the efficiency and productivity of the CRM process?

____

APPENDIX B:

Arena Flow Process of Overlapped Upstream and Downstream Activities



APPENDIX C:

Rework Criteria

		R–C1	R–C2	R–C3	R–D1	R–D2	R–D3	R–P1	R–P2	R–P3
RRS1 *Stepped Linear*	***RC ratio***	***0.4***	***0.45***	***0.5***	***0.6***	***0.65***	***0.7***	***0.8***	***0.85***	***0.9***
	Mkt	1344	1512	1680	2016	2184	2352	2688	2856	3024
	Eng	1536	1728	1920	2304	2496	2688	3072	3264	3456
	Mfg	1920	2160	2400	2880	3120	3360	3840	4080	4320
RRS2 *Linear*	***RC ratio***	***0.4***	***0.4625***	***0.525***	***0.5875***	***0.65***	***0.7125***	***0.775***	***0.8375***	***0.9***
	Mkt	1344	1554	1764	1974	2184	2394	2604	2814	3024
	Eng	1536	1776	2016	2256	2496	2736	2976	3216	3456
	Mfg	1920	2220	2520	2820	3120	3420	3720	4020	4320
RRS3 *Convex–Up*	***RC ratio***	***0.4***	***0.5***	***0.6***	***0.675***	***0.75***	***0.8***	***0.85***	***0.875***	***0.9***
	Mkt	1344	1680	2016	2268	2520	2688	2856	2940	3024
	Eng	1536	1920	2304	2592	2880	3072	3264	3360	3456
	Mfg	1920	2400	2880	3240	3600	3840	4080	4200	4320
RRS4 *Concave–UP*	***RC ratio***	***0.4***	***0.425***	***0.45***	***0.5***	***0.55***	***0.625***	***0.7***	***0.8***	***0.9***
	Mkt	1344	1428	1512	1680	1848	2100	2352	2688	3024
	Eng	1536	1632	1728	1920	2112	2400	2688	3072	3456
	Mfg	1920	2040	2160	2400	2640	3000	3360	3840	4320

APPENDIX D:

Possible Starting Point of Iteration/EEC Loop

	R–C1	**R–C2**	**R–C3**	**R–D1**	**R–D2**	**R–D3**	**R–P1**	**R–P2**	**R–P3**
0%									
Iteration Loop	C1	C2; C1	C3; C2; C1	D1	D2; D1	D3; D2; D1	P1	P2; P1	P3; P2; P1
EEC Loop				C3; C2; C1	C3;C2; C1	C3;C2; C1	D3; D2; D1; C3;C2; C1	D3; D2; D1; C3;C2; C1	D3; D2; D1; C3;C2; C1
33%									
Iteration Loop	C1	C2; C1	C3	D1	D2; D1	D3	P1	P2; P1	P3; P2; P1
EEC Loop			C2; C1	C2; C1	C3;C2; C1	D2; D1; C3;C2; C1	D2; D1; C3;C2; C1	D3; D2; D1; C3;C2; C1	D3; D2; D1; C3;C2; C1
66%									
Iteration Loop	C1	C2	C3	D1	D2	D3; P1	P1	P2; P1	P3; P2; P1
EEC Loop		C1	D1; C2; C1	C1	D1; C2; C1	D2; D1; C3;C2; C1	D1; C2; C1	D2; D1; C3; C2; C1	D3; D2; D1; C3;C2; C1

APPENDIX E:

Arena Module and Expression Model Variables

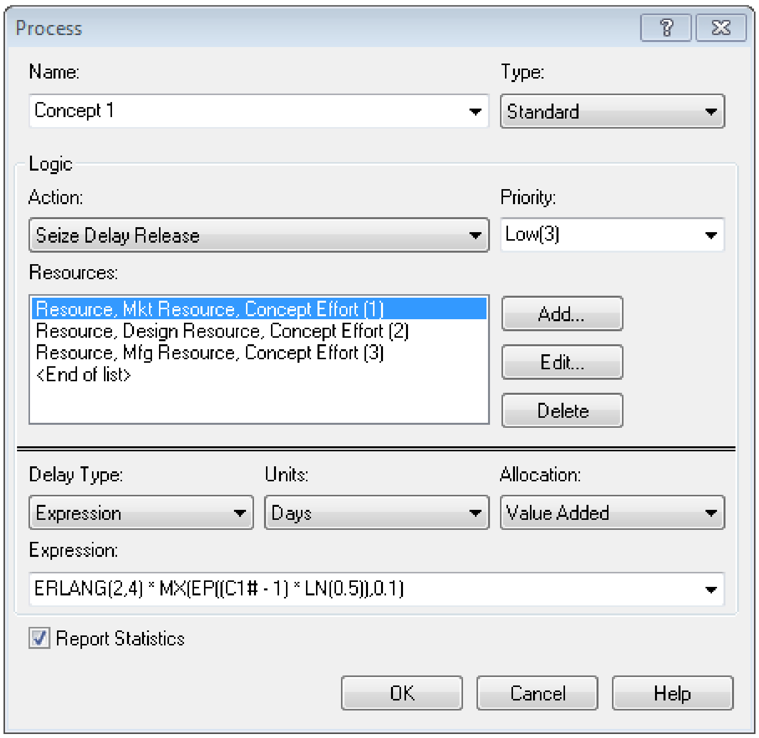
PROCESS MODULE

Process Name:

Concept 1, 2, 3 ; Design 1, 2, 3 ; Production 1, 2, 3

Delay Expression:

ERLANG(2,4) * MX(EP((Xn# – 1) * LN(0.5)),0.1); *(X = C, D, P ; n = 1, 2, 3)*

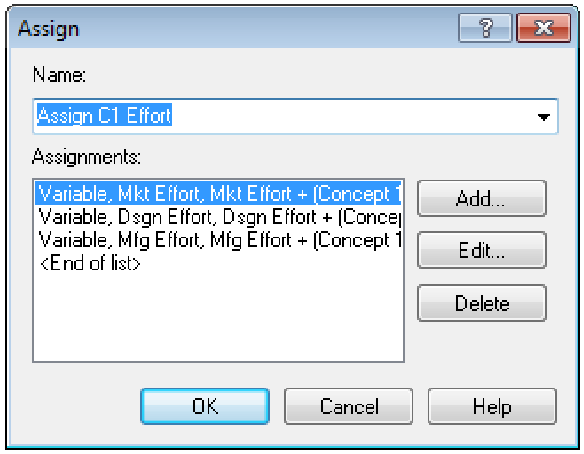


ASIGN MODULE (NPD Section)

Mkt Effort = Mkt Effort + (“NPD ACTIVITY”.VATime – “NPD ACTIVITY” Total Time) * “NPD PHASE” Effort (1)

Dsgn Effort = Dsgn Effort + (“NPD ACTIVITY”.VATime – “NPD ACTIVITY” Total Time) * “NPD PHASE” Effort (2)

Mfg Effort = Mfg Effort + (“NPD ACTIVITY”.VATime – “NPD ACTIVITY” Total Time) * “NPD PHASE” Effort (3)



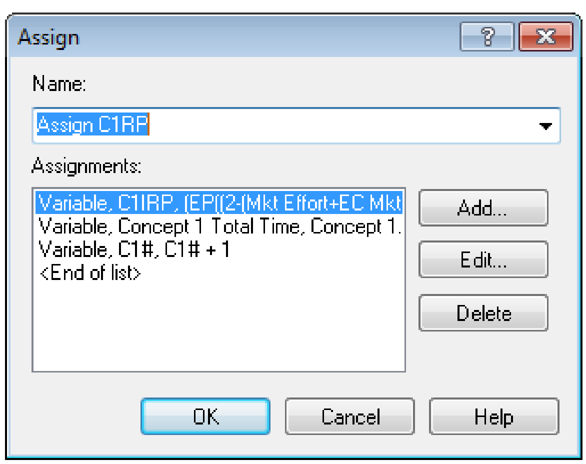
CnIRP = (EP((2 – (Mkt Effort + IEC Mkt Effort)/(Concept Scope + IEC Mkt Effort))*LN(alpha))*60+EP((2 – (Dsgn Effort + IEC Dsgn Effort)/(Design Scope + IEC Dsgn Effort))*LN(alpha))*20 + EP((2 – (Mfg Effort + IEC Mfg Effort)/(Production Scope + IEC Mfg Effort))*LN(alpha))*20)/100 ; *(n = 1, 2, 3)*

DnIRP = (EP((2 – (Mkt Effort + IEC Mkt Effort)/(Concept Scope + IEC Mkt Effort))*LN(alpha))*20+EP((2 – (Dsgn Effort + IEC Dsgn Effort)/(Design Scope + IEC Dsgn Effort))*LN(alpha))*60 + EP((2 – (Mfg Effort + IEC Mfg Effort)/(Production Scope + IEC Mfg Effort))*LN(alpha))*20)/100 ; *(n = 1, 2, 3)*

PnIRP = (EP((2 – (Mkt Effort + IEC Mkt Effort)/(Concept Scope + IEC Mkt Effort))*LN(alpha))*20+EP((2 – (Dsgn Effort + IEC Dsgn Effort)/(Design Scope + IEC Dsgn Effort))*LN(alpha))*20 + EP((2 – (Mfg Effort + IEC Mfg Effort)/(Production Scope + IEC Mfg Effort))*LN(alpha))*60)/100 ; *(n = 1, 2, 3)*

"NPD ACTIVITY" Total Time = "NPD ACTIVITY".VATime

Xn# = Xn# + 1; *(X = C, D, P ; n = 1, 2, 3)*



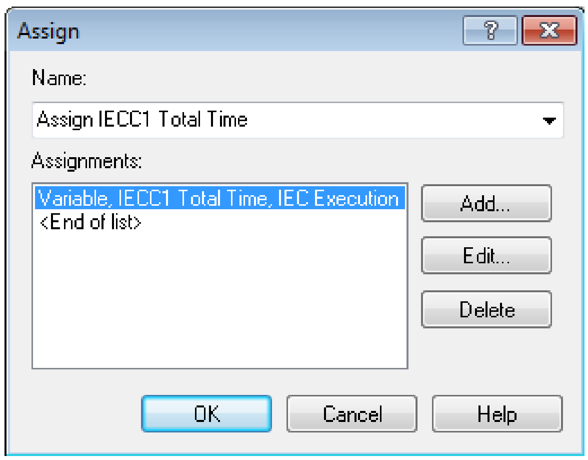
ASIGN MODULE (IEC Section)

IEC Mkt Effort = IEC Mkt Effort + (“IEC ACTIVITY”.VATime – IEC ACTIVITY Total Time) * IEC Effort (1)

IEC Dsgn Effort = IEC Dsgn Effort + (“IEC ACTIVITY”.VATime – IEC ACTIVITY Total Time) * IEC Effort (2)

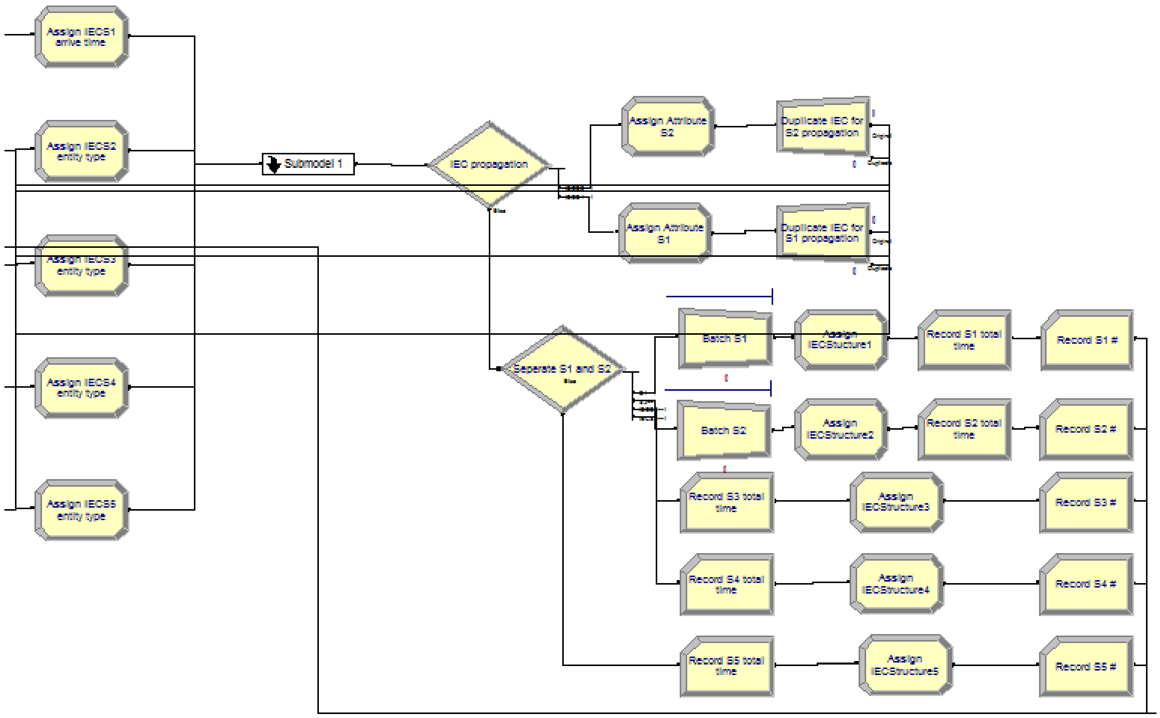
IEC Mkt Effort = IEC Mkt Effort + (“IEC ACTIVITY”.VATime – IEC ACTIVITY Total Time) * IEC Effort (3)

“IEC ACTIVITY” Total Time = “IEC ACTIVITY”.VATime



APPENDIX F:

Arena Flow Process of Change Propagation almong Product Items



VITA

Personal Biographical Data:

Birthplace and date: Nanjing, China
December 22, 1984

Education: B.S., Shanghai Jiao Tong University, 2007

Fields of Interest:

- Modeling and simulation of complex product design and development process,
- Engineering change management, and
- Process improvement to product realization

List of Published Papers:

W. Li, Y.B. Moon. "Modeling and managing engineering changes in a complex product development process", International Journal of Advanced Manufacturing Technology, in press, DOI: 10.1007/s00170-012-3974-x.

K.R. Reddi, **W. Li**, B. Wang, Y.B. Moon. "System dynamics modeling of hybrid renewable energy systems and combined heating & power generator", International Journal of Sustainable Engineering, accepted.

Proceedings Published:

W. Li, Y.B. Moon. "Modeling and managing engineering changes in a complex product development process", in Proceedings of 2011 Winter Simulation Conference, Phoenix , AZ, Dec. 11-14, 2011.

W. Li. "A simulation model for managing engineering changes by a multi-method modeling approach of discrete-event simulation and agent-based simulation" 2010 Winter Simulation Conference, Baltimore, MD, Dec. 5-8, 2010.

W. Li, Y.B. Moon, “A simulation model for managing engineering change along with new product development”, in Proceedings of 11th International Conference on Enterprise Information Systems, Milan, Italy, May 6-10, 2009.

Book Chapter:

W. Li. “Modeling engineering changes using discrete event simulation”, Discrete Event Simulations, ISBN 979-953-307-806-8, publication July 2012.

Awards:

University Fellowships, Syracuse University (Academic Year 2007/08, 2009/10)

Teaching Assistantships, Syracuse University (2008/09, 2010/11, Fall 2011)

Research Assistantship, Syracuse University (Spring 2012)

Nunan Graduate Travel Fund to present work at conferences (Dec 2009, Dec 2010)

The Honor Society of Phi Kappa Phi (2008–present)

Excellent Academic Scholarship, Shanghai Jiao Tong University (2005/06, 2006/07)