A Theory of Information Quality and a Framework for its Implementation in the Requirements Engineering Process

by Michael W. Grenn

B.S. in Biology, May 1982, The College of William and Mary
M.S. in Systems Engineering, May 2009, The George Washington University

A Dissertation submitted to

The Faculty of
The School of Engineering and Applied Science
of The George Washington University
in partial fulfillment of the requirements
for the degree of Doctor of Science

January 31, 2014

Dissertation directed by

Shahram Sarkani
Professor of Engineering Management and Systems Engineering

Thomas A. Mazzuchi
Professor of Engineering Management and Systems Engineering & of Decision Sciences

The School of Engineering and Applied Science of The George Washington University certifies that Michael W. Grenn has passed the Final Examination for the degree of Doctor of Science as of September 18, 2013. This is the final and approved form of the dissertation.

A Theory of Information Quality and a Framework for its Implementation in the Requirements Engineering Process

Michael W. Grenn

Dissertation Research Committee:

Thomas A. Mazzuchi, Professor of Engineering Management and Systems Engineering & of Decision Sciences, Dissertation Co-Director

Shahram Sarkani, Professor of Engineering Management and Systems Engineering, Dissertation Co-Director

Pavel Fomin, Associate Professorial Lecturer of Engineering Management and Systems Engineering, Committee Member

E. Lile Murphree, Jr., Professor Emeritus of Engineering Management and Systems Engineering, Committee Member

James Wasek, Associate Professorial Lecturer of Engineering Management and Systems Engineering, Committee Member

© Copyright 2013 by Michael W. Grenn
All rights reserved

Dedication

This dissertation is dedicated to my father Carl and my mother Mary, who are the most loving and supportive parents a son could ever have, my wonderful wife Molly whose strength and encouragement sustains our family, my daughter Gracie who is my joy and inspiration, and my Lord and Savior Jesus Christ without whom I would be lost.

Acknowledgements

I would like to thank my professors and instructors who helped lay the foundation for this work, particularly Dr. Gulu Gambhir for sharing his knowledge of systems engineering and requirements engineering, Dr. Christopher Willy for his rigorous review of probability and stochastic processes, Dr. William J.J. Roberts for his invaluable course in discrete systems simulations which enabled me to implement the Requirements Entropy Framework (REF) in MATLAB® and perform the simulation experiments, Dr. Salim Zafar for his outstanding course in multivariate statistics, Dr. Enrique Campos-Nanez for demonstrating the importance and practical utility of system dynamics modeling, Dr. Thomas Holzer for his emphasis on systems thinking in his systems analysis course, Dr. James Wasek who brought program management and systems engineering concepts to life through his practical experience, Dr. Marvine Hamner whose course in R&D management motivated me to think about applying systems engineering principles early in the development cycle, Dr. Arturo Balana for sharing his expertise in linear and nonlinear programming, Dr. David Rico whose course in research methods helped prepare me for this work, Dr. Andrew McDonald for his insight into decision analysis and utility theory and for his words of encouragement to pursue my doctorate, and Dr. Tim Eveleigh for sharing his experience and perspective on requirements engineering. I thank Dr. E. Lile Murphree for his thoughtful comments and suggestions during my dissertation defense.

I would also like to thank Mr. Tom Devendorf for his encouragement of my academic pursuits over the past several years. I would have been hard-pressed to finish this work

without his support and willingness to accommodate my schedule. Not many Directors would have been as understanding as Tom and for this I am grateful.

A special thank you goes to my parents, family and friends for your patience with me over the past several years. I missed countless opportunities to enjoy your company and I look forward to seeing you more often in the future. I also thank my parents for my role models. I would not have earned my doctorate without your encouragement and help.

I would like to thank Randy Brown of Haynes and Boone LLP for his assistance with filing my patent application for the REF. It was comforting to know that the legal details were covered as I focused my attention on the journal article and dissertation.

It is a pleasure to acknowledge my classmates in the PhD cohort, several of whom have become good friends. I wish you all the very best in your future pursuits. It is also a pleasure to acknowledge my friends and colleagues in the Department of Defense who work diligently every day to develop and protect our most critical technology. It has been an honor and privilege to serve with those who put our men and women in uniform and our national security interests ahead of financial or personal gain.

Finally, I would like to acknowledge my advisors, Dr. Shahram Sarkani and Dr. Thomas Mazzuchi, for your guidance, suggestions and patience as I completed the requirements of the doctoral program. Your support made the arduous task of formulating, implementing and evaluating this new theory bearable and your patience allowed me to explore the many paths that led to this work. I would also like to express my gratitude for the opportunity to teach in the off-campus EMSE program. I have thoroughly enjoyed my teaching experience and consider it a privilege to contribute to the program from which I have received so much benefit. Thank you.

Abstract

A Theory of Information Quality and a Framework for its Implementation in the Requirements Engineering Process

This dissertation introduces a theory of information quality to explain macroscopic behavior observed in the systems engineering process. The theory extends principles of Shannon's mathematical theory of communication [1948] and statistical mechanics to information development processes concerned with the flow, transformation, and meaning of information. The meaning of requirements information in the systems engineering context is estimated or measured in terms of the cumulative requirements quality Q which corresponds to the distribution of the requirements among the available quality levels. The requirements entropy framework (REF) implements the theory to address the requirements engineering problem. The REF defines the relationship between requirements changes, requirements volatility, requirements quality, requirements entropy and uncertainty, and engineering effort. The REF is evaluated via simulation experiments to assess its practical utility as a new method for measuring, monitoring and predicting requirements trends and engineering effort at any given time in the process. The REF treats the requirements engineering process as an open system in which the requirements are discrete information entities that transition from initial states of high entropy, disorder and uncertainty toward the desired state of minimum entropy as engineering effort is input and requirements increase in quality. The distribution of the total number of requirements R among the N discrete quality levels is determined by the number of defined quality attributes accumulated by R at any given time. Quantum statistics are used to estimate the number of possibilities P for arranging R among the

available quality levels. The requirements entropy H_R is estimated using R, N and P by extending principles of information theory and statistical mechanics to the requirements engineering process. The information I increases as H_R and uncertainty decrease, and the change in information ΔI needed to reach the desired state of quality is estimated from the perspective of the receiver. The H_R may increase, decrease or remain steady depending on the degree to which additions, deletions and revisions impact the distribution of R among the quality levels. Current requirements trend metrics generally treat additions, deletions and revisions the same and simply measure the quantity of these changes over time. The REF evaluates the quantity of requirements changes over time, distinguishes between their positive and negative effects by calculating their impact on H_R, Q, and ΔI, and forecasts when the desired state will be reached, enabling more accurate assessment of the status and progress of the requirements engineering effort. Results from random variable simulations suggest the REF is an improved leading indicator of requirements trends that can be readily combined with current methods. The increase in I, or decrease in H_R and uncertainty, is proportional to the engineering effort E input into the requirements engineering process. The REF estimates the ΔE needed to transition R from their current state of quality to the desired end state or some other interim state of interest. Simulation results are compared with measured engineering effort data for Department of Defense programs published in the SE literature, and the results suggest the REF is a promising new method for estimation of ΔE.

Table of Contents

Dedication iv

Acknowledgements v

Abstract vii

Table of Contents ix

List of Figures xii

List of Tables xiv

List of Acronyms xv

List of Mathematical Symbols xviii

Chapter 1: Introduction 1

1.1: The Origins of Modern Systems Engineering (SE) 1

1.2: Research Problem Situation 3

1.3: Statement of Need 5

1.4: The SE Process 7

1.5: Research Significance 10

1.6: Dissertation Overview 12

Chapter 2: Literature Review 13

2.1: System Complexity 13

2.2: Requirements Engineering 17

2.2.1: Requirements Trends and Metrics 18

2.2.2: Requirements Quality 24

2.2.3: Engineering Effort 26

2.3: Quality Function Deployment 27

2.4: Relationship between Entropy, Information and Energy 30
Chapter 3: Systems Thinking 31
3.1: The Importance of Systems Thinking in SE 31
3.2: The SE Process as an Open System 32
3.3: Partial Similarities between SE and an Open Physical System 36
3.4: Partial Similarities between SE and Information Theory 42
Chapter 4: The Theory of Information Quality 48
4.1: Axioms of the Theory of Information Quality 48
4.2: Summary of Mathematical Foundation of the Theory 52
Chapter 5: The Requirements Entropy Framework (REF) 53
5.1: The Requirements Quality Attributes 53
5.2: Theoretical Construct 55
5.3: Mathematical Foundation 60
5.4: Theoretical Requirements Entropy Behavior 64
5.5: Examples of Requirements Quality Distributions 66
5.6: REF Model Context Diagram 70
5.7: Key Assumptions of MATLAB® Implementation of the REF 71
Chapter 6: Research Approach 77
6.1: Research Objectives and Methods 77
6.2: Important Factors in the REF Implementation of the Theory 80
6.3: Research Design 82
6.3.1: Research Premise 82
6.3.2: Research Questions and Hypotheses 83

6.3.3: Requirements Trend Factor Relationships 84
6.3.4: Requirements Engineering Effort Factor Relationships 85
6.4: Data Collection Approach 86
Chapter 7: Simulation Experiments 88
7.1: Requirements Trends 88
7.1.1: Experimental Results 90
7.1.2: Data Analysis 95
7.1.2.1: Analysis of Forecasts and Strength of Correlations 97
7.2: Requirements Engineering Effort 99
7.2.1: Experimental Results 100
7.2.2: Requirements Engineering Effort Data Analysis 107
7.2.2.1: Statistical Analysis of REF Results and Measured Data 107
Chapter 8: Conclusions and Future Research 108
8.1: Summary of Research Findings 112
8.2: Areas of Future Research 115
Appendices 117
Appendix A: Regression Analysis Plots of Engineering Effort vs Requirements 117
Appendix B: Null Hypothesis Individual Value Plots with Confidence Intervals 120
Appendix C: Regression Analysis Plots of Fractional Method vs REF Entropy 123
Appendix D: Summary of REF and Fractional Method Simulation Forecasts 126
Appendix E: Summary of REF Engineering Effort Simulation Data 127
References 128

List of Figures

Figure 1.1: DoD System Development Lifecycle with Emphasis on Early SE 5

Figure 1.2: The SE Process in Relation to SE Activities 8

Figure 2.1: Fractional Method Requirements Trend [Rhodes et al, 2009] 22

Figure 2.2: Fractional Method Requirements Trend [Costello and Liu, 1995] 23

Figure 2.3: Probability Plot of Honour [2004] 27

Figure 3.1: Representation of Particle Energies in One-Dimensional Energy Space 38

Figure 3.2: Possible Arrangements of 2 Particles in 3 Available States 39

Figure 3.3: Similarities Between the SE process and an Open Physical System 40

Figure 3.4: The Requirements Engineering Problem in Context of Shannon's Theory .. 43

Figure 3.5: The Shannon information entropy 47

Figure 5.1: The Requirements Entropy Framework (REF) 56

Figure 5.2: Example of Theoretical Requirements Entropy 65

Figure 5.3: Example of Theoretical Requirements Entropy versus Time 66

Figure 5.4: Examples of Requirements Quality Distributions 67

Figure 5.5: Examples of Lower and Higher Requirements Reuse Distributions 69

Figure 5.6: Context Diagram of the REF Modeling Process 70

Figure 6.1: Summary of Research Approach 78

Figure 6.2: Top Level Model of Important Research Factors 80

Figure 6.3: Model of Important Relationships in REF Implementation of Theory 81

Figure 6.4: Model of Important Requirements Trend Relationships 85

Figure 6.5: Model of Important Engineering Effort Relationships 86

Figure 7.1: Requirements Trend Simulation Run4 91

Figure 7.2: Requirements Trend Simulation Run10 92

Figure 7.3: Requirements Trend Simulation Run11 92

Figure 7.4: Requirements Trend Simulation Run2 93

Figure 7.5: Requirements Trend Simulation Run5 93

Figure 7.6: Requirements Trend Simulation Run12 94

Figure 7.7: Comparison of Simulation Run3 Data with Uniform Transition Data 94

Figure 7.8: Example Regression Analysis of Fractional Method vs REF 98

Figure 7.9: Example Regression Analysis of Engineering Effort vs R 100

Figure 7.10: Requirements Engineering Effort Simulation Run1 104

Figure 7.11: Requirements Engineering Effort Simulation Run2 104

Figure 7.12: Requirements Engineering Effort Simulation Run3 105

Figure 7.13: Requirements Engineering Effort Simulation Run4 105

Figure 7.14: Requirements Engineering Effort Simulation Run5 106

Figure 7.15: Lower Bound of Requirements Engineering Effort Data..... 107

List of Tables

Table 4.1: Axiomatic Summary of the Theory of Information Quality..... 51

Table 4.2: Summary of Mathematical Foundation of the Theory 52

Table 5.1: REF Quality Attributes and Definitions 54

List of Acronyms

AFCSE	Air Force Center for Systems Engineering
ASR	Alternative System Review
BE	Bose-Einstein
BMI	Baseline Measurement Interval
C^3	Command Control and Communication
CBA	Capability Based Assessment
CDD	Capability Development Document
CDR	Critical Design Review
CJCSM	Chairman of the Joint Chiefs of Staff Manual
COCOMO	Constructive Cost Model
COSYSMO	Constructive Systems Engineering Cost Model
CPD	Capability Production Document
DAG	Defense Acquisition Guidebook
DAU	Defense Acquisition University
DoD	Department of Defense
DODI	Department of Defense Instruction
EOL	End of Life
FM	Fractional Method
FRP	Full Rate Production
GAO	Government Accounting Office
ICD	Initial Capabilities Document
IEC	International Electrotechnical Committee

INCOSE	International Council On Systems Engineering
IOC	Initial Operating Capability
ISO	International Standards Organization
JCIDS	Joint Capabilities Integration and Development System
LCC	Life Cycle Cost
MDAP	Major Defense Acquisition Program
MS A	Milestone A
MS B	Milestone B
MS C	Milestone C
MSA	Materiel Solution Analysis
NRC	National Research Council
PDR	Preliminary Design Review
QFD	Quality Function Deployment
QRF	Quantum Requirements Framework
R&D	Research and Development
RDEC	Research, Development and Engineering Center
RE	Requirements Engineering
REF	Requirements Entropy Framework
RV	Requirements Volatility
SE	Systems Engineering
SOP	Standard Operating Procedure
SRD	System Requirements Document
SRR	Systems Requirements Review

TBD	To Be Determined
TBR	To Be Resolved
U.S.	United States
V&V	Verification and Validation

List of Mathematical Symbols

$\langle b \rangle$	Average information output bitrate for ΔI
$\langle \Delta E \rangle$	Average change in engineering effort
e	Base of the natural logarithm, approximately 2.71828
H	Shannon's information entropy
H_R	Requirements entropy
$H_{R\max}$	Maximum requirements entropy
$H_{R\min}$	Minimum requirements entropy
k	Boltzmann's constant
ln	Natural log
N	Number of discrete quality levels
$N_{\max}$	Maximum number of quality levels available to occupy
$N_{\min}$	Minimum number of quality levels available to occupy
P	Number of possible distributions available to R
$p(r_a)$	Probability of a requirement being added
$p(r_d)$	Probability of a requirement being deleted
$p(r_{new})$	Probability of a requirement being a new requirement
$p(r_{new}\text{quality})$	Probability of a new requirement having a quality attribute
$p(r_r)$	Probability of a requirement being revised
$p(r_{reuse})$	Probability of a requirement being a reused requirement
$p(r_{reuse}\text{quality})$	Probability of a reused requirement having a quality attribute
P_{BE}	Number of possibilities using Bose-Einstein statistics
P_{MB}	Number of possibilities using Maxwell-Boltzmann statistics

P_{SA}	Stirling's approximation of the Bose-Einstein distribution
Q	Cumulative requirements quality
q_n	Any given quality level
R	Number of requirements
r_a	Number of requirements added
r_d	Number of requirements deleted
r_j	Any given requirement in the set of R
r_n	Number of requirements occupying a given quality level
r_{new}	Number of new requirements
r_r	Number of requirements revised
r_{reuse}	Number of reused requirements
RV	Requirements volatility
S	Physical entropy
W	Number of microstates
z	Test statistic for normal distribution with $n \geq 30$
ΔI	Additional information corresponding to entropy decrease
σ	Standard deviation
$\forall$	For any
$\in$	An element of the set
Σ	Summation
!	Factorial
$\rightarrow$	Maps the elements to a given quality level

Chapter 1: Introduction

1.1: The Origin of Modern Systems Engineering

The origin of modern systems engineering (SE) has been traced to systems thinkers who sought new methods to resolve problems encountered during the development of sophisticated military systems prior to World War II [Brill, 1998:260; INCOSE, 2011:8]. These systems generally fell into the category of detail complexity where the system functions and architectures could be accurately represented by functional hierarchies with clearly defined partitions, localized relationships and predictable behavior [Senge, 1990; Calvano and John, 2004]. Although the number of possible system design alternatives was astronomical, the relationships and possible interactions between their constituents were well understood. Requirements engineering for such systems was relatively straightforward. The strengths and weaknesses of a given design option could be analyzed and its impact on system performance modeled to support the selection of a near optimal solution. System behavior was predictable because the complexity of these systems was not beyond the technical capacity of the SE team.

SE has matured into a formal discipline to specifically address the technical challenges associated with highly complex systems [Simpson and Simpson, 2010:1]. Architectures of advanced systems today are software-intensive and highly integrated with increasingly lateral, microscopic and complex interactions between their parts [Calvano and John, 2004]. These complex systems often exhibit dynamic complexity and emergent properties that are difficult to predict before the system is operational in its intended environment [Senge, 1990; Checkland, 1999; Bar-Yam, 2003; Sage and Rouse, 2009:1285; Sage, 2002; INCOSE, 2011]. Requirements engineering is particularly challenging for these systems [DAG, 2011; INCOSE, 2011:8; Hull et al, 2011].

Viable system solutions require a balance of reductionism and holism [Checkland, 1999:74; Rouse, 2007:261] to understand the internal and external impact of design considerations. Functional decomposition is a key reductionism activity supporting requirements engineering and system design efforts [DAG, 2011:236; DAU, 2001:6], however the nature of complex systems makes it difficult to fully consider the impact of functional architectures and design alternatives on system behavior [Calvano and John, 2004]. The ability to grasp important properties governing the behavior of truly complex systems [Sage, 2002] may "exceed the finite capacity" [Ashby, 1958] of the SE team. These difficulties often result in poor system requirements that need to be reworked late in the system development effort after considerable program resources have been expended.

The importance of SE in the development of complex systems is well documented [DAG, 2011; DAU, 1993; INCOSE, 2011:17; Boehm et al, 2008; Honour, 2004], however SE practitioners and program managers often struggle with timely delivery of cost-effective systems that are fully compliant with technical objectives [GAO, 2011; Azizian et al, 2011; AFCSE, 2010; GAO, 2009; NRC, 2008; Valerdi, 2005; Valerdi et al, 2003]. Requirements instability and underestimation of engineering effort have been correlated with problems experienced in complex system development programs [GAO, 2011; AFCSE, 2010; GAO, 2009; NRC, 2008; Valerdi, 2005; Valerdi et al, 2003], and these problems appear to be getting worse over time [NRC, 2008]. The trend of increasing system complexity is expected to continue in the future as systems become larger in scale and new technologies enable greater interconnectivity and functionality. This trend combined with the rapid pace of technology advancement will challenge

systems engineers to improve the efficiency of their requirements engineering process and reduce the negative effects of requirements volatility on the requirements engineering effort. System developers with the ability to deliver more capable systems in less time will be able to adapt more quickly to changing customer demand and gain a competitive advantage in the marketplace.

The law of requisite variety requires system control mechanism(s) to be at least as complex as the system(s) being controlled [Ashby, 1958; Rechtin and Maier, 1997]. Thomas and Mog [1998] generalize this concept to include the "full development process, including engineering and project management" [Thomas and Mog, 1998]. The requirements engineering process is the mechanism by which the requirements engineering effort is managed and controlled. The process must account for the complexity of the system, which would reasonably include the number of system requirements possibilities. The ability to effectively manage and control the process requires considerable knowledge, skill and expertise combined with precise and accurate measurement methods.

1.2: Research Problem Situation

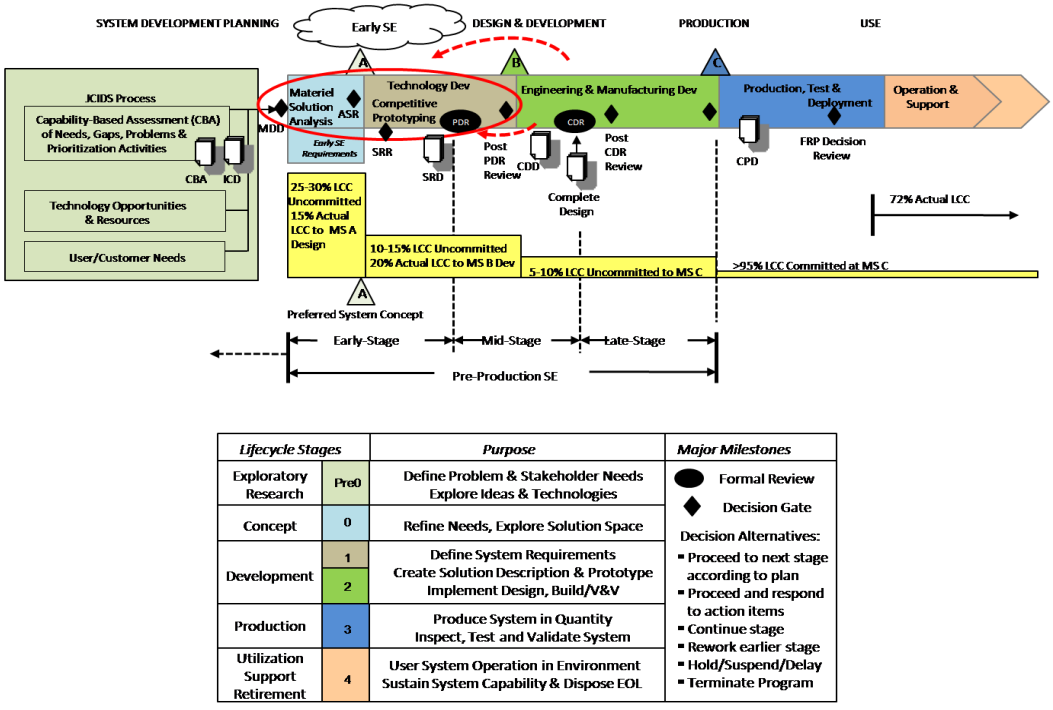
Major Defense Acquisition Programs (MDAPs) develop highly complex systems for the U.S. military. MDAPs have struggled in recent years to meet performance objectives within original cost and schedule estimates, particularly when the technology envelope is pushed [Azizian et al, 2011], requirements are unstable [GAO, 2011; NRC, 2008], or engineering effort is underestimated [Valerdi, 2005; Valerdi et al, 2003; GAO, 2009]. In a study of 96 MDAPs, the GAO found the average cost growth was $3.1 billion, 42

percent of programs had at least a 25 percent unit cost growth, and the average schedule delay in delivering initial operating capabilities (IOCs) was 22 months [GAO, 2011; GAO, 2009]. Early SE, stable requirements and accurate estimation of engineering effort have been identified as important characteristics of DoD system development programs that experienced *less* cost and schedule problems [GAO, 2011; GAO, 2009; NRC, 2008; AFCSE, 2010]. These studies and others [AFCSE, 2010; Blanchard and Fabrycky, 2005] recommend better requirements engineering activities and identify requirements volatility, underestimation of engineering effort and the lack of early SE as contributing factors to poor program performance.

As part of the strategy to address these problems, the DoD implemented major changes to its system acquisition practices in order to place greater emphasis on early SE and reduce the negative impact of requirements volatility later in the program. Revisions to the DoD acquisition lifecycle are illustrated in Figure 1. SE activities that in the past represented the formal start of system development at Milestone B have been moved into Milestone A and Materiel Solution Analysis (MSA) to ensure the system design receives rigorous review by the milestone decision authority prior to Milestone B. These reviews include, in part, assessments of the functional architecture based on the system requirements and program staffing which includes engineering effort [DAG, 2011]. The reader is referred to the DAG [2011], DODI [2008], DAU [2009] and CJCSM [2007] for more information on the revised DoD acquisition strategy and its implementation.

Quality Function Deployment (QFD) [Akao, 2004] is a mature process that has been used to engineer quality commercial systems for decades [Clausing and Cohen, 1994], however QFD practitioners have also struggled with the development of complex

systems [Hari et al, 2007; Ginn et al, 2005; Cristiano et al, 2001]. These findings suggest that the problems encountered in complex system development programs go beyond SE processes and methods. The harsh reality is that customers and users must often choose between lower performance, schedule delay or higher cost after significant program resources are expended, resulting in less successful projects and lower customer satisfaction [Miller, 2000; Kludze, 2004]. Several recent studies have suggested that a different approach is needed to improve the outcomes of complex system development programs [Calvano and John, 2004; Rouse, 2007; Boehm, 2006].



Lifecycle Stages		*Purpose*	*Major Milestones*
Exploratory Research	Pre0	Define Problem & Stakeholder Needs Explore Ideas & Technologies	Formal Review Decision Gate Decision Alternatives: ▪ Proceed to next stage according to plan ▪ Proceed and respond to action items ▪ Continue stage ▪ Rework earlier stage ▪ Hold/Suspend/Delay ▪ Terminate Program
Concept	0	Refine Needs, Explore Solution Space	
Development	1	Define System Requirements Create Solution Description & Prototype	
	2	Implement Design, Build/V&V	
Production	3	Produce System in Quantity Inspect, Test and Validate System	
Utilization Support Retirement	4	User System Operation in Environment Sustain System Capability & Dispose EOL	

Figure 1.1: DoD system development lifecycle with emphasis on early SE (adapted from DAG, 2011; JCIDS, 2007, INCOSE, 2011; DAU, 1993) mapped to generic system lifecycle (adapted from INCOSE, 2011; ISO/IEC, 2002).

1.3: Statement of Need

The DoD reforms seek to improve the quality and cost-effectiveness of complex military systems, but it is unlikely that new acquisition policies and procedures alone will

solve the problem. Systems engineers and program managers need new methods to improve the outcomes of complex system development programs [Calvano and John, 2004; Rouse, 2007; Boehm, 2006], and requirements engineering has been identified as one of the most important areas in need of improvement [NRC, 2008; AFCSE, 2010; Blanchard and Fabrycky, 2005; GAO, 2011; GAO, 2009]. "Successful technologies are predicated on precise engineering, which in turn requires high precision measurement" [Dowling and Milburn, 2002:1657]. A better understanding of the important underlying properties influencing the behavior of the requirements engineering process is needed in order to measure, manage and control the process more effectively and improve the outcomes of SE programs that rely on requirements information to design and build the right system the right way. There is little evidence in the literature to suggest that a precise and accurate method for measurement of the status and progress of the requirements engineering effort is available. There is a need for such a method to measure, monitor and predict requirements trends and the impact of requirements changes and volatility on the engineering effort. A method that simply measures the number of requirements changes is expected to be imprecise and inaccurate when one considers that changes can have positive or negative effects on the program [INCOSE, 2010:40]. If requirements are the information that enables the right system to be designed and built the right way, and quality attributes are indicators of the fitness of the requirements for this purpose [Hull et al, 2011], then a practical method that estimates, measures and forecasts the positive, negative or neutral impact of requirements changes on requirements quality would be expected to decrease the technical uncertainty of the

requirements information and improve the assessment of the status and progress of the requirements engineering effort relative to the desired end state.

SE practitioners and organizations responsible for implementing SE earlier in the system development lifecycle need a better understanding of the important relationships between underlying properties of the requirements engineering process and observed requirements engineering behavior and outcomes. Heuristics, rather than an accepted theory [INCOSE, 2007:19] or mathematical equations, typically guide the systems engineer in requirements engineering process. Better methods are needed to measure the status and progress of the requirements engineering effort relative to the desired end state, monitor the impact of requirements changes on the requirements engineering effort, and more accurately predict when the requirements will reach the desired state. This dissertation introduces a new theory and the REF for this purpose in order to improve the measurement, management, and control of the process.

1.4: The SE Process

SE is a comprehensive, interdisciplinary process for identifying a cost effective system solution that meets user needs while taking into account the entire system lifecycle from initial concept exploration and technology development planning to disposal [DAG, 2011; INCOSE, 2003; Eisner, 2002]. The SE process facilitates the documentation of the technical effort that defines the system solution and guides the systems engineer and program management team through the system development phases. The DAU Guidebook [DAG, 2011] emphasizes that "as systems become larger and more complex, the design, development, and production of such systems or systems

of systems (SoS) require the integration of numerous activities and processes" [DAG, 2011:168]. SE is the overarching engineering approach for "integrating all these acquisition life cycle activities" [DAG, 2011] in order to design and build the right system the right way.

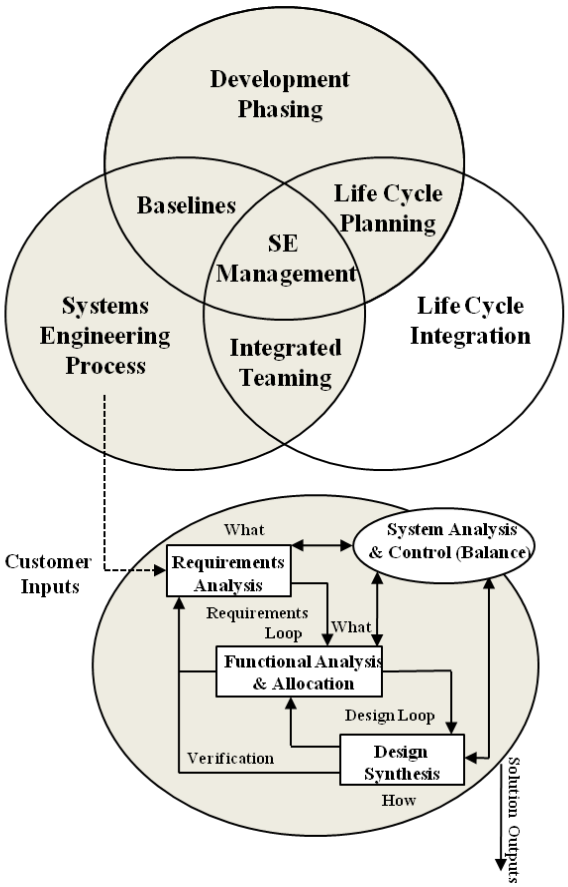


Figure 1.2: The SE Process in Relation to SE Activities [Adapted from DAU, 2001]

This dissertation focuses on the requirements engineering activities of the technical SE process illustrated in Figure 1.2 [adapted from DAU, 2001] because system requirements underpin all the activities that follow. The SE process starts with an effort to fully understand the problem and system operational environment. These activities focus on understanding the needs, top level system requirements, the system functions and the

required levels of system performance. The functional architecture is defined and allocated to the requirements to ensure all the requirements are addressed. The design loop synthesizes the functional architecture into a physical architecture that represents how the system will satisfy the requirements. Requirements are validated in the process to ensure they are necessary to build the right system that satisfies the need. The design loop serves as the basis for analysis of alternatives to determine the preferred near-optimal system solution. Requirements verification ensures the system or lower level design implementations meet the requirements.

The systems analysis and control activities include the technical and process management "required to measure progress, evaluate and select alternatives, and document data and decisions*"* [DAU, 2001:33] for each step in the SE process, including design reviews and "risk management, configuration management, data management, and performance-based progress measurement" [DAU, 2001:33]. Methods that assess the status and progress of the requirements engineering effort would generally fall into the category of systems analysis and control activities. It is well known in other scientific disciplines that continuous measurement, monitoring and control during product development enables adjustments to be made during the process that improve quality and increase the probability of a desirable outcome. For example, semiconductor epitaxial layer growth processes can be highly complex, and continuous measurement, monitoring and control of the process enables growth of higher quality interfaces and epitaxial layers with the right chemical composition, electrical properties and defect density. Continuous measurement, monitoring and control of the growth process increases the information available to the crystal grower, reduces uncertainty of the status and progress of the

material growth process, and enables corrections and adjustments to be made early to avoid poor quality layers. As a result, quality is built into the starting material that serves as the foundation for the design and fabrication of opto-electronic devices that satisfy customer needs.

Similarly, requirements are the "source material" that serves as the foundation for the design and realization of successful systems, and quality is built into the requirements during the requirements engineering process. Practical methods that enable continuous measurement, monitoring and control of requirements trends are needed to better understand the status and progress of the requirements engineering effort and make more effective adjustments during the effort to improve requirements quality and increase the probability of a successful outcome. There is little evidence to suggest that a practical quantitative method for continuous measurement, monitoring and control of the requirements engineering process is available to the SE community. The reader is referred to INCOSE [2011], DAG [2011] and DAU [2001] for more detail on the SE process and its engineering activities.

1.5: Research Significance

This dissertation introduces a theory of information quality to explain requirements engineering behavior observed in system development programs. The theory extends principles of information theory and statistical mechanics to the requirements engineering process for the purpose of gaining a better understanding of important underlying properties influencing the behavior. The requirements entropy framework (REF) is

introduced as a practical method for implementing the theory in the requirements engineering process.

The REF estimates, measures and forecasts requirements trends and engineering effort based on the proposed relationships between requirements changes, requirements volatility, requirements quality, requirements entropy and uncertainty, and engineering effort. Simulation results suggest the REF agrees with engineering effort data for DoD programs published in the literature [Valerdi, 2005]. If technical uncertainty of requirements information decreases as requirements quality increases and the objective of the requirements engineering effort is to define a complete set of requirements with sufficient quality to enable the right system to be designed and built the right way, then a method that quantifies information uncertainty and entropy in terms of requirements quality and accounts for the number of possible requirements quality distributions relative to the desired state is likely to provide a better understanding of the status and progress of the requirements engineering effort than methods that simply measure the number of requirements changes alone.

Technology refers to "the application of scientific knowledge for practical purposes" [OED, 2013]. The REF is evaluated using simulation experiments to investigate its practical utility as a new technology for measuring, monitoring, and forecasting requirements trends and estimating the additional engineering effort necessary to transition the requirements from their current state of quality to the desired end state.

1.6: Dissertation Overview

This dissertation is organized into eight chapters. Chapter 2 reviews the SE literature with particular emphasis on the requirements engineering problem, the positive and negative impact of requirements changes on the engineering effort, and the importance of requirements quality. Also included is a brief review of the literature on the relationship between entropy, energy and information which lays the foundation for constructing a new theory of the requirements engineering process. Chapter 3 discusses the "partial similarity" [Maxwell, 1855 from Howard, 2007] between the SE process and an open physical system. Basic principles of statistical mechanics and information theory are extended to the SE process. The theory of information quality is introduced in Chapter 4 and axioms of the new theory are discussed in the context of the axiomatic foundation of information theory and statistical mechanics. Chapter 5 describes the REF as a means of implementing the theory in the SE process for the purpose of resolving requirements engineering problems in complex system development programs. The mathematical foundation of the REF is presented. Chapter 6 describes the research approach for evaluating the REF and includes a summary of the assumptions that were made in order to implement the REF in MATLAB® [Mathworks, 2010] and conduct the simulation experiments. Chapter 7 presents results from simulation experiments that suggest the REF is a promising new method for measuring, monitoring and predicting requirements trends and engineering effort at any time in the requirements engineering process. The results in Chapter 7 illustrate the advantages of the REF for estimation, measurement and forecasting of requirements trends relative to a current method that appears to be in widespread use [Rhodes et al, 2009; INCOSE, 2007; INCOSE 2010; Wiegers, 2006:180;

Costello and Liu, 1995; Loconsole et al, 2005; Xu et al, 2006; Stark et al, 1999]. The results from statistical analyses are also presented. Chapter 8 summarizes the findings and offers suggestions for future research of the theory and its implementation in SE.

Chapter 2: Literature Review

2.1: System Complexity

The Department of Defense (DoD) needs highly complex systems to conduct successful missions across the full spectrum of military operations. DoD systems are becoming increasingly complex as users demand greater technological advantage, increased situational awareness, and more timely information on the battlefield. The use of SE has become mandatory for Major Defense Acquisition Programs (MDAPs) to "translate operational needs and/or requirements into a system solution" [DODI, 2002], however cost-effective development of new military systems continues to challenge the defense acquisition community and industrial base [GAO, 2011; GAO, 2009].

In a retrospective analysis sponsored by the U.S. Air Force and published by the National Research Council (NRC), the review committee found that modern military systems "take two to three times longer to move from program initiation to system deployment than they did 30 years ago" [NRC, 2008]. Increased complexity of systems and operational tasks has been suggested as one possible contributing factor to this trend, but the general consensus of the committee is that better implementation of SE processes and functions earlier in the system development cycle by systems engineers with experience and knowledge in the program technical domain would enable more cost-effective delivery of quality systems on time and on budget [NRC, 2008].

The findings of the NRC are consistent with those of the Government Accounting Office study of 96 MDAPs [GAO, 2009]. The GAO study found major discrepancies between original program estimates and actual program performance [GAO, 2009]. Specifically, the GAO found that the average program cost growth was $3.1 billion, 42 percent of programs had at least a 25 percent unit cost growth, and the average schedule delay in delivering system initial operating capabilities (IOCs) was 22 months [GAO, 2009; GAO, 2011]. The GAO report concludes that early SE in addition to requirements stability and disciplined software development are important characteristics of programs that experienced *less* cost and schedule problems, however little insight is provided into the factors shared by successful programs. Supporting documentation in the appendices of the studies [GAO, 2009; GAO, 2011] suggest underestimation of complexity, and therefore underestimation of engineering effort, may have been a major contributor to the cost and schedule problems in at least 6 MDAPs.

Kludze in his dissertation on the engineering of complex systems suggested in 2004 that SE is necessary "very early" in the project "to maximize its technical performance while meeting project goals [Kludze, 2004]. The importance of early SE effort has been well documented by the Defense Acquisition University [DAU, 1993] and these findings have been included in the INCOSE SE Handbook [2003] for many years. DAU [1993, 2001] illustrated that as much as 85 percent of total system lifecycle costs are committed early in the SE design phase when only 15 percent of actual program money has been spent, and the cost to extract defects early in the SE design phase is orders of magnitude lower (e.g. 100-1000x) than the cost to extract defects in the system production phase. These data clearly underpin the need for effective SE early in the system development

cycle. SE processes, methods and tools that improve efficiency of requirements engineering activities or improve requirements quality early in the complex system development program continue to be sought by practitioners, program managers and decision-makers.

In an attempt to improve performance of DoD acquisition programs, Congress passed the Weapon Systems Acquisition Reform Act in 2009 [WSARA, 2009] establishing requirements for the Defense Acquisition System, including early SE. Senator Levin, Chairman of the Senate Arms Services Committee, expressed support for WSARA by stating "the key to successful acquisition programs is getting things right from the start with sound systems engineering, cost estimating, and developmental testing early in the program cycle" [DAU, 2009]. Many revisions to DoD acquisition policies, functions and processes were made as a result of the WSARA in order to place greater emphasis on early SE, including analysis of alternatives, cost estimation, competitive prototyping, technology maturity and risk analysis, performance assessment, root cause analysis and formal program certifications and design reviews [WSARA, 2009]. These reforms seek to improve the quality and cost-effectiveness of modern military systems and the processes and functions that are required to achieve them. Requirements quality enables design quality and design quality enables system quality.

The recent GAO [2009] findings suggest a relationship between underestimation of system complexity and inadequate resource allocation early in the Concept and Technology Development phase shown in Figure 1.1. For DoD MDAPs in which the use of SE is required, this underestimation may increase the risk of system requirement instability after Milestone B, which in the past often represented the formal start of the

DoD system acquisition. The GAO [2009] study suggests a strong correlation between MDAP cost growth relative to original estimates and requirements instability after Milestone B. These findings suggest that top level system requirements as a minimum should be stable before Milestone B to support the detailed system requirements engineering and design effort in EMD necessary to build the objective system with the required performance on schedule and within budget.

As a general rule of thumb and in an effort to mitigate risk that any given design level does not become too complex, INCOSE recommends that system hierarchies contain no more than 7 ± 2 functionally-derived subsystems that report to it [INCOSE, 2003:20] and system specifications incorporate between 50 and 250 functional and performance requirements, excluding physical, non-functional and environmental-based requirements [INCOSE, 2003:89]. In a study of 35 DoD system development programs, the total number of requirements necessary to describe the system varied from approximately 70 to 20,000 [Valerdi, 2005]. Clearly, limiting the total number of functional and performance requirements to 250 is inadequate for many of the more complex systems being developed today.

Thomas and Mog [1998] suggest a model for estimating system complexity based on a technology covariance matrix using technology readiness, development schedule, percent of total project cost, and organizational dispersion for each system component [Thomas and Mog, 1998]. There is little research available on the correlation between system complexity, early-stage system design and development factors, and the degree of system development program success. A method for estimating system complexity early in the development cycle and its corresponding requirements engineering effort may help SE

practitioners and program managers prepare more realistic program plans, forecast program milestones more accurately, more effectively prioritize requirements engineering and design activities, and more efficiently allocate program resources to increase probability of success.

Bashir and Thomson [1999] suggest a metric for estimation of system complexity based on functional decomposition, and retrospective analysis of various projects in two companies resulted in strong correlation between the system complexity metric and total design effort (greater than 0.90 and 0.91, respectively) [Bashir and Thomson, 1999], however the correlation did not include project success. Of the seventeen (17) total projects included in the analysis, only two involved design efforts that exceeded 10,000 hours. Botta and Bahill [2007] suggest "complex features should have high priorities" and identify twenty (20) criteria to consider in the prioritization process, including assigning the most capable people to work on the most complex features [Botta and Bahill, 2007]. Further study is needed to determine whether there is a statistically significant difference in the design effort required for systems of comparable complexity with and without prioritization. There does not appear to be a standard system complexity metric available in the SE literature that would underpin such a study or other system complexity dependent studies.

2.2: Requirements Engineering

The requirements engineering process includes all the activities involved in the discovery, definition, development, analysis, verification, validation and management of requirements [DAG, 2011:209; DAU, 2001; INCOSE, 2011; SEI, 2010; Pohl, 2010].

Requirements drive and bound the design process responsible for identifying the system solution [INCOSE, 2011:70] and are the "formal technical statements that define the capabilities, characteristics, or quality factors of a system" [Honour and Valerdi, 2006:6]. Requirements engineering is a "long-recognized core discipline of systems engineering" [Honour and Valerdi, 2006:6], is among the most important and difficult activities in SE [Bahill and Henderson, 2005; Kivinen, 2008; Gambhir, 2001; Lamsweerde, 2009:47] and is particularly challenging for complex systems [DAG, 2011; INCOSE, 2011:8; Hull et al, 2011]. Many recent studies recommend better requirements engineering activities to address problems in system development programs [NRC, 2008; AFCSE, 2010; Blanchard and Fabrycky, 2005; GAO, 2011; GAO, 2009], and DoD has implemented major changes in its system acquisition practices to place greater emphasis on early SE to reduce the negative effects of requirements volatility later in the program.

2.2.1: Requirements Trends Metrics

Poor requirements have been linked to requirements volatility [Anthes, 1994; Kulk and Verhoef, 2008], and requirements volatility has been identified as a major factor contributing to poor program performance [GAO, 2011; GAO, 2009; Rajnish, 2010:30; Clark et al, 2007:89]. Current requirements metrics "supply only a limited portion of the information necessary to identify, isolate, and resolve problems" during system development [Costello, 1995:39; Costello, 1994]. Requirements volatility problems extend to many hardware and software industries [Wiegers, 2006]. Requirements evolve over time during the requirements engineering process but "all changes are not equal [Lane and Boehm, 2007:2]. Requirements trends are among the most important leading

indicators of SE program technical effectiveness and requirements volatility has been suggested as an indicator of good SE practices [Rhodes et al, 2009], however there is little evidence to suggest that current leading indicators differentiate between good requirements changes that move the engineering effort in the right direction from the negative effects of requirements volatility. Many requirements volatility metrics published in the literature are requirements change metrics and take the general form:

$$RV = \frac{r_a + r_r + r_d}{R} \qquad (1)$$

where r_a, r_r and r_d are the number of requirements added, revised (changed, modified) and deleted, respectively, R is the number of requirements and RV is the total fractional requirements volatility [INCOSE 2010; INCOSE, 2007; Rhodes et al, 2009; Wiegers, 2006; Costello and Liu, 1995; Loconsole et al, 2005; Xu et al, 2006; Stark et al, 1999, Maadawy and Salah, 2012]. This metric has shed some light into requirements trends [Rhodes et al, 2009:31; Wiegers, 2006:180; Costello, 1995:56; Xu et al, 2006; Stark et al, 1999:297; Maadawy and Salah, 2012:5], however it has major limitations.

Equation (1) simply measures the quantity of requirements changes over time. In effect, equation (1) treats all changes the same because requirements additions, deletions and revisions are weighted equally. The relative importance of changes and the degree of their positive or negative effects are not clearly distinguished. For example, revisions that reduce technical difficulty, make the requirements more realistic given the current state of technology maturity, and enable system delivery on schedule [Bottelson, 2011] are treated the same as new requirements added late in the program that significantly

increase technical difficulty, impose unrealistic performance levels or are based on incorrect technical facts. INCOSE [2010] acknowledges that "change may be good, detrimental, or neutral to our system development" [INCOSE, 2010:40] and change can have negative "ripple effects" [INCOSE, 2010:39]. These "ripple effects" can reduce requirements quality. A single revision can negatively impact numerous other requirements but only a fraction of the volatility may be captured by equation (1) in the corresponding measurement interval if the requirements are not traced or annotated to facilitate analysis. Requirements volatility in this research refers to the negative change effects that result in the transition of one or more requirements from a lower entropy state to a higher entropy state. These effects and their impact on the engineering effort are often unpredicted and uncontrolled.

The importance of early SE [GAO, 2009; GAO, 2011; NRC, 2008] and technology maturity [Azizian et al, 2011] in successful system development programs is clear, however the impact of requirements volatility is more subtle when one considers that requirements engineering is an iterative process and by definition includes changes as requirements are *transformed* [DAG, 2011:168]. Implicit in this requirements transformation process is the transition from a less desirable state of quality to a more desirable state so the right system can be designed and built the right way. Wiegers [2006] acknowledges the "intent is not to eliminate requirements volatility but to ensure the project can manage the degree of requirements changes and still meet its commitments" [Wiegers, 2006:181]. Kulk and Verhoef [2008] suggest requirements changes are "necessary and healthy…..and a fact of life" but the challenge is to "control the volatility" [Kulk and Verhoef, 2008:138]. Requirements changes are inevitable to

uncover hidden requirements, delete unnecessary requirements, and transform needs into requirements with sufficient quality for system design [DAG, 2011:210].

The positive effects of requirements changes may be offset by negative effects of requirements volatility. For example, requirements volatility increases project scope in the vast majority of projects (0.78) and the negative effect is particularly high when requirements volatility occurs after the start of the formal design process [Ferreira et al, 2009:1570]. Requirements volatility at Milestone B in DoD system acquisition programs has been correlated with poor performance [NRC, 2008; AFCSE, 2010; GAO, 2009; GAO, 2011]. The need for more stable requirements at Milestone B has increased the emphasis on early SE in the DoD system development cycle.

The requirements added r_a, revised r_r, deleted r_d, and total requirements volatility *RV* in equation (1) correspond to the data shown in Figures 2.1 and 2.2. The Rhodes et al [2009] data [INCOSE, 2007; INCOSE, 2010] in Figure 2.1 include linear extrapolation of the *RV* to forecast when the *RV* will reach zero and would be suitable for formal review. The Rhodes et al [2009] data suggest that an event-based system requirements review (SRR) would likely be more productive than a schedule-driven SRR based on the *RV* at the time the SRR was originally scheduled. However, this linear regression forecast may result in significant errors [Franklin, 1986].

The data presented in Figures 2.1 and 2.2 illustrate the Fractional Method for measuring requirements trends and forecasting when the requirements changes will be zero. In both cases, new requirements contribute the most to the total *RV* early in the program (shown by the blue line at baseline measurement interval 0). This is intuitive given that programs often begin system development with top level requirements added

to reflect what the customer or user needs. As the requirements engineering effort progresses, requirements are further analyzed, defined, derived, uncovered, added, revised, and deleted as appropriate to sufficiently describe the system and its subsystems to the lowest level necessary to design and build the right system the right way.

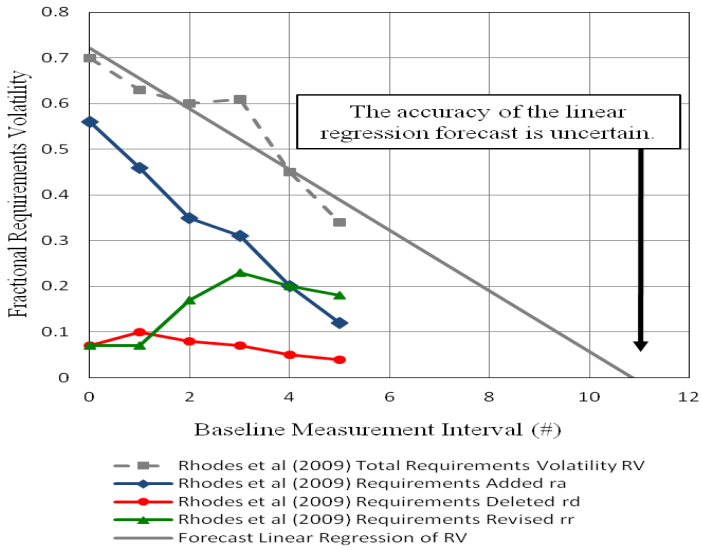


Figure 2.1: Fractional Method Requirements Trend Data [adapted from Rhodes et al, 2009; INCOSE, 2010].

The Fractional Method appears to infer an increase in requirements quality by simply monitoring the overall decreasing trend in the *quantity* of additions, deletions or revisions that are made over time. It is important to note in Figure 2.1 [Rhodes et al, 2009] that requirements revisions become the dominant contributor to the *RV* at the point where the data end. The overall trend in requirements revisions from program start to baseline measurement interval 5 is actually *increasing*, and yet the Fractional Method using linear regression forecasts that the *RV* will reach zero and the program will be ready for a formal review at baseline measurement interval 11. There is little evidence to suggest that the *RV* trend would reach zero at baseline measurement interval 11. For example, if the *increasing* trend of requirements revisions in Figure 2.1 leads to more volatility

because important hidden requirements are uncovered or the scope of the development program increases, which occurs in the majority (0.78) of projects with volatility [Ferreira et al, 2009], then the program may not be ready for a formal review at baseline measurement interval 11. The Fractional Method does not quantify the degree of positive or negative impact that additions, deletions and revisions have on requirements quality relative to the desired end state. The Costello and Liu [1995] data in Figure 2.2 show significant differences in requirements trend behavior compared to the Rhodes et al [2009] data [INCOSE, 2007; INCOSE, 2010], although r_d per time interval in both cases is less than 0.10 *R*. The Costello and Liu [1995] data illustrate the potential for significant errors when a linear extrapolation method is used to forecast when *RV* will reach zero because the intercept occurs 25 percent earlier than the end of the effort. Figures 2.1 and 2.2 underpin the need for a better leading indicator that "characterizes the stability and completeness of the system requirements which could potentially impact design and production" [INCOSE, 2007:10].

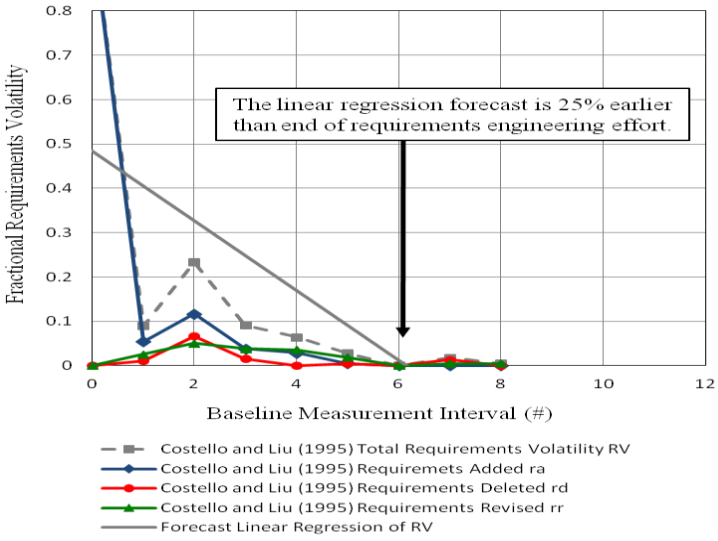


Figure 2.2: Fractional Method Requirements Trend Data [adapted from Costello and Liu, 1995].

The remainder of this dissertation refers to the metric defined by equation (1) combined with linear extrapolation to forecast when the requirements will reach $RV = 0$ as the Fractional Method. There are very few papers available in the literature with requirements change and volatility data for SE programs. The data shown in Figure 2.1 [INCOSE, 2010; INCOSE, 2007; Rhodes et al, 2009] and Figure 2.2 [Costello and Liu, 1995] provide valuable insight into the variability of the requirements trend phenomena. The engineering effort data measured by Valerdi [2005] for thirty-five (35) DoD SE programs are particularly noteworthy for their completeness, simplicity and value. The Valerdi [2005], Rhodes et al [2009] and Costello and Liu [1995] data represent the macroscopic behavior observed during the requirements engineering process. Results from simulation experiments that compare requirements trend forecasts of the REF and the Fractional Method are presented in Chapter 7.

2.2.2: Requirements Quality

Requirements quality has "the largest potential impact on the success of system developments [Davis, 1993:5]" [Gambhir, 2001] because the engineering effort "may not satisfy user needs" or may lead to "the wrong system" being built if requirements quality is ignored [Davis, 1993]. Poor requirements lack quality attributes that are important to effectively communicate the information that enables the right system to be built the right way. Unlike Shannon's mathematical theory of communication in which the *meaning* of the message is "irrelevant to the engineering problem" [Shannon, 1948:1], the meaning of the requirements information and how the requirements are interpreted and implemented by the design team is highly relevant to the systems engineering problem. In the systems

engineering context, the meaning of the requirements information is reflected by its quality, requirements quality enables design quality and design quality enables system quality.

Davis [1993] defined twenty-four (24) quality attributes that should apply to a requirements document, approximately fourteen (14) of which are applicable to individual requirements. Poor requirements often result in system development problems [Gambhir, 2001; Ricker, 1995] and are often cited as a primary reason why systems engineering and development efforts fail [Eisner, 2002:225]. Requirements validation [INCOSE, 2011:136] includes activities that support the assessment of requirements quality during the requirements engineering process, and requirements quality attributes [Davis, 1993; INCOSE, 2011:76] are often used in the assessment. Poor requirements lack one or more quality attributes. Rigorous requirements V&V reduces uncertainty about whether the program is moving in the right direction and the right system will be designed and built the right way.

INCOSE generally describes SE Quality Management as a process that establishes, implements, and continuously improves the customer-focus of the organization within the organization's goals and objectives. This is similar to the focus of quality function deployment (QFD) discussed in Section 2.3. There is general recognition that costs and benefits are associated with quality management, and the "effort and time required to manage quality should not exceed the overall value gained from the process" [INCOSE, 2003:295]. The INCOSE SE Handbook emphasizes that a primary objective of Quality Assurance is to "produce an end result that meets or exceeds stakeholder expectations….. and quality management is closely related to the V&V processes" [INCOSE, 2003:297].

V&V are important steps in the SE process because these activities serve as test points to measure compliance with customer and technical requirements and identify deficiencies in the design that need to be addressed either before or after the system is built.

2.2.3: Engineering Effort

Honour [2004] measured the comparative success of system development programs as a function of the SE effort for a variety of sytem development programs (n=30), and the data provide further insight into the importance of an adequate level of SE effort to increase the probability of a successful outcome. The data suggest random variable behavior of comparative success for system development programs with SE effort less than 8 percent of the total program budget. A probability plot of the Honour [2004] data is shown in Figure 2.3. The mean comparative success of programs with SE effort less than 8 percent is 5.0, suggesting equiprobability of an outcome relative to the success of other programs. The Honour [2004] data suggest that the performance of programs with SE effort less than 8 percent is just as likely to be more successful relative to other programs as less successful. The mean comparative success of programs with SE effort greater than 8 percent increases significantly to approximately 8.2 [Honour, 2004]. The data provide interesting insight into the stochastic behavior of the complex SE process and its relationship with program success.

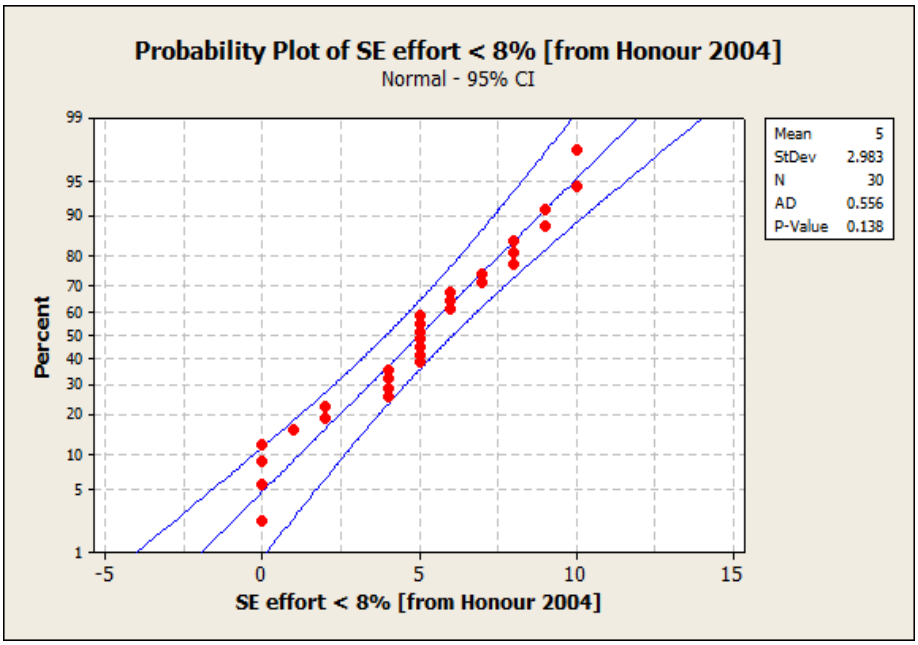


Figure 2.3: Probability plot of Honour [2004] data for thirty (30) SE programs with SE effort ≤ 8 percent of total program.

2.3: Quality Function Deployment (QFD)

SE and comprehensive QFD are complex processes for control of the design, development and production of complex systems. The comprehensive QFD system is often represented schematically as a step-by-step process for identifying quality, technology, cost and reliability characteristics of a product that are required to satisfy customer needs, and linking that information to functions and characteristics of parts for deployment throughout the production process [Akao, 2004]. In the flow of QFD, Akao [2004] shows three levels of final product quality characteristics that are mapped to three levels of customer requirements and subsequently deployed to a quality control process chart for production. The "subsystem deployment" activity within the phase I quality deployment column of the comprehensive QFD process is an extension of the 'House of

Quality' (HoQ) [Clausing and Cohen, 1994; Sawada, 1979]. The HoQ often appears to contain a mix of functional (e.g. electrical, mechanical, optical, etc.) and non-functional (size, shape, weight, etc.) characteristics, or technical attributes, in the engineer's room when shown in the literature.

The benefits of QFD in new product development are well documented [Akao and Mazur, 2003; Akao, 2004]. QFD has been generally characterized by practitioners as a system engineering approach [Akao, 2004; Dean, 1992] and a complement to the systems engineering process. The literature suggests that QFD practitioners are often frustrated with the time and difficulty associated with the comprehensive QFD process when applied to the development of complex systems [Ginn and Zairi, 2005]. Recent findings suggest QFD requires modification when applied to complex systems [Hari et al, 2007] and the 4-phase QFD model may not be effective for development of complex systems [Cristiano et al, 2001].

SE can be generally characterized as a function-based analysis approach to embed quality requirements into system design, and QFD can be generally characterized as a product-based analysis approach to embed quality functions into organizational product development processes for the purpose of delivery quality products that satisfy customer needs. These general categorizations are not intended to describe the complete SE and QFD processes or their full implementation, nor is it intended to imply constraints for process tailoring. The physical architecture-based perspective of QFD appears to generally reflect the perspective of many publications reviewed during this research.

QFD has been described by the QFD Institute [2011] as "a system of comprehensive development process(es)" for understanding customer needs, linking those needs

throughout the organization's "design, development, engineering, manufacturing and service functions", and determining the product attributes and performance level to deliver [QFD Institute, 2011]. Mizuno and Akao [1978] describe QFD as a system for "functional analysis of the business process phases in order to improve the quality of the product development process itself" [Mizuno and Akao, 1978]. Some interpret functions, in the broader context of QFD, to refer to the organizational functions required to develop a quality product, not the product functions required to meet customer needs. Quality functions are generally described as the organizational functions required to form or contribute to a quality product [Akao, 2004]. Quality management is an important element of SE management, however there appears to be major differences between SE and QFD in their emphasis on generating a detailed functional architecture of the system.

Cristiano et al's [2001] study of more than 400 U.S. and Japanese companies did not find a positive relationship between QFD techniques, including HoQ or HoQ quantitative tool integration (e.g. AHP) and project success [Cristiano et al, 2001]. Some familiar with the study suggest the results may reflect the lack of use of the comprehensive QFD model [Mazur, 2011]. Cristiano et al [2001] suggest the only positive relationship between the comprehensive QFD model and project success was for product innovation, but a *negative relationship* between QFD and complex system innovation was found [Cristiano et al, 2001]. A recent retrospective study of quality management did not find a relationship between product design and management, which included QFD, and product quality [Nair, 2006]. Nair [2006] does not discuss the complexity of the systems included in the study, and these findings may be consistent with those of other researchers who suggest QFD problems in complex system development programs.

2.4: Relationship between Entropy, Information and Energy

The relationship between entropy, information and energy is well documented in the literature [Szilard, 1929; Shannon, 1949; Brillouin, 1951; Brillouin, 1962; Jaynes, 1957; Jaynes, 1980; Renyi, 1961; Tribus and McIrvine, 1971; Bailey, 1987; Zurek, 1990], and this relationship is central to information theory and statistical mechanics. The number of possible microstates, or the *number of possibilities,* that comprise physical systems can be astronomical, and statistical mechanics describes the relationship between the observed macroscopic behavior and the number of possible microstates corresponding to that behavior. As the number of possibilities decrease, information and certainty increase.

Bailey [1987] discusses the relationship between energy, entropy and information in the context of information theory, classical thermodynamics and statistical mechanics equations, and his paper attempts to resolve the various anomalies and contradictions between these important system properties reported in the literature [Bailey, 1987]. Bailey [1987] suggests these anomalies and contradictions are resolved when one considers that "as entropy increases, order always decreases" [Bailey, 1987:91], "information will always vary inversely with entropy" [Bailey, 1987:91], and "a high degree of order indicates a system which has a low degree of entropy, and high degrees of energy and information" [Bailey, 1987:92]**.** These basic premises are consistent with the principles extended to the REF defined in Chapter 5.

Brillouin [1956] provides a detailed discussion on the relationship between information and statistical mechanics, and in the process suggests an entropy equation for a "message using n different types of pulses" [Brillouin, 1956:341] that lays a mathematical foundation for the requirements entropy equation reformulated in this

dissertation and extended to the REF. Brillouin [1956] suggests "entropy and probability are practically synonymous for the physicist who understands the second principle of thermodynamics as a natural tendency from unprobable to more probable structures" [Brillouin, 1956:338]. The "more probable structures" are characterized by higher entropy and disorder. The requirements engineering process works against higher entropy and disorder by transitioning requirements from more probable distributions to less probable distributions having lower entropy and higher order.

Tribus and McIrvine [1971] also discuss the relationship between energy and information, and suggest values of energy per information output associated with analog and digital information systems. This previous work led to the question of whether principles of information theory and statistical mechanics could be used to define a relationship between energy in the form of engineering effort, requirements entropy and information, and cumulative quality as a measure of the disorder of the system.

Chapter 3: Systems Thinking

3.1: The Importance of Systems Thinking in SE

The SE literature is replete with definitions and interpretations of systems thinking [Checkland, 1999:74; Senge, 1990; Sage, 1995; Kim and Wilemon, 2003; Davidz, 2006:5; Gill, 2009]. For the purpose of this research, systems thinking is the process of exploring and conceptualizing important relationships between macroscopic system behavior and the fundamental properties and interactions of constituents contributing to that behavior. This definition captures elements of the INCOSE [2011:8] definition while using terminology that reflects the systems thinking that led to the REF.

The role of systems thinking in the SE process is becoming increasingly important as systems complexity increases [Gill, 2009:1; Frank, 2000:1; Bahill and Gissing, 1998]. As new technology enables the design and development of increasingly complex systems, engineering organizations will place greater emphasis on systems thinking and allocate resources to identify systems engineers with systems thinking skills [Gill, 2009]. Consideration of system properties as a whole is particularly important in complex systems engineering [Sage and Rouse, 2009:1294]. Viable solutions to the unique challenges posed by complex systems require a balance of reductionism and holism [Checkland, 1999:74; Rouse, 2007:261]. For example, functional decomposition is a key reductionism activity supporting requirements engineering and system design efforts [DAG, 2011:236; DAU, 2001:6], however the nature of complex systems makes it difficult to fully consider the impact of functional architectures and design alternatives on system behavior [Calvano and John, 2004]. The ability to grasp the important microscopic properties governing complex system behavior and develop theoretical constructs to explain their relationships requires systems thinking. The role of systems thinking in the formulation of the probabilistic theory of information quality and its implementation using the REF is summarized in the following sections of this chapter.

3.2: The SE Process as an Open System

The concept of entropy is among the most important in science [Curado and Nobre, 2004] and was first introduced in the study of thermodynamics by Boltzmann [1896] to explain the macroscopic behavior of physical systems. Boltzmann's insight into the 2nd law of thermodynamics enabled him to describe the *physical* entropy *S* of a macroscopic

system in terms of probability, or the number of microstates *W* corresponding to the observed system macrostate, using:

$$S = k \log W \qquad (2)$$

thereby establishing the foundation for modern statistical mechanics and the theory for the "interpretation of equilibrium properties of macroscopic systems" [Hill, 1960:2]. It was well known that the entropy of a physical system expresses its disorder [Dincer and Yunus, 2001:125].

The concept of entropy continues to be applied in new and creative ways. For example, information entropy is being applied in the study of physical quantum-mechanical systems [Massen et al, 2001], and the principle of maximum entropy has been observed in many disparate complex physical systems [Dewar, 2003:1]. The original concept of physical entropy was extended to information entropy, and now information entropy is being applied to better understand physical systems. Systems thinkers are applying an interdisciplinary approach to discover new ways for gaining insight into the relationship between macroscopic system behavior and the important underlying properties contributing to that behavior.

The law of requisite variety [Ashby, 1958] requires an effective control system to be at least as complex as the system it is intended to control. The number of requirements possibilities and therefore requirements uncertainty increases as the number of requirements *R* increases, and *R* increases as system complexity increases. If the objective of the requirements engineering process is to define the right set of *R* and the number of possible requirements choices is astronomical, then it would be reasonable to expect an effective method for measurement and control the requirements engineering

process to take into account all the possibilities. Statistical mechanics explains the macroscopic behavior of highly complex physical systems comprised of an astronomical number of particles in terms of the number of possible microstates available to the particles and their most probable particle energy distribution.

We know from information theory that information increases as uncertainty, entropy and the number of possible choices, or degrees of freedom, decrease [Shannon, 1948; Shannon and Weaver, 1949; Bailey, 1987; Schroeder, 2004]. Would it be worthwhile to explore the complex systems engineering problem in the context of complex system problems encountered in other disciplines? Would a new method that accounts for the astronomical number of requirements choices and the most probable requirements quality distribution at any given time in the process explain the macroscopic behavior published in the literature and improve our understanding of the status and progress of the requirements engineering effort in complex system development programs?

The requirements engineering process is particularly difficult for complex systems [DAG, 2011; INCOSE, 2011:8; Hull et al, 2011]. Systems thinking during this research led to the realization that a viable theoretical construct of the requirements engineering process for complex systems should account for the *number of requirements possibilities* and the impact of their dynamics as requirements change, transform and improve in quality during the process. How can the SE team consider all these possibilities? In the case of truly complex systems [Sage, 2002], it may "exceed the finite capacity" [Ashby, 1958] of the SE team to manage and control particularly when high level system or operational requirements are volatile, which may explain in part the requirements engineering difficulties experienced in complex system development programs.

The requirements engineering process is considered to be in dynamic equilibrium and therefore stable when the positive effects of requirements changes resulting from the input of engineering effort balance and compensate for the negative effects of poor requirements and requirements volatility such that the engineering effort continues to transform the requirements toward the desired state of quality. The process is in static equilibrium when the requirements reach the desired end state and no additional changes are necessary. Processes that are dynamically stable can adapt to the changing needs and requirements of customers and maintain their stability over a wide range of external conditions.

The relationship between entropy, information and energy is well documented in the literature [Szilard, 1929; Shannon, 1949; Brillouin, 1951; Brillouin, 1962; Jaynes, 1957; Jaynes, 1980; Renyi, 1961; Tribus and McIrvine, 1971; Bailey, 1987; Zurek, 1990], and this relationship is central to information theory and statistical mechanics. The number of possible microstates, or the number of possibilities, that comprise physical systems can be astronomical, and statistical mechanics describes the relationship between the observed macroscopic behavior and the number of possible microstates corresponding to that behavior. As the number of possibilities decrease, information and certainty increase.

Paraphrasing James Clerk Maxwell [1864], the process of becoming familiar with the physical analogy or "partial similarity between the laws of one science and those of another which makes each of them illustrate the other" [attributed to Maxwell by Howard, 2007] led to the realization that an open physical system and the requirements engineering process share 'partial similarities'. The requirements engineering process is

a system [INCOSE, 2011:5] comprised of interacting elements organized for the purpose of engineering requirements with sufficient quality to enable the right system to be built the right way. Further, the requirements engineering process is an open system because systems engineers interact with it, control the rate at which engineering effort is input across its boundary, and manage the transformation of requirements information toward states of lower entropy and uncertainty. The requirements engineering process is a dynamic process whose purpose is to transform and refine requirements in a controlled manner. Requirements volatility occurs when the impact of changes is unpredictable and uncontrollable, resulting in increased requirements uncertainty and disorder. When this occurs, the engineering effort is unable to maintain stability of the current state and entropy increases. The duration of the negative volatility effects is an indication of effectiveness of the requirements engineering effort, its management activities and control processes.

3.3: Partial Similarities between SE and an Open Physical System

In an open physical system, the order and predictability of particles can increase over time as energy flows in and entropy and energy flow out. Although the entropy of a closed system can only increase, the entropy of an open system can decrease locally when energy is transported across its boundary [Dincer and Yunus, 2001:125] or other forces are exerted. Physical entropy decreases as the randomness or uncertainty of the number of possible particle microstates decreases. The number of possibilities increases as the number of requirements R increases, and R increases as system complexity increases.

Particles in a physical system occupy energy levels corresponding to their accumulated energy, and the physical entropy reflects the number of ways that the particles can reconfigure themselves among the available states. An illustration of the particle energies within a one-dimensional energy space of a physical system is shown in Figure 3.1. In this example, there are fourteen discrete energy levels available for the total number of particles to occupy, with ε_1 representing the lowest energy state. In quantum statistical mechanics, constraints are imposed in this case on the number of particles *N* distributed among the *j* energy levels and the total system energy *E* by the following equations:

$$N = \sum_j n_j \qquad (3)$$

and

$$E = \sum_j n_j \, \varepsilon_j \qquad (4)$$

where ε_j is the energy of the particles occupying the *j*th level and n_j are the number of particles occupying the *j*th level.

For a given *E*, the number of possible distributions of *indistinguishable* particles among the levels can be estimated using Bose-Einstein statistics as follows:

$$P_{BE} = \frac{(N+G-1)!}{N!(G-1)!} \qquad (5)$$

where *G* is the number of states, or cells, available for the *N* particles to occupy. At any given time, the energy of every particle can be represented in a one-dimensional energy space as illustrated in Figure 3.1 [Richtmyer et al, 1969].

Stability in a closed, isolated system corresponds to the most probable state which is maximum entropy and $dS = 0$. An open system can be stable when energy is input and the most probable distribution corresponds to lower entropy for a given number of particles. The most stable macrostate is the *least likely to change* in an uncontrolled, unpredictable way under the current conditions. The equilibrium state can be estimated from the most probable state corresponding to the constraints imposed by equations (3) and (4) on the number of particles N and total energy E, respectively at a given time within a one-dimensional energy space.

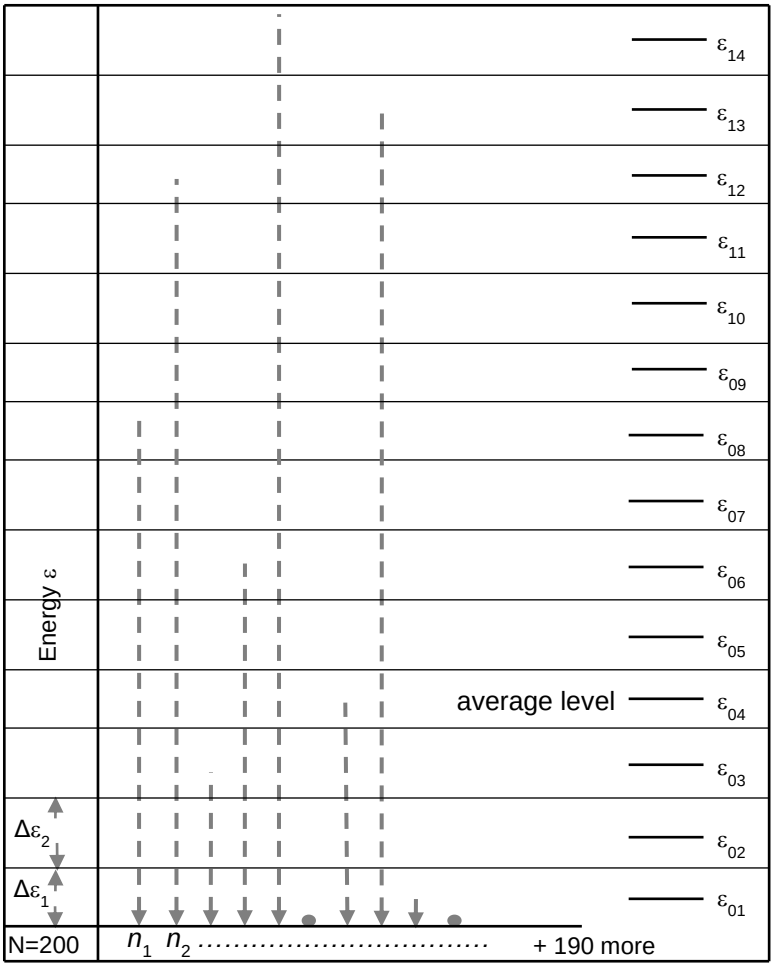


Figure 3.1: Representation of particle energies in one-dimensional energy space [adapted from Richtmyer et al, 1969:532].

The number of possible arrangements of two indistinguishable particles with three energy levels available for them to occupy is illustrated in Figure 3.2. For this simple case, the total number of possibilities, or the number of possible arrangements of the two particles among the three available energy states, is six. If there was a need to distinguish between the particles to determine which particle occupied which energy state, Maxwell-Boltzmann statistics could be used given the total number of particles and total energy of the system is conserved for a temperature above zero.

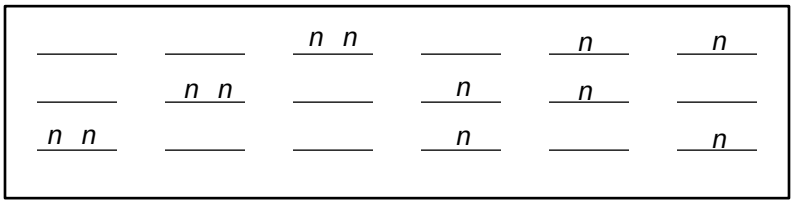


Figure 3.2: The possible arrangements for 2 particles among 3 states.

For the example illustrated in Figure 3.1 where N = 200 and all energy levels are available for the particles to occupy where G = 14, $P_{BE} = 2.1 \times 10^{20}$ possible particle distributions. As temperature decreases, the particles have fewer energy levels available for them to occupy and therefore the physical system becomes less disordered, random and uncertain. Temperature is a measure of disorder [Brissaud, 2005].

Particles in a physical system behave randomly and tend toward maximum entropy and disorder because such a condition represents the most possible ways the system can reconfigure itself and therefore is most probable without the input of energy to counteract this natural tendency toward maximum entropy. In some cases, particles may tend toward the lowest energy states for a given set of system conditions rather than maximum entropy. The lowest energy state of a physical system can be represented by its ground

state where there is one possible configuration and the entropy of its constituents is zero. In the ground state, the disorder, uncertainty and particle movement is lowest and entropy decreases as $T \rightarrow 0\text{K}$.

The concept that entropy flows out of the system as a result of the transformation of energy into information is well documented [Tribus and McIrvine, 1971; Jaynes, 1957; Szilard, 1929]. Extending this concept to the SE process, the requirements entropy can decrease and the order, stability and certainty of the requirements information can increase as engineering effort flows in and entropy flows out. Figure 3.3 illustrates this concept by comparing the behavior of particles in an open physical system with requirements that are treated as discrete information entities, or constituents, contained within the SE process. As engineering effort is input, the requirements become less random and more ordered over time, and this departure from randomness reflects the decrease in requirements entropy. Requirements engineering is an information transformation process. The *transition* from higher information uncertainty to lower information uncertainty occurs as the engineering effort *transforms* the requirements from a state of lower quality to higher quality.

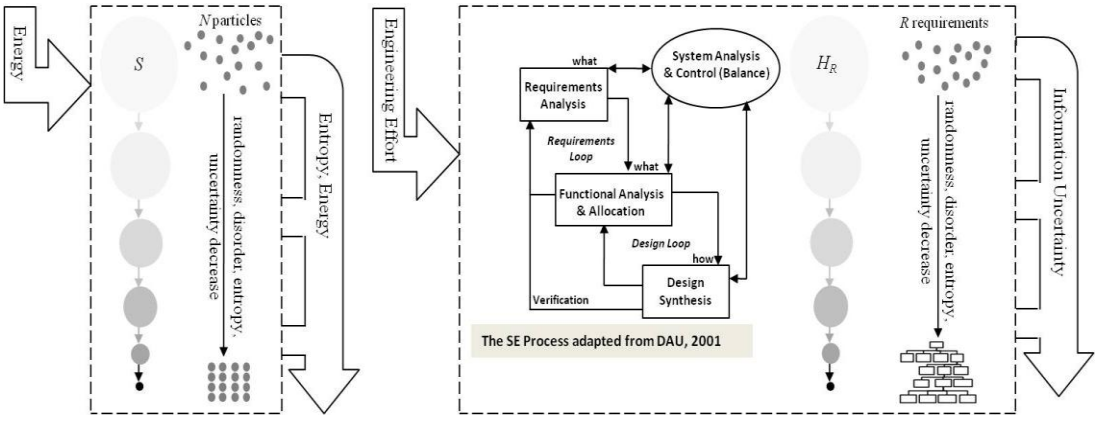


Figure 3.3: Illustration of 'partial similarities' between the SE process and an open physical system.

Information uncertainty, or entropy, is reduced to an acceptable level in this context when the requirements have reached a state of quality that instills sufficient confidence that the right system will be designed and built. The number of possible requirements distributions (or microstates), and therefore uncertainty, decreases as the requirements quality increases. For example, the number of possible poor requirements is much greater than the number of possible high quality requirements for the objective system of interest. "Disordered states are more probable than ordered ones" [Dincer and Yunus, 2001:125] and "the "entropy" of the state of a system is a measure of the probability of its occurrence. States of low probability have low entropy; states of high probability have high entropy" [Dincer and Yunus, 2001:125]. This is intuitive because there are many more choices that result in poor quality requirements than high quality requirements. Poor quality requirements have high entropy because their probability of occurrence is high and high quality requirements have low entropy because their probability of occurrence is low. Experienced requirements engineers know all too well that it takes considerable effort to transform poor quality requirements into high quality ones.

These 'partial similarities' led to the question: if the requirements engineering process can be treated as an open system and engineering effort is the energy that transforms requirements from initial states of high entropy to states of lower entropy, can a quantum statistical framework be developed that describes the macroscopic behavior observed during the requirements engineering process? Jaynes [1980] in his comprehensive paper on entropy emphasizes the applicability of predictive statistical mechanics in engineering and other fields based on its stochastic properties. Jaynes [1980] suggests the application of statistical mechanics can answer the question "(g)iven the partial information that we

do, in fact, have, what are the best predictions we can make of observable phenomena?" [Jaynes, 1980:593]. This dissertation attempts to bridge a theoretical gap by applying principles of information theory and statistical mechanics to the requirements engineering process in order to gain insight into important underlying properties influencing the behavior observed in requirements trends and engineering effort data.

3.4: Partial Similarities between SE and Information Theory

The concept of entropy was extended by Shannon [Shannon, 1948] to define a new measure of information that was independent of the means used to generate it [Tribus and McIrvine, 1971:179]. In information theory, the decrease in entropy is proportional to the increase in information certainty [Shannon, 1949; Schroeder, 2004]. The concept that entropy flows out of an open system as certainty of information increases is consistent with information theory [Shannon, 1948; Schroeder, 2004; Seife, 2007; Jaynes, 1957]. The Shannon entropy *H* in information theory describes uncertainty in the encoding, decoding, reproducing and transmitting of a *message* from a source to a destination via a channel with or without noise [Shannon, 1948; Shannon and Weaver, 1949] as illustrated in Figure 3.4. The *meaning* of the information is "irrelevant to the engineering problem" [Shannon, 1948:1] in Shannon's mathematical theory of communication, which is widely acknowledged as the origin of information theory.

The information entropy is generally expressed by:

$$H(\mathrm{X}) = -\sum_{i=1}^{n} p\,(\mathrm{x_i}) \log_2 p(\mathrm{x_i}) \qquad (6)$$

where:

$$\sum_{i=1}^{n} p(\mathrm{x_i}) = 1 \qquad (7)$$

and the entropy is in bits. For the case of two possible outcomes with probabilities p and q = 1-p, the entropy is:

$$H = -(p\log_2 p + q\log_2 q) \qquad (8)$$

Only the possibilities that are positive numbers are considered in the calculation of the information entropy [Shannon, 1948], not the possibilities equal to zero [Schroeder, 2004:399]. This will be an important constraint that is considered in the formulation of the REF as requirements increase in quality and certain states become unavailable to occupy and therefore have a zero probability of occurrence for the given process conditions at the time.

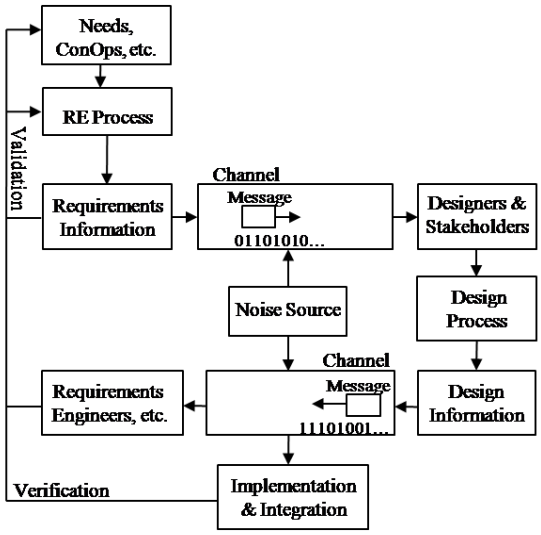


Figure 3.4: The Requirements Engineering Problem in the Context of Engineering Problem in Information Theory Addressed by Shannon [1948] in his Mathematical Theory of Communication (Adapted from Shannon, 1948; Shannon and Weaver, 1949; Schroeder, 2004).

The $H = 0$ when the outcome is certain, independent of whether or not one would consider the outcome to be successful. The meaning of the information contained in the message is "irrelevant to the engineering problem" in Shannon's information theory, however the meaning of the information contained in requirements is very important to the systems engineering problem. This fundamental difference may initially appear to exclude principles of information theory from being extended to the systems engineering problem. Given that the requirements engineering process develops and refines the information that enables the right system to be designed and built, a theory that explains requirements engineering behavior would have to account for the degree of meaningfulness of the requirements information for that purpose. Information theory focuses on the number of bits required to encode the message. Systems engineering is more concerned with requirements quality than the number of bits required to communicate the requirements information.

Information theory is concerned with the *reproduction* of source information and minimizing the probability of error or unwanted transformation during the transmission process. The information contained in the message is transformed into bits that represent the symbols in the original message. The stream of information bits is physical in nature during the process of transmission via a physical channel and there is uncertainty in the behavior of this communication process. Information theory permits the transmission of information with zero probability of error [Shannon, 1956].

Considering SE in the context of information theory (Figure 3.4), the requirements engineering process is concerned with the *production* of information while minimizing the probability of unwanted transformation or error during the process. Requirements

contain the information that *becomes the source* [Shannon, 1948], and requirements communicate the technical information needed to design and build a successful system. The systems engineer is interested in minimizing the uncertainty of the requirements information as much as possible before the system is designed and built. Requirements entropy should reflect this uncertainty relative to a desired end state that maximizes the probability of a successful outcome.

The requirements engineering process is transformative in nature where requirements evolve over time. Requirements changes uncover hidden requirements, delete unnecessary requirements, and ensure the requirements information is of sufficient quality to support system design [DAG, 2011:209]. The objective of the process is to minimize error while performing the transformations necessary to fully describe the system that is needed. A SE theory of information must therefore account for the transformative nature of the process, the meaning of the requirements information, the difference between good and bad changes, and the fact that "all changes are not equal [Lane and Boehm, 2007].

Another underlying principle of information theory that is consistent with the systems engineering problem is Shannon's observation that "the significant aspect is that the actual message is one *selected from a set of possible messages*" [Shannon, 1948:1]. This probabilistic aspect is generally applicable to the systems engineering problem given the astronomical number of possible requirements choices that are available to the engineer during the process. Can the requirements engineering effort can be viewed from the perspective that the significant aspect is the selection of the right set of requirements from a set of possible requirements? Can requirements quality attributes be defined to reflect

the meaningfulness of the requirements information and therefore bound the number of requirements quality possibilities?

In the information theory context, the requirements engineering process develops the information contained in the message that becomes the source before transmission. A destination for the requirements message is the system designer who makes tradeoffs and informed decisions based on the meaning of the requirements information. The requirements are the messages that drive, bound and inform the system design process after they are transmitted and received by the system design team. The design team's interpretation of the meaning of the requirements information is an important factor in the success of the design effort. The probability of success is a function of the degree to which the requirements contain all the meaningful information necessary for the right system to be designed and built. Requirements quality attributes can be defined for the purpose of verifying that the requirements communicate their intended meaning, their technical uncertainty has been reduced to an acceptable level and the risk of misinterpretation by knowledgeable recipients is minimal.

In systems engineering, we are more interested in how much additional information we need in order to design and build the right system the right way, rather than the reproduction of the source information that we currently have available to us. The requirements information can vary significantly before the start of the requirements engineering effort, depending on the technical uncertainty of the requirements information necessary to design and build the system. Rather than view the systems "engineering problem" from the communication perspective of Shannon's conduit metaphor illustrated in Figure 3.4, "we want to find an appropriate methodology to study

information as it is, not as it flows" [Schroeder, 2004:400]. More specifically in the SE case, we want a method to estimate, measure and predict the amount of additional information that is needed to design and build the right system the right way. Systems engineers need a methodology to determine how much engineering effort is necessary to gain the additional information needed to decrease uncertainty that the right system will be designed and built the right way. The goal is to minimize technical uncertainty and requirements entropy by increasing the requirements quality which, in the SE context, reflects the meaning of the requirements information. Principles of Shannon's information entropy are applicable to the SE problem even though SE is concerned with the formulation, analysis, interpretation and flow of information [Sage and Rouse, 2009:6]. If we consider the *H* as a function of probability for an event with two possible outcomes as determined by equation (8) and illustrated in Figure 3.5, uncertainty and H decrease as information increases. In this case, one bit of information is gained for an equiprobable event with 2 possible outcomes. The *H* of multiple events is additive [Shannon, 1948; Renyi 1961]. The *H* and uncertainty decrease as the information increases, and information increases as the number of choices or possibilities decrease.

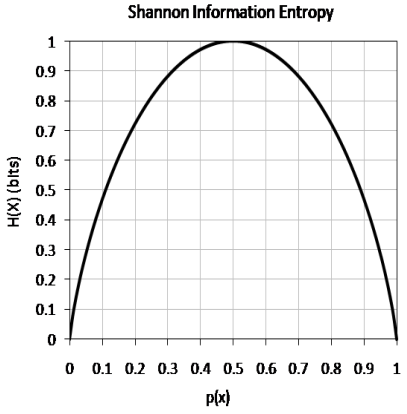


Figure 3.5: The Shannon [1948] information entropy.

The Shannon information entropy is the same for an event with two possible outcomes with a probability of occurrence of 10 percent and 90 percent. Extending this concept to the SE problem and assuming two possible outcomes of a requirement quality assessment where it is either determined to have a given quality attribute or not, it is not very satisfying to know that the entropy for a requirement with a 10 percent probability of having a quality attribute is the same as the entropy of a requirement with a 90 percent probability of having a quality attribute. A different form of entropy would be useful in order to discriminate the difference in technical uncertainty associated with 10 percent probability of requirement quality and 90 percent probability of requirement quality. It is important for the entropy to distinguish between a requirement with a 90 percent probability of a quality attribute and a requirement with a 10 percent probability of a quality attribute. The entropy in equations (6) and (8) would not distinguish between these two requirements. Can a different entropy equation be defined that sufficiently reflects the meaning of requirements information in terms of its quality attributes and distinguishes between requirements with high and low probabilities of quality?

Chapter 4: A Theory of Information Quality

4.1: Axioms of the Theory of Information Quality

In the Introduction of his seminal paper on a mathematical theory of communication, Shannon [1948] describes the fundamental "engineering problem" in the context of the reproduction of a message in telecommunication when it is transmitted from one place to another without consideration of the meaning of the message [Shannon, 1948]. Shannon's view that the meaning of the message is "irrelevant to the engineering

problem" [Shannon, 1948:1] is perfectly reasonable if one is interested in the reproduction of the symbols contained in a message during the transmission, or flow, of the information from one place to another. The theory of information quality introduced in this dissertation is intended to complement Shannon's theory by addressing a missing element, the *meaning of information*, in the context of the SE problem concerned with the formulation, analysis, interpretation and flow of information [Sage and Rouse, 2009:6] that *becomes the source* before it is encoded and transmitted. Information theory is concerned with the selection of a message from a set of possible messages. The theory of information quality extended to the SE problem is concerned with the development of the message, in the form of a set of requirements, and the number of possible quality distributions available for message.

The foundation for Shannon's work was laid nearly twenty years earlier by Nyquist [1924] and Hartley [1928]. Nyquist [1924] defined limits for the speed of transmitted information and described the efficiency of different types of waveforms [Nyquist, 1924], while Hartley [1928] discussed the amount and precision of information in terms of their symbols and their *meaning*. The possibilities for the information are reduced or eliminated as the symbols representing the information are received and their intended meaning is communicated. The mathematical theory of communication [Shannon, 1948] is now generally referred to as information theory even though its premise is to ignore the meaning of the information transmitted from one place to another. The Shannon model focuses on the reproduction of information from a source to a destination (Figure 3.4) and "the flow of information between these two points" [Schroder, 2004:398].

The entropy in the communication model measures information, choice and uncertainty [Shannon, 1948:11]. Entropy has also been described as a measure of the lack of information [Brillouin, 1951]. The different opinions about entropy are reconciled by Brissaud [2005] in his discussion of the concept in the fields of information theory, quantum mechanics, statistical thermodynamics and classical thermodynamics when he argues that entropy is appropriately associated with either information or the lack of information depending on the chosen frame of reference. This conclusion is consistent with the systems thinking that led to the theory introduced in this dissertation and the REF evaluated later. Brissaud [2005] makes another insightful observation when he suggests a stronger relationship between disorder and temperature than disorder and entropy [Brissaud, 2005:69]. This is consistent with the principles of the 2nd law of thermodynamics and the relationship described in the literature between entropy and information [Brillouin, 1951; Schroder, 2004], and opens interesting possibilities when one considers the implications of temperature in the 2nd and 3rd laws of thermodynamics.

The axioms of the theory of information quality extends these principles generally to information processes in which the objective is to develop information entities having a desired meaning for the intended user and the degree to which the desired meaning has been achieved can be determined by the entropy that reflects the cumulative state of information quality as defined by discrete quality attributes. This dissertation extends the theory of information quality to the special case of the requirements engineering problem in systems engineering. The axioms of the theory are summarized in Table 4.1. The equations can be generalized to address other forms of information entities, and the corresponding variables can be modified accordingly.

Table 4.1: Summary of the Theory of Information Quality Axioms and Principles

Special Case of Requirements Engineering in the REF
1. Information I and requirements entropy H_R are non-negative quantities, where $I(P) \geq 0$, $H_R(P) \geq 0$, and P is the number of possibilities corresponding to the distribution of R among the specified number of quality levels. I increases as H_R decreases, where $I(P=1)$ is maximum.
2. For a given number of requirements R where R occupy the desired end state, there is only one possible distribution and therefore $P = 1$. We get no additional information ΔI after R occupy the desired end state. The outcome uncertainty is minimum when $\Delta I(P=1) = 0$; $H_R(P=1) = 0 = H_{R\min}$.
3. Requirements quality is a good indicator of the status and progress of the requirements engineering effort relative to the desired end state because requirements quality is the objective of the requirements engineering process. $\forall\, r_j \in R$, the specified quality attributes q are necessary and sufficient for r_j to reach the desired end state when all attributes apply to r_j. There is no need to change r_j when it reaches the desired end state. Therefore, $r_j \rightarrow q_{14}$ is a stable condition.
4. H_R and uncertainty is maximum when randomness is maximum, which corresponds to R occupying the q_0. For a given R, the number of possibilities P is maximum and $H_{R\max} = R\log_2 P$ when R occupy the zero quality state and all quality levels are available to R to occupy. H_R measures the degrees of freedom or number of possible arrangements for R assuming quality attributes are not randomly exchanged between R.
5. H_R and uncertainty monotonically increase as the number of possible arrangements P increases and R increases. $H_R(R) > H_R(R\text{-}1)$ for a given quality level because there are more possibilities for R than R-1. Requirements distributions where $p(r_n) = 0$ are not included because requirements do not randomly redistribute among the quality levels.
6. Additivity: I and H_R are sums of the information and entropies, respectively, for the requirements from each quality level. I and H_R may increase, decrease or remain steady depending on the impact of changes on the requirements quality distribution.
7. H_R or uncertainty decrease as I increases where $I = H_{Rmax} - H_R$ is the information we have at any given time in the process and $\Delta I = H_R - H_{Rmin}$ is the additional information needed to reach the desired end state. As such, H_R is the additional information that is necessary to minimize uncertainty and is the lack of information from the perspective of the receiver.
8. H_R decreases as $Q \rightarrow$ max and $H_R = 0$ when $P = 1$, which corresponds to R reaching the desired end state. H_R of a *closed* process always increases (e.g. entropy begins to increase when R or system design are "frozen"). H_R of an *open* process may decrease locally as engineering effort E is input. If the process becomes closed, H_R increases over time as customer needs evolve, new technology becomes available and operational requirements change without input of E to maintain R in the desired end state.
9. Conservation of the cumulative requirements quality Q is assumed for estimations of P at a given time in the process with R in equilibrium; the input of E increases Q and I, and decreases P and H_R; the relationship between E, H_R and I is defined by the REF.
10. Q is a measure of the randomness of the requirements information. Randomness is the departure from order [25]. Disorder, randomness and uncertainty decrease as $H_R \rightarrow 0$ and $Q \rightarrow$ max. E is required to increase and control Q and I during the process, thereby decreasing H_R and uncertainty. The REF defines the relationship between E, I and Q in terms of the additional engineering effort ΔE necessary to transform R from their current state to the desired state.

4.2: Summary of Mathematical Foundation of the Theory

The mathematical foundation extended in this research is summarized in Table 4.2.

Table 4.2: Summary of Mathematical Foundations

Information Theory and Statistical Mechanics	Theory of Information Quality
$\mathrm{N} = \sum_j \mathrm{n}_j$	$R = \sum_{n=0}^{14} r_n$
$\mathrm{E} = \sum_j \mathrm{n}_j \varepsilon_j$	$Q = \sum_{n=0}^{14} r_n q_n$
$P_{BE} = \frac{(\mathrm{N}_1 + G - 1)!}{\mathrm{N}_1!\,(G-1)!}$	$P_{BE} = \frac{(r_n + N - 1)!}{r_n!\,(N-1)!}$
$dW = \sum_j \mathrm{n}_j d\varepsilon_j$	$dP = \sum_{n=0}^{14} r_n dq_n$
$S = k log_2 W$ $S_{phys} = -k \sum_{i=1}^{n} \mathrm{N}_i log_2 \frac{\mathrm{N}_i}{P_i G}$ $H = -\sum_{i=1}^{n} p_i \, log_2 p_i$	$H_{Rmax} = R log_2 P$ $H_R = -\sum_{n=0}^{14} r_n log_2 \frac{r_n}{P_n R}$
$I(p) = H_{max} - H(p)$	$I = H_{Rmax} - H_R$ $\Delta I = H_R - H_{min}$
$d\mathrm{E} = TdS$	$\Delta E \stackrel{\text{def}}{=} \frac{\Delta I Q}{< b >}$

Chapter 5: The Requirements Entropy Framework (REF)

The REF presented in this chapter is one of several developed for the purpose of implementing the theory of information quality. The REF described herein was selected first for evaluation because of its relative simplicity and mathematical implementation. There are many possible variants of the REF, however their basic theoretical construct is the same. All the REF variants treat requirements as discrete constituents contained within an open requirements engineering process where the input of engineering effort transitions the requirements from initial states of lower quality and higher uncertainty to states of higher quality and lower uncertainty. Further, the requirements quality macrostate reflects the cumulative requirements quality at any given time in the process and the distribution of the requirements among the available quality levels. The number of quality levels in the REF corresponds to the number of quality attributes determined to be necessary to reduce requirements information uncertainty to an acceptable level for design of a system solution that satisfies customer needs.

5.1: Quality Attributes

If information reduces uncertainty and entropy decreases as the uncertainty or number of possibilities decreases, requirements are the technical information "that define the capabilities, characteristics, or quality factors of a system" [Honour and Valerdi, 2006:6], then it stands to reason that the quality of the system definition increases and the entropy and uncertainty of the requirements information decreases as requirements quality increases. Customer needs are satisfied by system quality. Design quality enables system quality (e.g. quality must be designed into the system), and requirements quality

enables design quality. The requirements engineering effort underpins the system design and development effort.

The degree to which the requirements engineering effort reduces the uncertainty of a successful outcome corresponds to the degree to which the requirements engineering effort increases requirements quality. Requirements quality increases as the number of their quality attributes increases. The requirements engineering effort reduces the technical uncertainty that a quality system will be designed, built and delivered to satisfy customer needs, and the cumulative requirements quality should reflect the degree to which the requirements engineering effort reduces the technical uncertainty. Assessment of the status and progress of the requirements engineering effort without considering the state of requirements quality or technical uncertainty would be expected to provide only partial information at best. The strength of correlation between the Fractional Method and a method that measures requirements quality is expected to be relatively weak.

The following table defines the quality attributes used to implement the theory in the REF described in this dissertation based on Davis [1993] and INCOSE [2011].

Table 5.1: REF Quality Attributes and Definitions

Requirement Quality Attribute	Definition
1. Realistic	The requirement can be met within the cost and schedule constraints of the program.
2. Correct	The requirement represents something necessary for the system to be built and satisfaction of some need.
3. Understandable	The stakeholders can easily comprehend the meaning of the requirement with minimal explanation.
4. Complete	The requirement captures everything that is intended with no TBDs or TBRs.
5. Verifiable	The requirement is written in such a way that enables cost effective verification that the system fulfills it.
6. Validated	The requirement is necessary to build the right system and meet user/customer needs.

7. Traced/Traceable	The requirement is allocated and linked to all other supporting requirements at higher and lower levels.
8. Unambiguous	The requirement has only one possible interpretation.
9. Consistent	The requirement is not in conflict with any other requirement.
10. Precise	The requirement numeric quantities are accurate and specified to an appropriate level.
11. Concise	The requirement communicates the information using as few words as possible.
12. Design Independent	The requirement enables flexibility in the design process by defining what is necessary, not how it should be designed or implemented.
13. Annotated	The requirement necessity and relative importance is documented.
14. Not Redundant	The requirement information is not stated more than once without a good reason to do so.

5.2: Theoretical Construct

The REF used to obtain the results presented in this dissertation is illustrated in Figure 5.1.[1] The REF is analogous to an open physical system in which the entropy is defined by the log of the number of microstates [Boltzmann, 1896]. Each requirement is treated as an information entity that occupies a discrete level corresponding to its cumulative quality at any given time in the requirements engineering process. The *R* may occupy any of the $N = 15$ levels having one state per level in this case. *N* corresponds to the number of quality attributes determined by the SE team to represent the desired requirement quality end state plus one additional state representing zero quality q_0. The REF is completely flexible in this regard and can reflect the observations and experience of the SE team in terms of the *N* that best meets the need of the program. The *number of possibilities P* increases as *N* and *R* increase.

[1] The REF described herein and variants thereof are patent pending under patent number 61824415. Inquiries can be made to the author at grennm@gwu.edu.

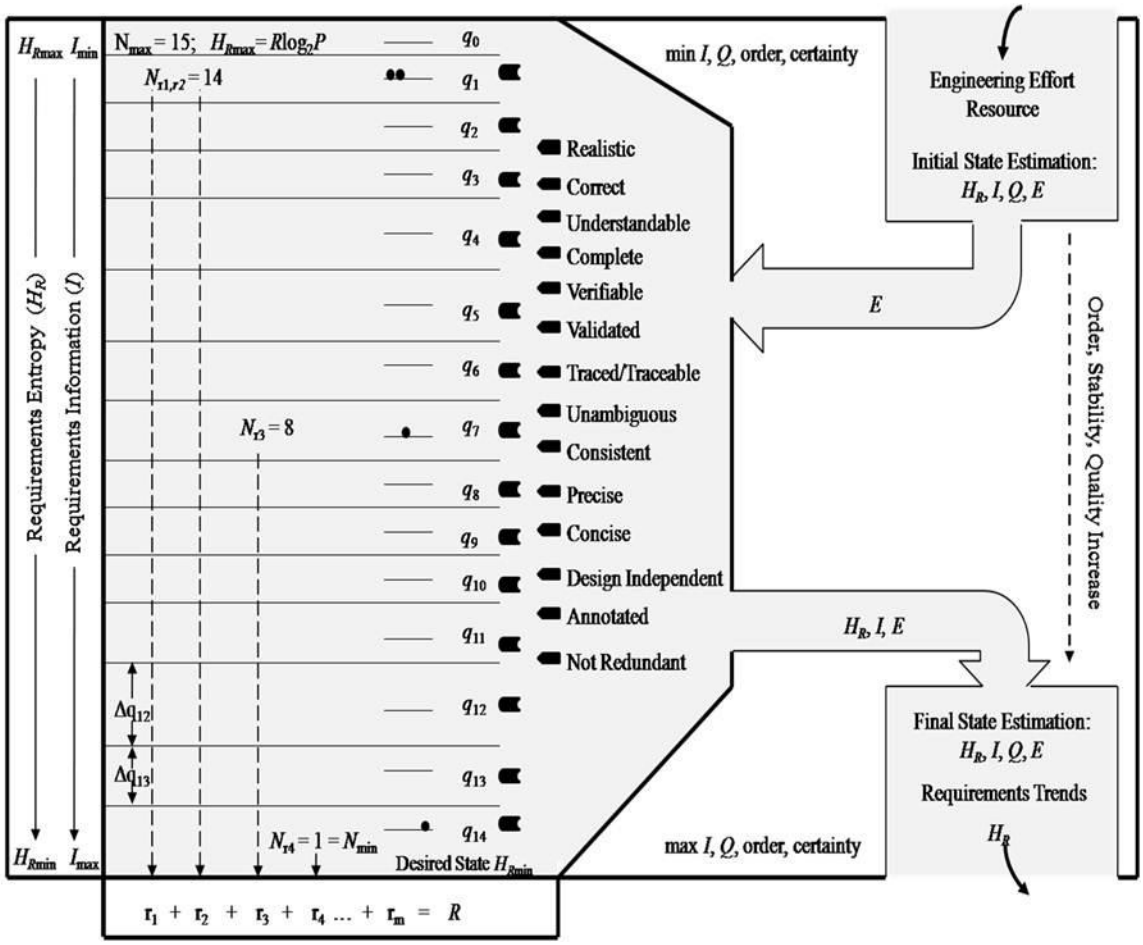


Figure 5.1: The REF treats the *R* as information entities that transition from initial high probability, high entropy states to low probability, low entropy states as engineering effort is input and *R* accumulate quality.

Requirements in the REF occupy quality levels corresponding to their accumulated quality, and the requirements entropy reflects the number of ways that the requirements can be configured among the available states. Higher quality levels become available and more probable for requirements to occupy in the REF as the cumulative requirements quality *Q* increases. This principle is analogous to higher energy levels that become available and more probable for particles to occupy in a physical system as the total energy of the system increases. For the purpose of this research, we assume that

requirements at a higher level of quality are not allowed to reconfigure themselves to lower quality levels randomly. This may occur, however, if a new requirement is added or an existing requirement is revised such that it has a negative impact on other existing requirements, such as if a key performance parameter is increased and becomes more difficult to achieve and has a negative ripple effect on other requirements. This effect would be expressed in the entropy when the calculation is redone because of its impact on the requirements quality distribution.

The REF illustrated in Figure 5.1 shows four requirements: r_1 and r_2 occupy q_1 corresponding to *any* one bound quality factor representing relatively low quality requirements and high entropy, r_3 occupies q_7 corresponding to *any* seven bound quality factors and r_4 occupies the desired end state q_{14} corresponding to maximum quality with *all* fourteen quality factors bound to it and minimum requirements entropy $H_{R\min}$. There is no constraint on the order in which the quality factors bind to the requirements.

For the purpose of this research, the levels in the REF are assumed to be discrete and nondegenerate with negligible interactions because there is only one state per quality level available for requirements to occupy, and any given *existing* requirement occupying a given quality level has only the current level and higher levels available to it. This is important in our analysis because we are more interested in the estimation and forecasting of the effort to transform R from its current state of requirements entropy H_R, which reflects the current distribution of R among the quality levels, to the desired state $H_{R\min}$ or some other improved interim state. The H_R becomes an indicator of the status and progress of the engineering effort if we assume that each requirement in the set of R must be a good requirement that enables the right system to be designed and built the

right way and each good requirement will have the specified quality attributes. In this dissertation, the engineering effort is assumed to increase quality and lower H_R of existing requirements. However, it is more likely that new requirements added in the future will be distributed among lower quality levels than higher quality levels because they have not yet been subjected to the requirements engineering process. Therefore, the number of quality levels available to a given existing requirement decreases as its quality increases. The implementation of the REF in this paper restricts $H_{R\min}$ to the desired end state only and enables the status and progress of the requirements engineering effort to be quantified relative to $H_{R\min}$.

A requirement r_j may occupy any quality level q_n corresponding to the number of quality attributes 'bound' to it. The initial state can range from zero quality when R occupy the minimum quality state or $R \rightarrow q_0$, to maximum quality when $R \rightarrow q_{14}$ prior to the input of requirements engineering effort E, however unlikely the latter condition may be. Requirements occupying q_{14} have reached the desired end state because all quality attributes are bound to it.

The q_{14} state in the REF has the highest order, stability and certainty. Order is "departure from randomness" [Bailey, 1987:90], and q_{14} is the least random state because there is only one possibility for any requirements occupying it. This is analogous to the ground state of a physical system in which there is only one microstate available where W in equation (2) is one, the entropy S is zero and uncertainty is minimum. The q_{14} state in the REF has the highest certainty because there is only one possible configuration for requirements occupying it. Requirements occupying the q_{14} state are least likely to change and are therefore the most stable because they exhibit maximum quality and there

is little need for the SE team to change them. The state least likely to change is when requirements are ready to support system design and have been formally reviewed and determined to be in such a state, which is a function of the requirements quality.

Requirements *stability* corresponds to a static condition where the macrostate is not likely to change, which is the least probable and most desirable q_{14} state where $H_R = 0$, or a dynamic condition where the macrostate is changing in a controlled and predictable way. For example, static stability can occur for a period of time when *E* is no longer input into the process and the requirements are "frozen", which may occur at a critical design review. In effect, the process becomes closed, or isolated, at this point and the requirements information begins the process of tending toward maximum entropy over time as customer needs change, technology advances and demand for more capable systems increases. As a result, the natural tendency is for the quality of the requirements information to decrease and the requirements entropy to increase over time. The input of engineering effort into the process counteracts this natural tendency toward maximum entropy by continuously updating the information to reflect what the customer needs. A steady state of requirements quality can be maintained with the input of engineering effort.

Like Shannon [1948], the REF only considers the possibilities that are positive numbers in the calculation of entropy, not the possibilities equal to zero [Schroeder, 2004:399] as requirements increase in quality. The probability of a lower quality level being randomly available to requirements at a higher quality level is zero unless requirements changes are made or new requirements are added during the engineering effort that would result in such movement to lower quality levels. For example,

discovery of technical errors or negative "ripple effects" from other requirements could result in such movement. However, this would occur as part of the engineering effort and not randomly.

5.3: Mathematical Foundation of the REF

Quantum statistics are used in the REF to estimate the number of possibilities given that the total number of requirements *R* is the sum of the requirements occupying each quality level n from zero to 14 where:

$$R = \sum_{n=0}^{14} r_n = r_0 + r_1 + \cdots + r_{14} \qquad (9)$$

and the total requirements quality *Q* is defined by:

$$Q = \sum_{n=0}^{14} r_n\, q_n = r_0 q_0 + r_1 q_1 + \cdots + r_{14} q_{14} \qquad (10)$$

where any number of requirements *r* in the set of *R* can occupy any quality level q_n available to it and r_n is the number of requirements that occupy the given quality level. The *Q* represents the cumulative quality *macrostate*, and the possible distributions of *R* among the available *N* represent the corresponding *microstates*. Equations (9) and (10) are similar in form to the equations that must be satisfied by elementary particles in a one-dimensional energy space in quantum statistical mechanics [Richtmyer et al, 1969:533; Hill, 1960:8]. These principles are extended to the REF by substituting elementary particles that occupy energy levels with requirements that occupy quality

levels. Bose-Einstein quantum statistics are used to estimate the number of ways, or possible distributions, in which *R* can be allocated among the available quality levels.

The number of possible microstates available for requirements to occupy is estimated using Stirling's approximation:

$$P_{SA} = e^{(r_n+N-1)\ln(r_n+N-1)-(N-1)\ln(N-1)-r_n \ln r_n} \quad (11)$$

of the Bose-Einstein distribution:

$$P_{BE} = \frac{(R+N-1)!}{R!(N-1)!} \quad (12)$$

where $R \lesseqgtr N$, and N is the number of *higher* states available for any given r_n in the set of R to occupy including their current state. For example, if $r_0 = R = 4$ and all levels are available for R to occupy because $R \rightarrow q_0$ and $N = 15$, then there are:

$$P_{BE} = (4+15-1)!/4!(15-1)! = 3060$$

possible ways to arrange the R among the $N = 15$ quality levels available to them.

For $R = 1000$ under the same conditions, the number of possibilities $P_{BE} \approx 4.47\text{x}10^{2609}/3.49\text{x}10^{2578} = 1.28\text{x}10^{31}$. The number of possibilities can be used to express complexity [Campbell, 1988]. This example provides some insight into the number of possibilities encountered during the requirements engineering process and the potential utility of quantum statistics for describing this complexity.

The requirements entropy H_R in bits is estimated by extending the physical entropy equation suggested by Brillouin [1951] to our requirements engineering problem, resulting in:

$$H_R = -r_0 log_2 \frac{r_0}{P_0 R} - \cdots - r_{14} log_2 \frac{r_{14}}{P_{14} R} = -\sum_{n=0}^{14} r_n log_2 \frac{r_n}{P_n R} \quad (13)$$

and the maximum requirements entropy $H_{R\max}$ can be estimated using:

$$H_{R\max} = R\log_2 P \quad (14)$$

$H_{R\max}$ for the Maxwell-Boltzmann distribution P_{MB} is included for reference in the estimations of requirements engineering effort presented in Section 4, however simulations using P_{MB} quantum statistics are not included in this paper due to computational limitations encountered in the MATLAB® [Mathworks, 2010] implementation of the REF for $R > 330$. Additivity is among the properties of entropy [Brillouin, 1951; Wehrl, 1978] and this enables continuous measurement and monitoring of H_R during the requirements engineering process. For the simple example illustrated in Figure 5.1, $H_{R\max} = 46.3$ and $H_R = 22.43$. To put this in context, $H_{R\max} = 103{,}336$ for $R = 1000$. The $H_R(t)$ increases, decreases or remains steady depending on the impact of additions, deletions or revisions on the requirements quality distribution.

The requirements engineering effort E is the energy that *transforms* the requirements from a lower state of quality to the desired state during the requirements engineering process. The estimated average increase in E converted to time units (e.g. person-hours)

is defined as:

$$\langle \Delta E \rangle \stackrel{\text{def}}{=} \Delta I Q / \langle b \rangle \qquad (15)$$

where ΔI in bits is the increase in requirements information necessary for the R at the given time to reach the desired state, Q is the corresponding requirements quality defined by equation (4), and $\langle b \rangle$ is the average information output rate for ΔI in bits per unit time. The ΔI to reach the desired end state is calculated by extending the method suggested by Schroeder [2004]:

$$\Delta I = H_R - H_{R\min} \qquad (16)$$

where H_R is defined by equation (7). Schroeder [2004] takes the perspective of the system or the sender of the message where entropy represents the *information* the system or sender currently has and is calculated using, in effect, $H_{R\max} - H_R$ when considered in the context of this work. The REF takes the perspective of the observer outside the system or the receiver of the message where entropy represents the *lack of information*. This lack of information is the additional information that is necessary to reduce technical uncertainty to a minimum and achieve the desired state of information quality for the right system to be built the right way. This lack of information is calculated using equation (10). The initial requirements entropy H_{Ri} can be used to estimate the initial information I_i using:

$$I_i = H_{R\max} - H_{Ri} \qquad (17)$$

where $H_R = H_{Ri}$ at the start of the effort. This may be useful to quantify the quality of the source requirements at the start of the program and gain insight into the effectiveness of early requirements efforts. For cases where $R \rightarrow q_0$ represents the initial state and $R \rightarrow q_{14}$ represents the desired state, ΔI is maximum and $I_i = 0$. In such extreme cases, $H_R = H_{Ri} = H_{R\max}$. For SE programs with an initial requirement quality distribution other than $R \rightarrow q_0$, H_{Ri} decreases as the initial requirements distribution increases in quality, resulting in less ΔE required to transition R from the initial state to the desired state. The difference in initial quality between new requirements and reused requirements that have previously been developed in the requirements engineering process can be accounted for in the REF by assuming a higher probability that a quality attribute will be bound to r_j at the start of the effort if r_j is reused rather than new. The H_R may increase, decrease or remain steady depending on the impact of r_j on H_R when r_j is added, deleted or revised.

5.4: Theoretical Requirements Entropy Behavior

The maximum requirements entropy H_R calculated using equation (13) as a function of R where R occupy each of the available N levels is shown in Figure 5.2. The entropy behavior shown in Figure 5.2 resembles the entropies described by Curado and Nobre [2003] and Kapur [1986] in their analysis of alternative forms of entropy. The H_R monotonically increases from $H_R = 0$ when R occupy q_{14} and $N = 1$ to $H_{R\max}$ when R occupy q_0 and $N = 15$.

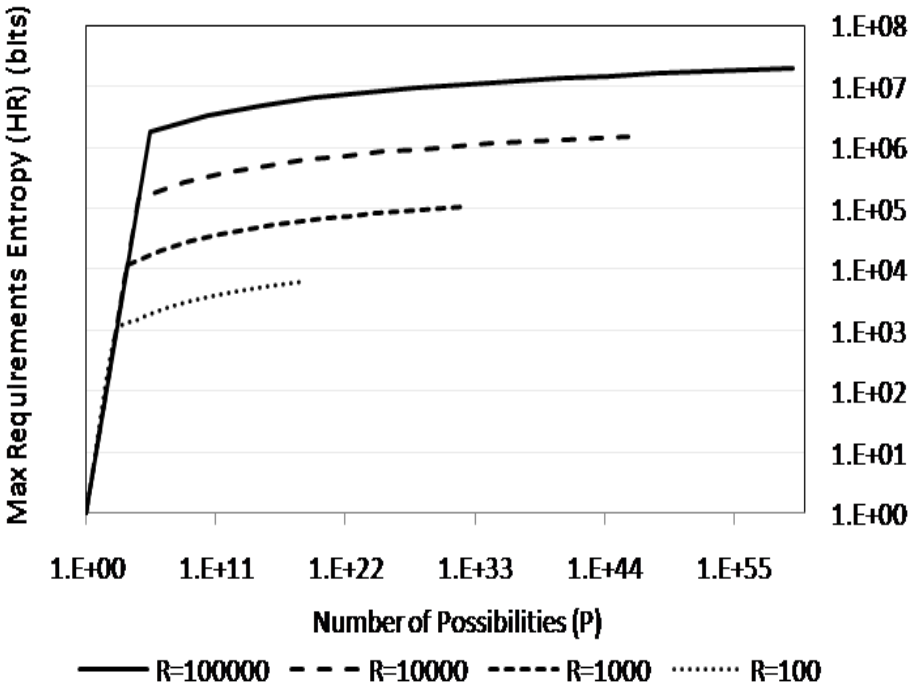


Figure 5.2: Requirements entropy for all *R* in each level from N_{max} to N_{min}.

Figure 5.3 plots this condition of H_R with the x-axis reversed to illustrate the possible behavior of *R* over time if all the requirements were to transition from q_0 to q_{14} in single level increments. This is a theoretical case that is highly unlikely to occur in practice, however the simulation results presented in Chapter 7 show similar behavior as the mean quality of the requirements distribution increases over time.

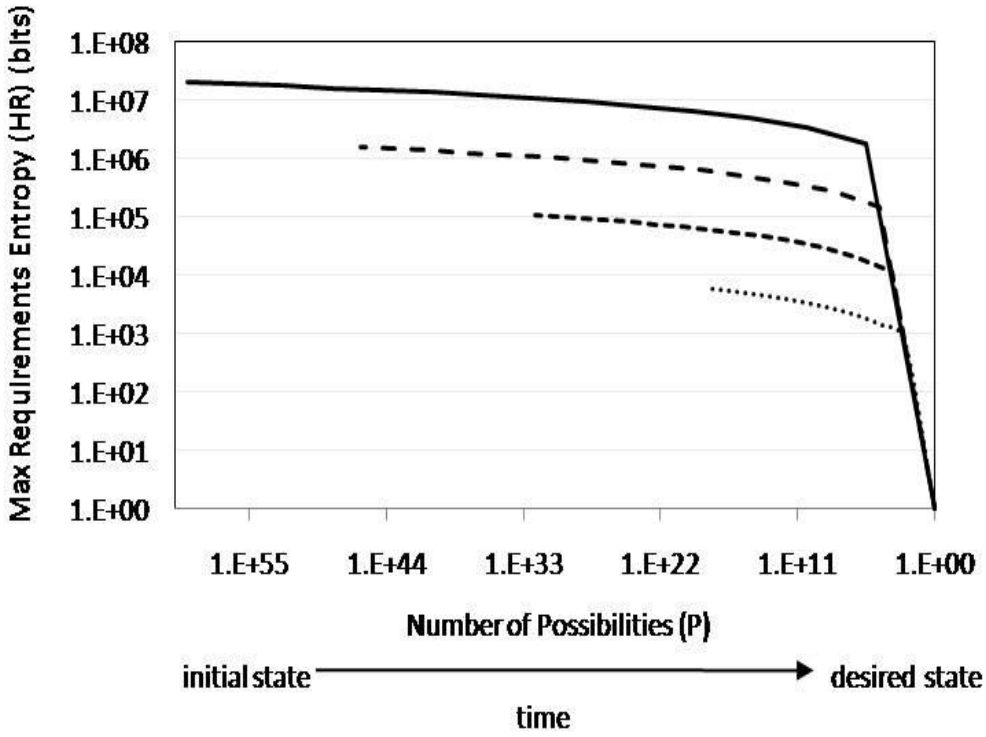


Figure 5.3: Requirements entropy with x axis reversed to illustrate behavior of all *R* transitioning from N_{max} to each discrete level toward N_{min} over time.

5.5: Example Requirements Quality Distributions

Figure 5.4 shows two examples of initial state requirement quality distributions that were generated during the MATLAB® [Mathsoft, 2010] simulations of the REF. The top distribution reflects a relatively low initial state quality with the majority of requirements occupying either the zero quality level q_0 or the q_1 level corresponding to any one quality attribute. The middle distribution reflects a relatively high initial state requirements quality distribution with a much higher initial state *Q* and the most populated states being either the maximum quality level q_{14} or the q_{13} level just below it. Reused requirements

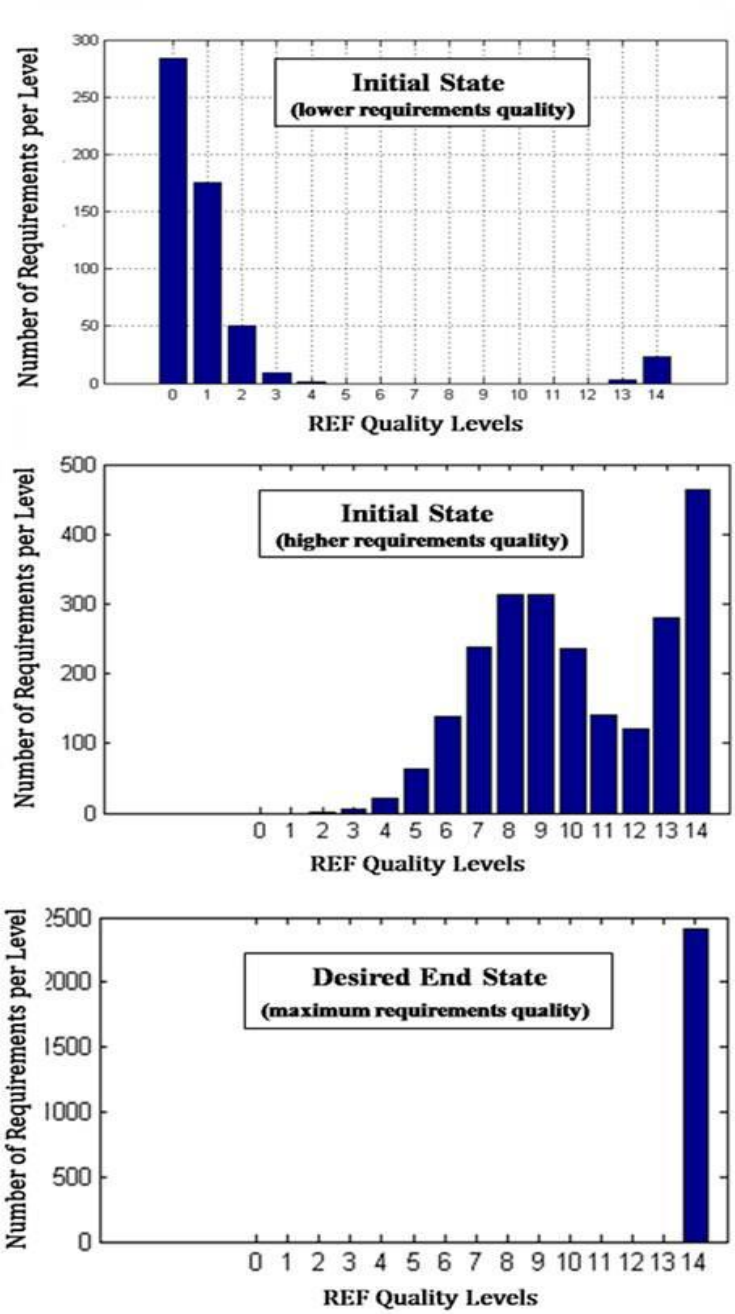


Figure 5.4: Examples of initial state requirements quality distributions (top two plots) and desired end state requirements quality distribution (bottom).

in these simulated SE program initial state requirements quality distributions have a much higher probability of occupying the q_{13} or q_{14} levels than new requirements that have not yet been subjected to a rigorous requirements engineering process. For example, a

ground-based multi-sensor system with new requirements for imaging and C^3 capabilities but reuse of existing requirements for other elements of the system that have little impact on the new capabilities could exhibit an initial state requirement quality distribution similar to the top distribution. The middle distribution shows a higher initial state quality distribution and suggests a higher degree of reuse given that a majority of the requirements occupy q_{14} at the start of the program. The requirements quality distribution shown in the bottom plot corresponds to the desired end state where all R from the middle initial state distribution occupy q_{14}.

Figure 5.5 shows two examples of initial state requirements quality distributions with varying degrees of requirements reuse, where the top distribution reflects low to moderate reuse and the bottom distribution reflects a higher degree of requirements reuse. Higher requirements reuse is a reasonable indicator of previous SE experience with one or more similar systems, otherwise the requirements would not be appropriate for reuse. Given this experience, the higher degree of requirements reuse is assumed to have a positive impact on the initial state requirements quality distribution of the *new* requirements relative to the initial state requirements quality distribution of new requirements in SE programs with little to no experience with the system under development.

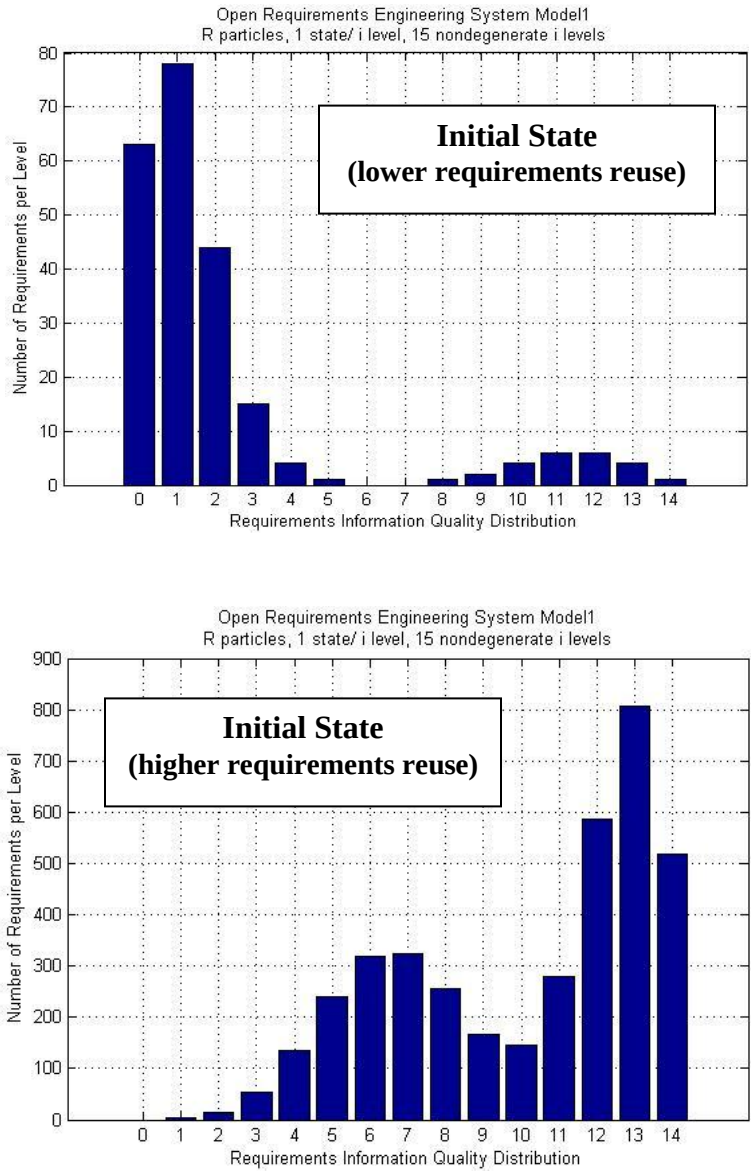


Figure 5.5: Examples of initial state requirements quality distributions with relatively low requirements reuse (top) and high requirements reuse (bottom).

5.6: REF Model Context Diagram

A context diagram of the REF with the model inputs, activities, controls, enablers and outputs is shown in Figure 5.6. The input parameters for the REF can be manually entered based on measured or expected values or estimated by random variable simulations using probability distributions that approximate the observed behavior. For example, the number of requirements *R* can be manually entered if known at a given time in the program.

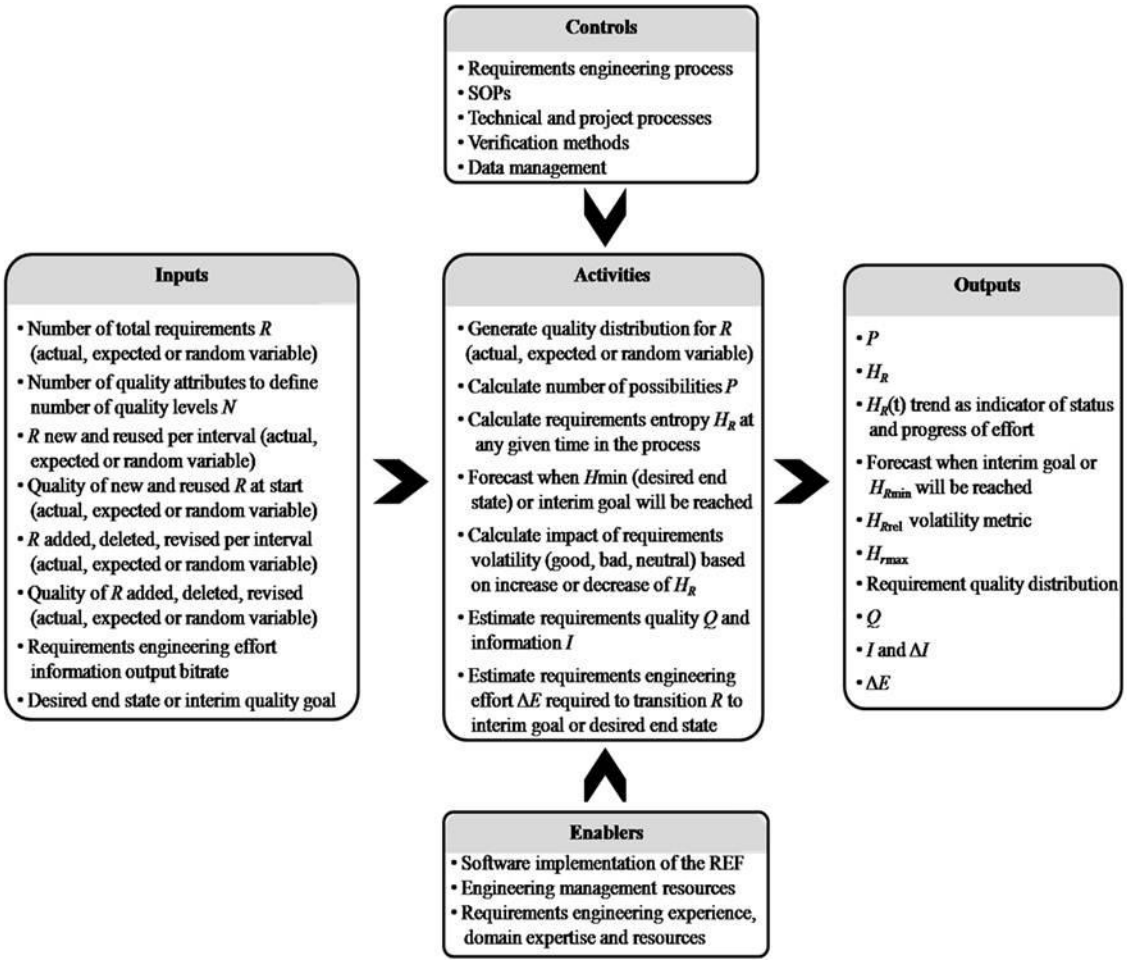


Figure 5.6: Context diagram of the REF modeling process.

If the REF is being used to forecast, then the expected R can be entered into the model. The R can also be estimated in the REF using a normally distributed random variable with a specified $\bar{R}$ and standard deviation. Similarly, deleted requirements can be manually entered if known or, using the Rhodes et al [2009] and Costello and Liu [1995] data as a guide, estimated by a uniform random variable. The controls facilitate the effective integration and implementation of the REF in the organization's requirements engineering process. As with any new technology or tool, it is important to define the requirements and procedures for its effective use in the organization and demonstrate its value in the process to gain organizational buy-in.

5.7: Key Assumptions in MATLAB® Implementation of the REF

Assumptions were necessary to write the software code for the MATLAB® [Mathworks, 2010] implementation of the REF and run the simulation experiments for the special case described in this dissertation. Most of the assumptions do not represent limitations of the REF, but rather constraints that were imposed in the software implementation evaluated here. Assumptions annotated with an asterisk (*) are fundamental. Many key assumptions are summarized here to provide additional insight into the simulation results.

1. The requirements engineering process is an open system (*).

2. The open requirements engineering process does not violate the laws of nature (*).

3. Measurements of engineering effort and requirements trends are observations of the macroscopic behavior of the requirements engineering process (*).

4. Good requirements enable the development of quality systems, and good requirements exhibit quality attributes determined by the SE team to be important for quality systems to be designed and built (*).

5. The cumulative quality of requirements being worked in the requirements engineering process is a good indicator of the status and progress of the requirements engineering effort (*).

6. Neither the quality of an individual requirement r_j nor the cumulative requirements quality Q increase without the input of engineering effort into the process (*).

7. Not all requirements quality distributions in the REF are equally probable. A requirement will be more likely to occupy a low quality state than a high quality state without the input of engineering effort (*).

8. The q_0 corresponds to maximum uncertainty, randomness and entropy where $H_{R\max} = R\log_2 P$ (*)

9. The q_{14} corresponds to minimum uncertainty and entropy, and maximum departure from randomness, corresponding to the lowest probability state where $H_{R\min} = 0$. The

REF illustrated herein specifies fifteen (15) quality levels, however the number of quality attributes and therefore the number of quality levels may be tailored by the SE team based on experience with similar systems to best meet the needs of the program.

8. Nondegeneracy is assumed in the REF such that if $R = 1$ and R occupies q_1, then $N = 14$ in the calculation of P_{BE} because R would have $N = 14$ possible states available for it to occupy. There is only one equiprobable state per quality level and the *existing* requirements do not randomly exchange quality or reconfigure themselves into other quality distributions without the input of engineering effort. If R occupy q_{14}, then $N = 1$ and $P = 1$ because there is only 1 state available for R to occupy, which is the maximum quality state, and there is only 1 possible distribution of R when all R occupy the maximum quality state. This state corresponds to $H_{R\min} = 0$ because entropy and information uncertainty are minimum, and this only occurs when $R \rightarrow q_{14}$. The REF distinguishes between the information we have at any given time in the process and the additional information we need in order to reach the desired end state defined by the SE and program management team.

9. R may vary as a function of time as requirements are added, derived, uncovered, deleted, etc. The Q varies as a function of the cumulative quality of R according to equation 4. Equations (9) and (10) are consistent with principles of quantum mechanics for an *equilibrium* state. The REF assumes that these equations apply [Richtmyer et al, 1969:533].

10. At the start of the requirements engineering process t=0, all requirements are either new or reused where $R = r_{reuse} + r_{new}$ and the requirements volatility calculated using the Fractional Method equals one at the start because $R = r_a$, and r_r and r_d are zero.

11. The initial state quality of $r_{reuse} >> r_{new}$ given the experience with r_{reuse} for similar systems. New requirements have not yet been verified or validated using a rigorous requirements engineering process. An empirical study on the impact of requirements reuse on software defect density and volatility suggests that there is a significant difference between the number of defects and modifications required for non-reused requirements than reused requirements [Mohagheghi et al, 2004; Mohagheghi and Conradi, 2008). Requirements reuse has been shown to decrease the negative effects of volatility and increase the quality of software [Mohagheghi and Conradi, 2004] but caution should be exercised when reusing requirements [Botta et al, 2006]. These results are consistent with the author's experience and are extended to the REF MATLAB® [Mathworks, 2010] implementation.

12. Any requirement that transitions to the maximum requirement quality state q_{14} remains there for the duration of the requirements engineering effort and is therefore *stable*. The requirement may be deleted if determined to be unnecessary, but $p(r_{revise}) = 0$. Revisions are limited to R that occupy requirement quality states other than q_{14}.

13. The probability that a given requirement r_j will increase in quality as a result of engineering effort >> than the probability that r_j will decrease in quality. The simulation

results shown in this paper permit *existing* requirements to only increase in quality as a result of the requirements engineering effort. This assumption, in effect, does not account for the possibility of a mistake or a detrimental change to an existing requirement.

14. For the simulations in this paper, the probability of transition for any given requirement r_j is higher for the next quality level than other higher quality levels, and the probability of multiple transitions decreases as the number of quality levels increases. This constraint stems from the assumption that it is unlikely to transform a requirement from low quality to high quality in a single iteration.

15. Revisions are more likely for r_{new} than r_{reuse}, and revisions lead to more r_{new}. Ferreira et al [2009] found revisions led to r_{new} and an increase in project scope 78% of the time [Ferreira et al, 2009]. These findings are consistent with the author's experience and are extended to the REF.

16. Reused requirements are included in the requirements engineering effort as a result of their direct applicability to the objective system. An improperly reused requirement can be more costly than a poor new one. It may be easier to discard a new requirement than a reused requirement if there is bias toward a particular design approach or technology solution. This bias was not considered in the simulation results shown in this paper.

17. There is no difference between the probability of r_{new} quality at the start of the requirements engineering effort versus later in the effort. The Rhodes et al [2009] and Costello and Liu [1995] requirements trend data suggest a binomial probability distribution for $p(r_{new})$ and a uniform random variable with a maximum $p(r_d) = 0.1$. Revised requirements are modeled using a normally distributed random variable, with varying $p(r_{re})$ and $p(r_{new})$ for cases of low and high number of changes per interval.

18. The probability of r_{reuse} is higher for large SE programs (e.g. $R > 1000$) than small SE programs because the likelihood of starting from scratch on less complex systems is higher than on highly complex systems and SoS, which tend to evolve [Sheard, 2009]. The availability of requirements from earlier complex systems increase $p(r_{reuse})$ for these programs.

19. This paper assumes that requirements quality is a good indicator of the status and progress of the requirements engineering effort and the following fourteen (14) requirement quality attributes based on Davis [1993] and INCOSE [2011] are necessary and sufficient to achieve the desired requirements quality end state in the REF.

20. *Distinguishability* between requirements within any given quality level is not critical because requirements at q_{10} require an equivalent amount of ΔE to reach q_{14} even though they may not share the same ten quality factors, so the Bose-Einstein distribution is used. In practice, since the requirements engineering effort needs to distinguish between these two requirements to ensure they are appropriately worked, Maxwell-Boltzmann statistics

would account for the increased number of possibilities P in the H_R calculation if such treatment is necessary.

21. Although the entropy of a closed system can only increase, the entropy of an open requirements engineering process can decrease when energy, which in our case is in the form of engineering effort, is transported across its boundary [Dincer et al, 2001:125] (*).

22. “Disordered states are more probable than ordered ones” [Dincer et al, 2001:125] and “the "entropy" of the state of a system is a measure of the probability of its occurrence. Accordingly, not all possible requirements quality configurations, or distributions, in the REF are equally probable. States of low probability have low entropy; states of high probability have high entropy” [Dincer et al, 2001:125]. This is consistent with the assumptions that a requirement will be more likely to occupy a low quality state than a high quality state, and q_{14} is the lowest probability and corresponds to $H_{R\min} = 0$ (*).

Chapter 6: Research Approach

6.1: Research Objectives and Methods

The general research approach with top level objectives is shown in Figure 6.1. A systems engineering approach was taken to understand the problem and define the topic, and a considerable amount of time was allocated to the systems thinking necessary for this research. Many researchers define the research objectives after identifying a gap in the literature. Whenever a gap was identified in this research, it was considered in the context of the overall requirements engineering problem. In most cases, research

objectives addressing the gap would have made a contribution to the field in a relatively short period of time, however the fundamental problem would have remained unaddressed.

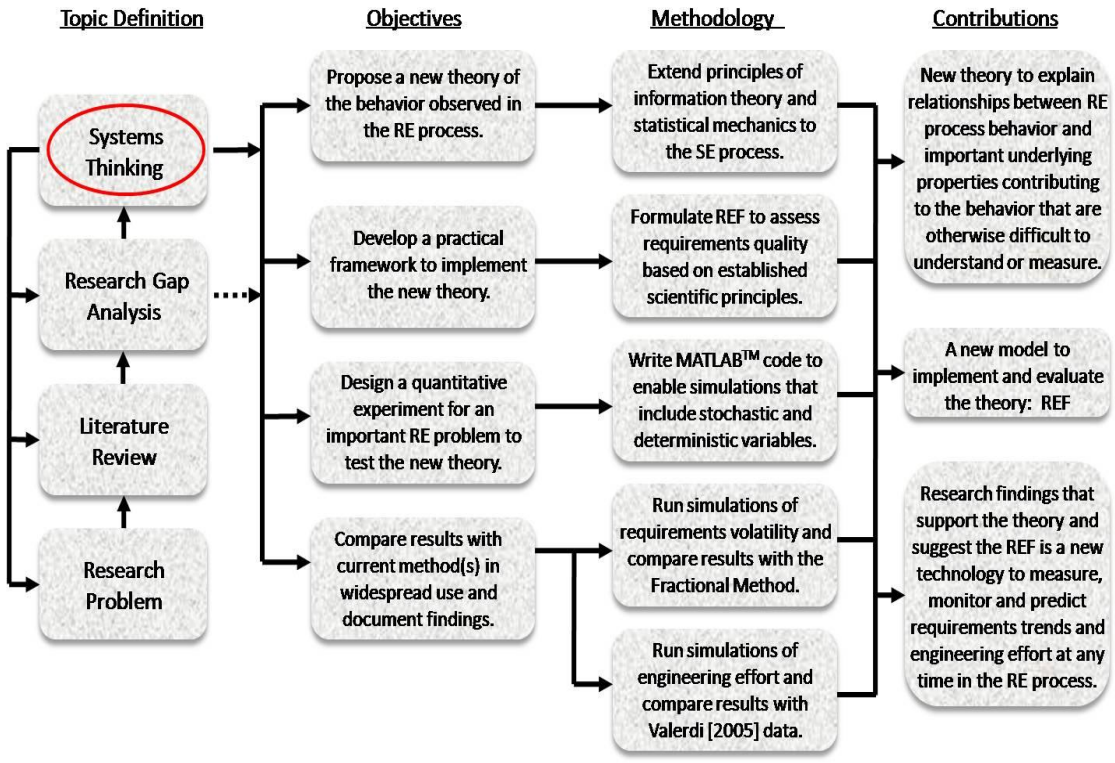


Figure 6.1: Summary of Research Approach

Analysis of the problem led to the conclusion that a new theory was needed to describe the macroscopic behavior observed during the requirements engineering process, and a new approach was needed to resolve the limitations of current methods used to measure requirements trends. Given the practical nature of the systems engineering problem and the realization that simply suggesting a new theory and a new method would not be enough to satisfy the engineering community, it was clear that the method would need to be evaluated to determine its potential utility. Given the schedule constraints imposed on the doctoral research phase and the time necessary to develop the theory and

the REF, the decision was made to review the existing SE literature for data on actual SE programs to determine if sufficient data were available to perform an acceptable evaluation.

After a thorough review of the literature, the engineering effort data measured by Valerdi [2005] on thirty-five (35) DoD programs was sufficient to perform simulation experiments of engineering effort based on the total number of requirements in the program. The requirements volatility data in the literature was extremely limited. This was a bit of a surprise given the importance of requirements engineering to the successful development of complex systems, the major problems experienced in complex system development programs, and the degree to which requirements volatility has been correlated with these problems. In any event, simulation experiments could be run to compare the forecasts of the REF with the Fractional Method and their strength of correlation to determine if the REF would be expected to provide significantly different insight into the status and progress of the requirements engineering effort as a function of requirements volatility. The Fractional Method measures the *quantity of changes* only with all the changes weighted equally, not their impact on quality or the degree to which negative "ripple effects" [INCOSE, 2011] of requirements volatility impact requirements quality or technical uncertainty. Requirements entropy as defined in the REF measures both the quantity of changes and the degree of their positive or negative impact on technical uncertainty reflected by the cumulative requirements quality. Given the fact that the REF accounts for requirements quality while the Fractional Method only accounts for the quantity of requirements changes, simulation experiments would be able to determine if there was a significant difference in their forecasts of when the

requirements would reach the desired state of order, certainty and stability such that additional changes were not needed and the requirements were ready to support system design.

6.2: Important Factors in REF Implementation of Theory

The relationships between requirements changes, requirements volatility, requirements quality, requirements entropy and engineering effort in the REF implementation of the probabilistic theory of information quality are summarized in Figure 6.2.

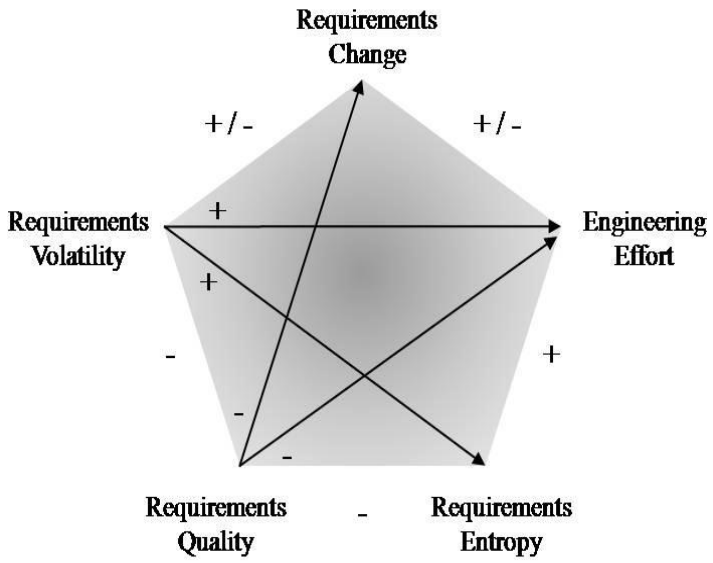


Figure 6.2: Top Level Model of Important Research Factors

The relationships suggest an interesting dynamic and positive reinforcing loop between requirements changes resulting in volatility that decreases requirements quality, which leads to more requirements changes. This dynamic is consistent with requirements churn that has been reported in the literature and observed in requirements engineering programs in which top level needs and requirements are continuously changing and the

objectives of the system are not well defined. It would be interesting to consider this relationship in a system dynamic model.

A more detailed representation of the relationships between important factors in the REF is provided in Figure 6.2. Also included in Figure 6.2 for reference are other relevant relationships and important factors suggested in the literature that are outside the scope of this research. Although this research does not include SE program success in the analysis directly, the relationships between requirements quality, requirements volatility and program success have been discussed previously.

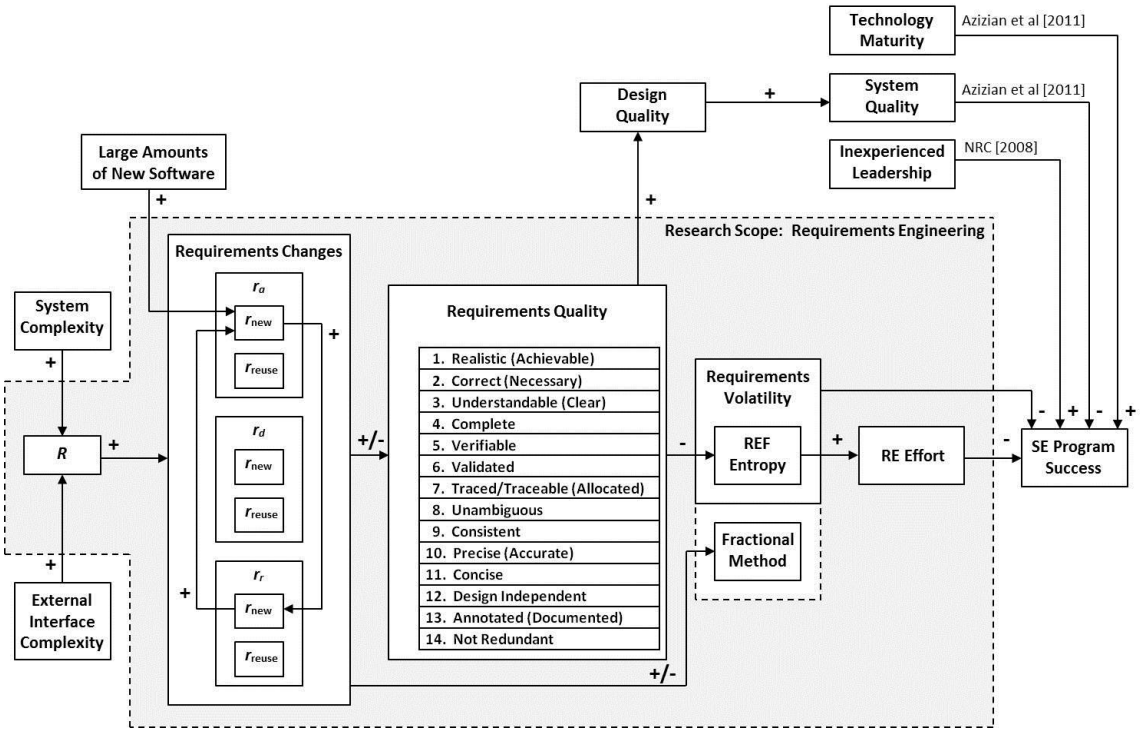


Figure 6.3: Model of Important Relationships in REF Implementation of Theory

The relationship between new and reused requirements and their impact on requirements quality is considered in the REF. The positive relationship between new requirements and requirements revisions is based on Ferreira et al [2009] and is consistent with researcher's experience in technology development programs.

6.3: Research Design

6.3.1: Research Premise

A major premise of this dissertation is that requirements changes can be good, bad or neutral depending on the degree to which additions, revisions and deletions have a positive or negative impact on the requirements entropy H_R. The REF described in Section 2 describes the relationship between H_R and requirements quality.

The objective of the requirements engineering effort is to define the objective system as efficiently as possible using a set of good requirements containing the information about what the objective system needs to do and how well it needs to do it in order to satisfy the customer. A good requirement will have the quality attributes determined to be important for the SE team to effectively document and communicate the meaning of the information necessary for the right system to be built the right way. Requirements determined to be in this state of quality reduce technical uncertainty and increase confidence that the design effort will result in a viable, near optimal system solution. Technical uncertainty increases risk of engineering effort underestimation, cost overruns and schedule delays. A requirements quality metric that includes information about the quantity of requirements changes would be a good indicator of the status and progress of the requirements engineering effort.

If requirements contain the information that enables the realization of a successful system, and requirements quality is an indicator of the information for this purpose [Hull et al, 2011], then it is reasonable to expect a precise and accurate metric to account for the number of requirements necessary to completely describe the system and the degree to which requirements changes impact requirements quality relative to the desired state.

6.3.2: Research Questions and Hypotheses

Given the significant impact of requirements quality on the overall success of the SE effort, would a requirements trend measurement method that accounts for the number of requirements changes and the degree to which the changes impact requirements quality be a more effective leading indicator of requirements trends than a method that does not?

The first research question is:

- Can the REF improve the forecast of requirements trends relative to the Fractional Method for requirements engineering efforts with varying degrees of requirements changes?

  Hypothesis 1: There is no difference between the Fractional Method using linear regression and the REF in their forecast of when the requirements will become stable.

The purpose of the first simulation experiment is to compare the REF and Fractional Method forecasts of requirements trends and evaluate their strength of correlation.

The second research question is:

- Can the REF accurately estimate requirements engineering effort using measured data published in the SE literature for comparison?

Hypothesis 2: There is no difference in the mean engineering effort measured by Valerdi [2005] and the mean engineering effort estimated by the REF.

The purpose of the second simulation experiment is to compare the REF estimation of requirements engineering effort with requirements engineering effort data measured by Valerdi [2005] for DoD programs.

6.3.3: Requirements Trend Factor Relationships

The important factors included in the requirements trend simulation experiments to evaluate the REF relative to the Fractional Method are shown in Figure 6.4. The number of requirements *R* and the requirements changes variables are the same for REF and Fractional Method in the simulation runs. These factors are input into the REF and Fractional Method for comparison in the simulations. The Fractional Method forecasts are based on the linear extrapolation of the total requirements volatility *RV* discussed earlier in section 2.2.1 in Chapter 2. The REF forecasts are based on the average rate of entropy change over the previous baseline measurement intervals and this average rate of change is extended to future intervals until the minimum entropy state reflecting maximum quality is reached.

Technical uncertainty decreases as requirements entropy decreases and requirements quality increases. In the REF, requirements entropy reflects the state of technical uncertainty and requirements quality at any given time in the requirements engineering effort, and therefore is evaluated as a new method for assessing the effects of requirements volatility on the status and progress of the requirements engineering effort

relative to the desired end state. The requirements entropy scales as a function of the number of requirements as shown in equation (13).

Requirements volatility requires changes to occur, however requirements changes do not necessarily result in negative requirements volatility effects. The requirements engineering effort by definition changes and transforms requirements as the iterative process defines the system in more detail to lower levels. The simulation assumes there are sufficient resources available to support the number of simulated changes per baseline measurement interval.

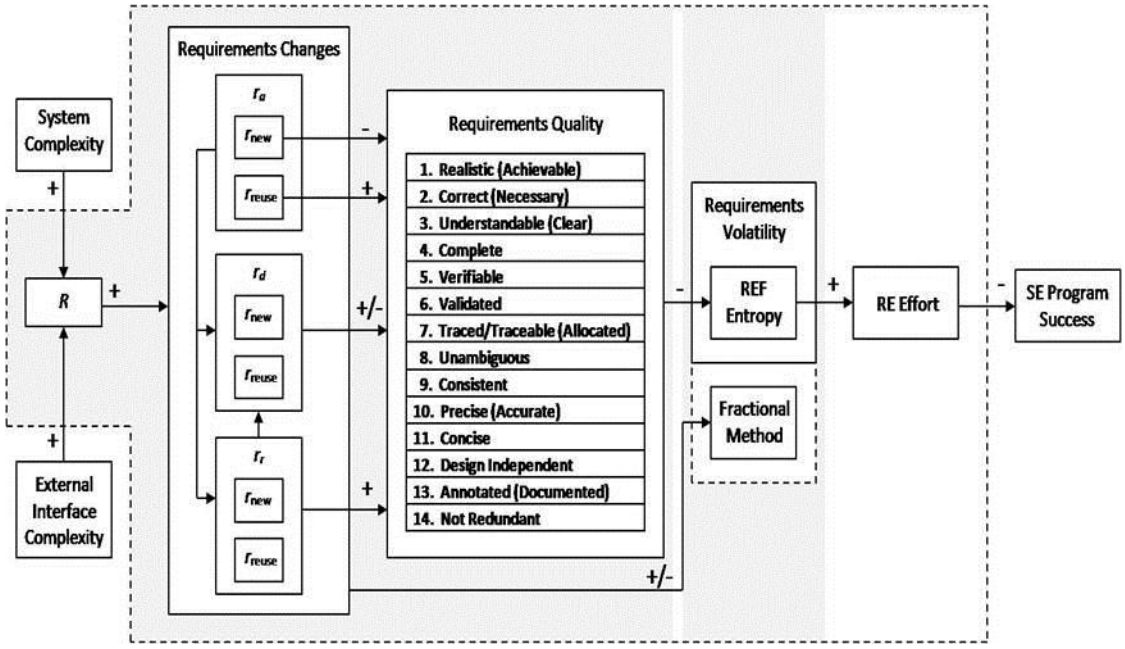


Figure 6.4: Model of Important Relationships in Requirements Trend Simulation Experiments

6.3.4: Requirements Engineering Effort Factor Relationships

The important factors included in the requirements engineering effort simulation experiments to compare the REF estimations with the measured engineering effort data

published by Valerdi [2005] are shown in Figure 6.5. For these simulations, the number of requirements *R* is randomly generated using the lognormal distribution with the mean *R* and standard deviation corresponding to the Valerdi [2005] data. This ensures that the average size of the SE programs used in the REF simulations in terms of their *R* is comparable to those in the Valerdi [2005] data. Comparison between the REF and COSYSMO in their estimations of engineering effort as a function of requirements volatility is outside the scope of this research but would be a reasonable next step.

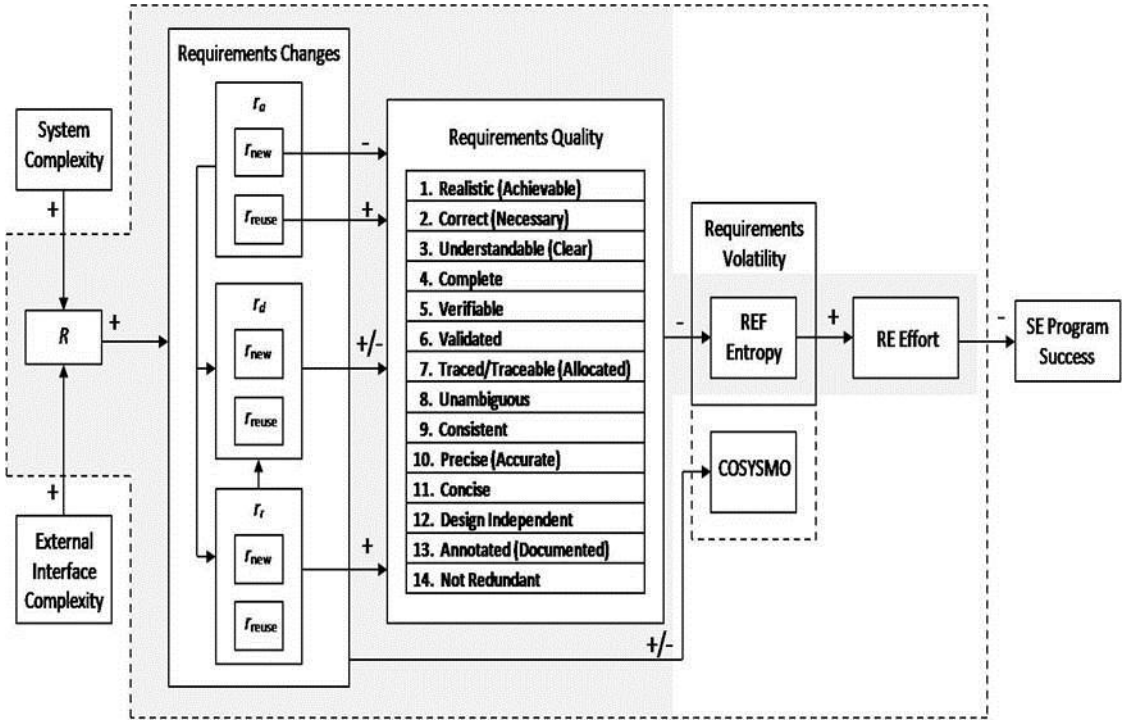


Figure 6.5: Model of Important Relationships in Requirements Engineering Effort Simulation Experiments

6.4: Data Collection Approach

Software code was written in MATLAB® [Mathworks, 2010] and simulation experiments were run to collect the data used to evaluate the REF as a new technology

for assessment of requirements trends and engineering effort. For the engineering effort simulations, the R is randomly generated at the start and remains constant throughout the simulation run from the initial state requirements quality distribution to the final state requirements quality distribution where all R occupy the desired end state $q_{14.}$

Five (5) separate engineering effort experimental runs were performed consisting of thirty-five (35) simulations of engineering effort per run to compare the REF engineering effort with the Valerdi [2005] data. A total of twenty-four (24) requirements trend simulation experiments were performed with varying degrees of requirements change behavior in order to compare REF and Fractional Method forecasts. The number of requirements deleted per measurement interval was randomly generated using a uniform distribution with a maximum of $0.1R$ based on the data published in the literature. The number of requirements revisions and additions per interval were modeled using normally distributed random variables and varying probabilities of new or reused requirements having a quality factor.

More than 40% of the Valerdi [2005] data are in the range between $400 \leq R \leq 1100$, and the engineering effort in this range varies from approximately 1.1 person-years for 1100 requirements to approximately 78 person-years for 750 requirements. The large variation in the macroscopic behavior observed for the requirements engineering effort in this range is taken into account in the REF by treating $p(r_{reuse})$ as a uniformly distributed random variable, which results in significant differences in the *number of possibilities* between efforts with relatively high requirements reuse (r_{reuse}) and high initial state requirements quality versus efforts with relatively low r_{reuse}, higher r_{new}, and relatively low initial state requirements quality. After the number of new and reused requirements

are either manually input or randomly generated, the *R* are distributed accordingly among the $N = 15$ quality levels using binomial probability distributions assuming that reused requirements will have a higher probability of a quality factor bound to it than new requirements when the requirements engineering effort starts. It was assumed in the REF simulations that SE programs with a higher $p(r_{reuse})$ at the start also have a higher probability of r_{new} quality because the experience on previous system development programs of a similar nature result in a better understanding the new system requirements.

Chapter 7: Simulation Experiments

7.1: Requirements Volatility

Some refer to requirements volatility as "the growth or changes in requirements during the project's development lifecycle" [Ferreira et al, 2009:1568] and the term is often used synonymously with "requirements change, requirements creep, scope creep, requirements instability, and requirements churn" [Ferreira et al, 2009:1568; Costello, 1994]. Requirements changes "may be good, detrimental, or neutral to our system development" [INCOSE, 2010:40]. Requirements volatility in this dissertation refers to the negative effects of requirements changes that increase technical uncertainty and decrease requirements quality. The effects of requirements volatility are often unpredictable, difficult to control, and a destabilizing influence in the requirements engineering process. Requirements trends are among the most important leading indicators of SE program technical effectiveness, and requirements volatility has been suggested as a good indicator of SE practices [Rhodes et al, 2009].

The impact of requirements volatility on the program is difficult to measure when one considers that the requirements engineering process is iterative and involves changes that transform the requirements [DAG, 2011:168] from a state of lower information quality to a desired state that reflects high information quality. Kulk and Verhoef [2008] suggest requirements changes are "necessary and healthy…..and a fact of life" but the challenge is to "control the volatility" [Kulk and Verhoef, 2008:138]. The positive effects of requirements changes may be offset by negative "ripple effects" [INCOSE, 2010b] that can significantly reduce requirements quality because a single revision can negatively impact numerous other requirements. Requirements volatility increases project scope in the vast majority of projects and the negative effects are particularly high when requirements volatility occurs after the start of the formal design process [Ferreira et al, 2009:1570].

The simulation results presented in this dissertation address the special case where the probability distribution for the transition of an *existing* requirement r_j is limited to the current state or higher states of quality. A revision to r_j either increases its quality or has no net effect. The probability for the transition of other existing requirements to lower states of quality as a result of a revision to r_j is low in this case and therefore negligible. However, the initial state quality of r_a is simulated using the binomial probability distribution where the probability of success, or having a quality attribute at the time the requirement is added to the effort, is p($r_{\text{new}}q$). The H_R and uncertainty may increase significantly if the initial state quality of r_a is low at the time the new requirements are added to the effort. As a result, the r_a in the special case may have a significant negative impact on Q, H_R, and I.

7.1.1: Experimental Results

Six (6) examples from twenty-four (24) runs of a MATLAB® [Mathworks, 2010] simulation experiment are shown in Figures 7.1 through 7.6 to illustrate the potential advantages of the REF relative to the Fractional Method [INCOSE, 2010:15; Rhodes et al, 2009:31] for estimating, measuring and forecasting requirements trends and the impact of requirements changes on the program. The data for each of the twenty-four (24) simulation runs comparing the REF and Fractional Method forecasts of when the requirements will become stable are included in Appendix D.

In each of the simulations shown in Figures 7.1 through 7.6, the Fractional Method is plotted against the right ordinate using red, green, blue and gray lines for deleted (rdeleted), revised (rrevised), added (radded) and *RV*, respectively. The requirements entropy H_R is plotted in black against the left ordinate. Each time interval in the simulation run contains a single data point for each variable representing the cumulative change measured over that baseline measurement interval as per the Fractional Method shown in Rhodes et al [2009] and Costello and Liu [1995]. Each baseline measurement represents a discrete iteration of the simulation that "baselines" the requirements engineering effort at uniform intervals [INCOSE SE Measurement Primer, 2010:38]. The simulation assumes there are sufficient SE resources available to support the number of simulated changes per baseline measurement interval.

The Fractional Method forecasts shown in Figures 7.1 through 7.6 use the same linear extrapolation method based on the total *RV* (which are actually requirements changes) as Figures 2.1 and 2.2 to extrapolate when the requirements are expected to reach zero fractional volatility and therefore become stable. These forecasts are compared in each

simulation run to the REF forecast based on the rate of requirements entropy change with respect to time over the baseline measurement intervals prior to the last simulation data point or dH_R/dt to forecast when the requirements will become stable. By definition, the H_R forecast shown in each plot corresponds to the baseline measurement interval when the requirements reach the desired end state q_{14} where $H_R = H_{R\min}$ and no additional changes to the requirements are necessary.

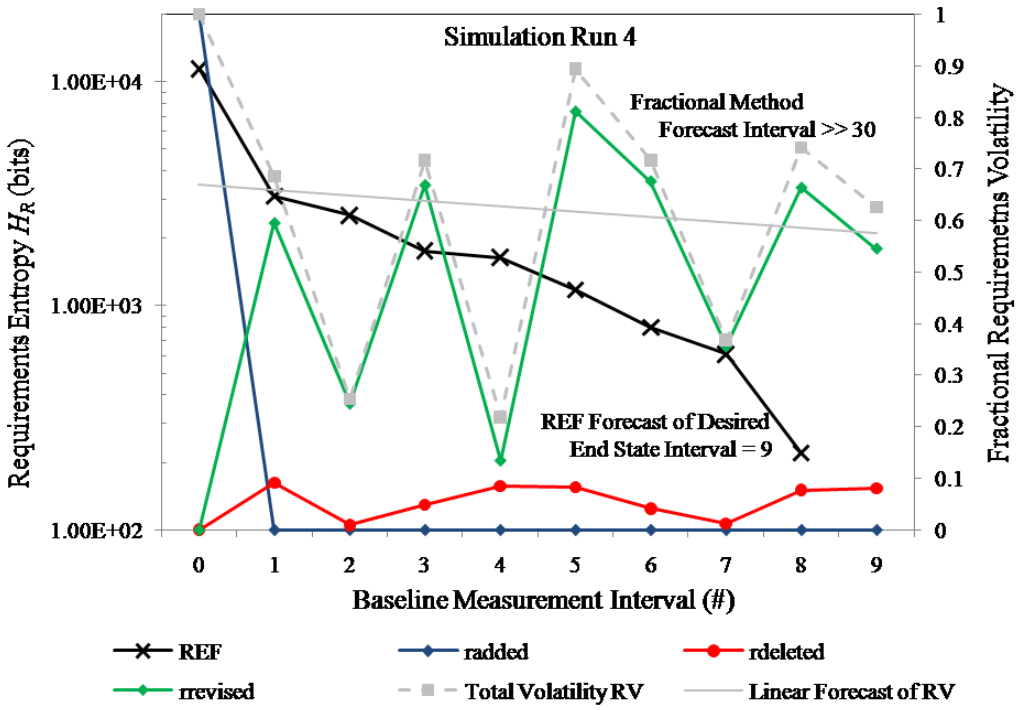


Figure 7.1: Requirements Trend Simulation Run4.

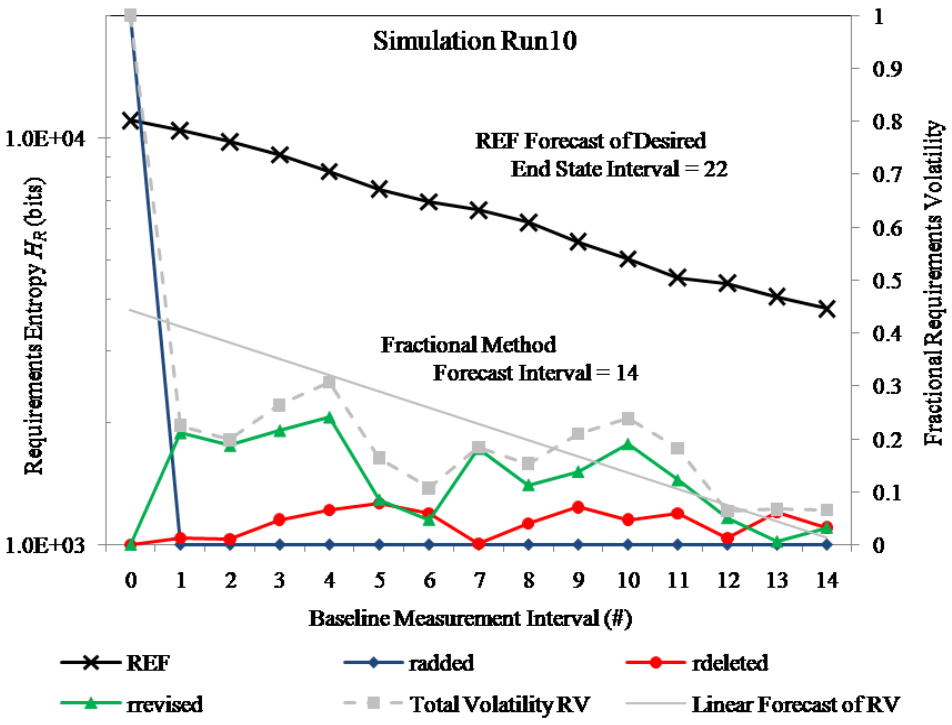


Figure 7.2: Requirements Trend Simulation Run10.

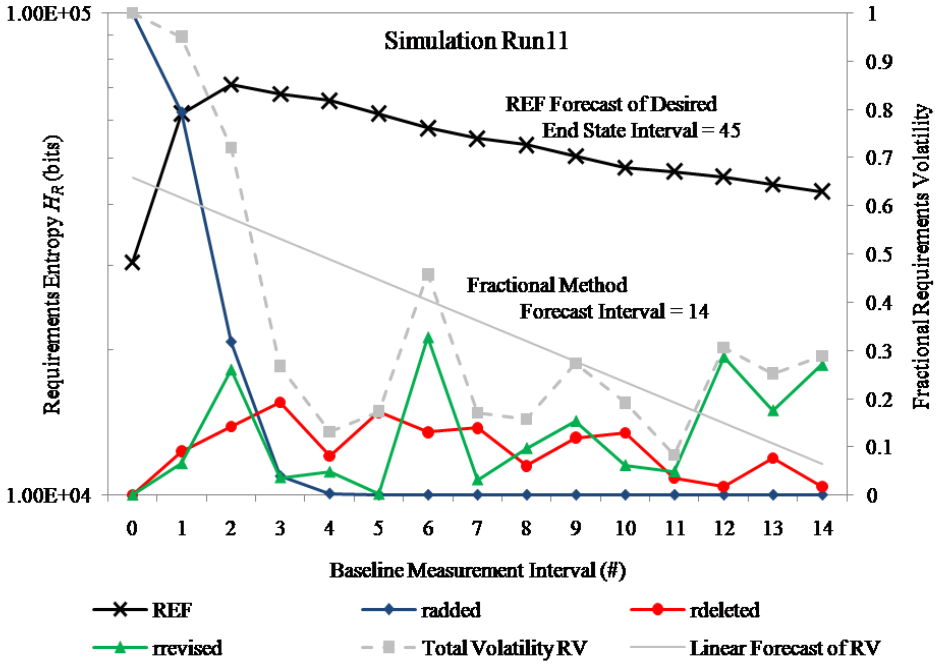


Figure 7.3: Requirements Trend Simulation Run11.

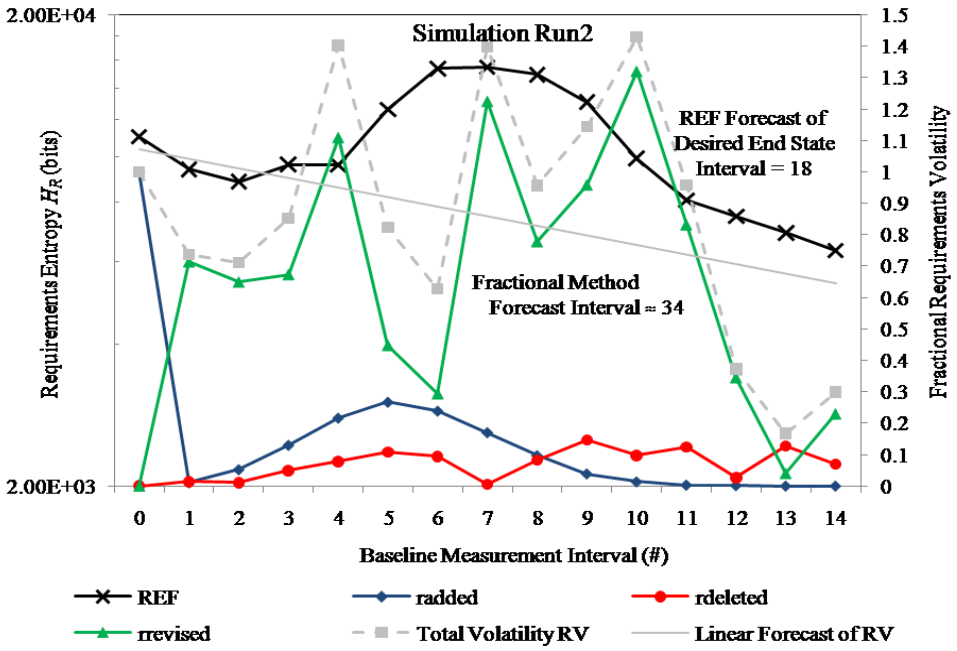


Figure 7.4: Requirements Trend Simulation Run2.

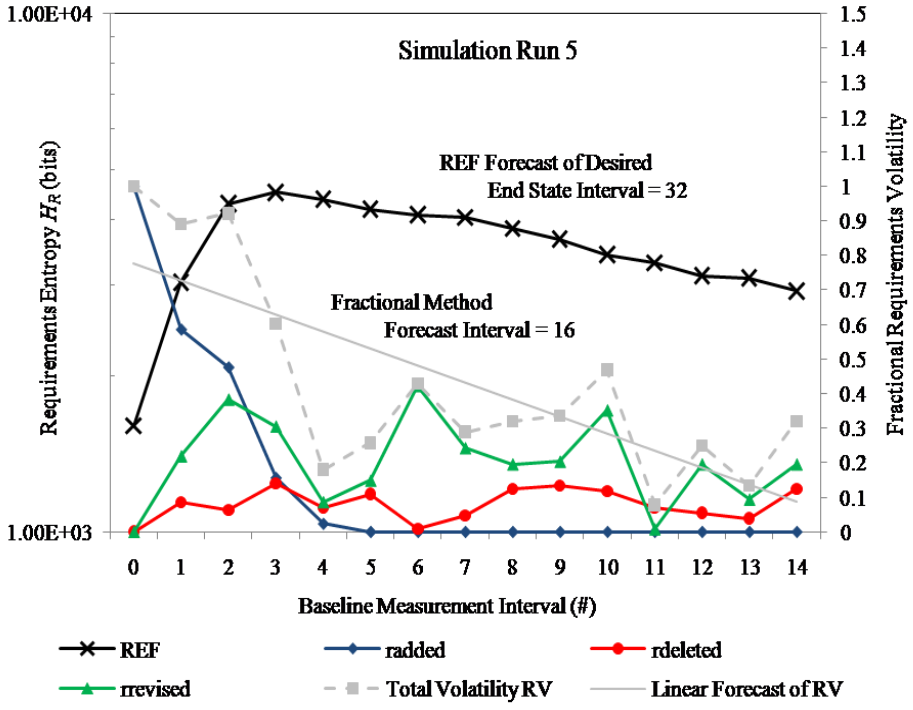


Figure 7.5: Requirements Trend Simulation Run5.

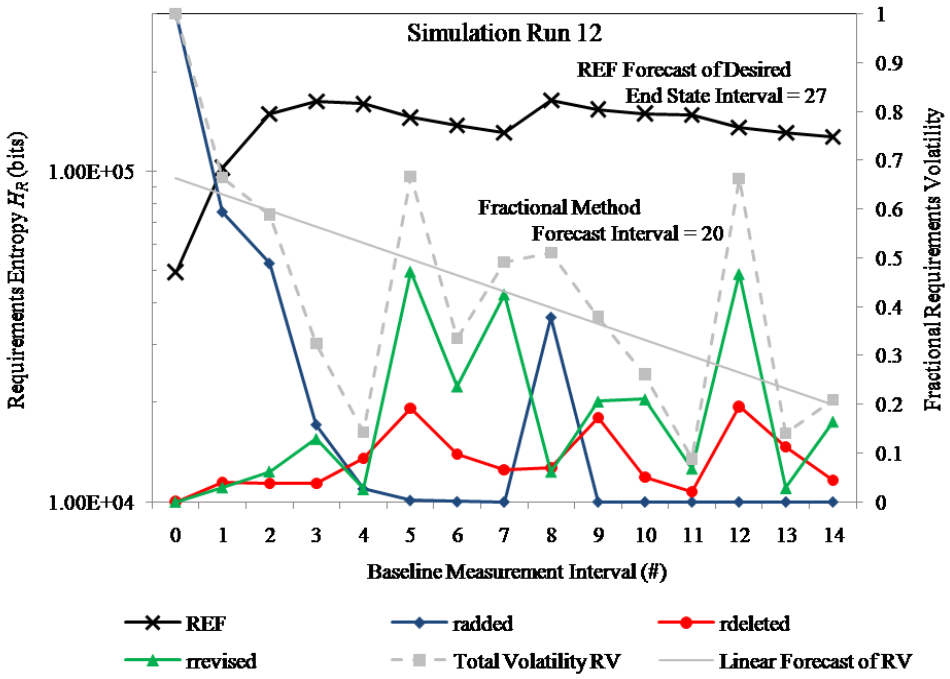


Figure 7.6: Requirements Trend Simulation Run12.

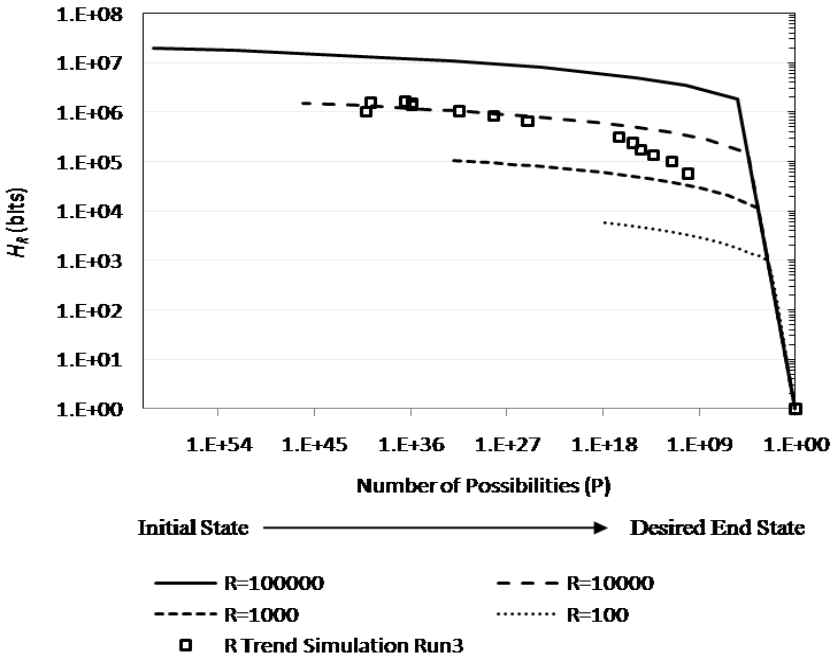


Figure 7.7: Requirements trend simulation run3 results plotted and compared to the theoretical entropies of Figure 5.3.

7.1.2: Data Analysis

In the simulations for Figures 7.1, 7.2, 7.3 and 7.5 (simulation runs 4, 10, 11 and 5, respectively), requirements are added early in the effort prior to baseline measurement interval 4. This behavior is consistent with the Rhodes et al [2009] and Costello [1995] data shown in Figures 2.1 and 2.2. In Figures 7.1 and 7.2, requirements were only added at the start of the program or baseline measurement interval 0. Such behavior could represent an incremental improvement program in which the customer needs and system requirements are well understood at the start and no hidden requirements were discovered during the process. In both of these "well-behaved" cases, there is a significant difference between the H_R forecast of when the desired end state will be reached and the Fractional Method forecast of when the changes will be zero. The Fractional Method predicts stability much later than the REF in Figure 7.1 and much sooner than the REF in Figure 7.2.

Different behavior is observed in Figures 7.4 and 7.6 when requirements are added after baseline measurement interval 6. In Figure 7.4 (simulation run 2), requirements are added from intervals 2 through 10. The positive impact of requirements *revisions* is unable to offset the negative impact of requirements *additions* between intervals 3 and 6, resulting in a net increase in H_R and evidence that the requirements are not transitioning toward the desired end state during this part of the effort despite a higher fraction of revisions than additions. The REF forecast of when R will reach the desired end state is nearly 2x earlier than the Fractional Method forecast based on RV. In this case, the requirements changes quantified by H_R are good because the changes are transitioning R to the desired end state more quickly but the forecast of the Fractional Method does not

provide the same insight. In Figure 7.6 (simulation run 12), approximately 40 percent of the requirements are added at interval 8. This could occur, for example, as a result of a formal review with the customer at which time many new requirements are identified and added to the program. The relative net increase in H_R at interval 8 is similar to the relative net increase in H_R from intervals 4 to 6 in Figure 7.4 but the fractional volatility in this case is less and the revisions are transitioning *R* toward the desired end state more slowly. As a result, the Fractional Method forecast of when the program would be ready for a formal review is 35 percent sooner than the desired end state forecast by the REF.

The baseline measurement interval forecasted by the Fractional Method is earlier than the REF in Figures 7.2, 7.3, 7.5, and 7.6 and later than the REF in Figures 7.1 and 7.4. The simulations shown in Figures 7.1 and 7.4 have the highest fractional changes and may be interpreted to exhibit negative effects of volatility but in fact the high fraction of revisions, or changes, is transitioning *R* toward the desired end state more quickly.

The simulation results assume that *R* do not need to undergo further changes or refinements when the q_{14} is reached. In effect, the fractional changes are zero at $H_{R\min}$ and by definition no additional requirements changes are necessary. The requirements have reduced technical uncertainty to a minimum and have achieved maximum stability when $H_R = 0$. At this point in the process, the requirements would be ready for a formal review and would be expected to support the system design process if the objective system is determined to be fully defined by *R*. The H_R metric informs the systems engineers, program managers and decision makers whether or not the effort is making progress toward the desired end state despite the complex behavior of requirements

changes. This information may increase confidence that the requirements engineering effort is under control and the SE program is moving in the right direction.

The H_R data from requirements trend simulation run3 are plotted in Figure 7.7 as a function of P with the theoretical entropy curves shown previously in Figure 5.3. The abscissa has been reversed to illustrate behavior of the requirements engineering process over time as the R transition from their initial state toward the desired end state. The maximum R during simulation run3 was 27,310. The behavior of the run3 H_R data are similar to the theoretical H_R for the uniform transition case when P is very large ($P >$ 1E18).

7.1.2.1: Analysis of Forecasts and Strength of Correlation

The mean difference between the Fractional Method and the REF in their forecast of when the desired state would be reached is 17 baseline measurement intervals for simulation runs in this experiment. Statistical analysis suggests that there is sufficient evidence to reject the null Hypothesis 1 ($p < .01$) and conclude that the difference between the Fractional Method and the REF in their forecast of when R will not need any more changes is statistically significant. The mean coefficient of determination (R^2) between the Fractional Method and REF is 0.317, indicating a weak to moderate strength of correlation between the two methods. An example regression analysis plot is shown in Figure 7.8.

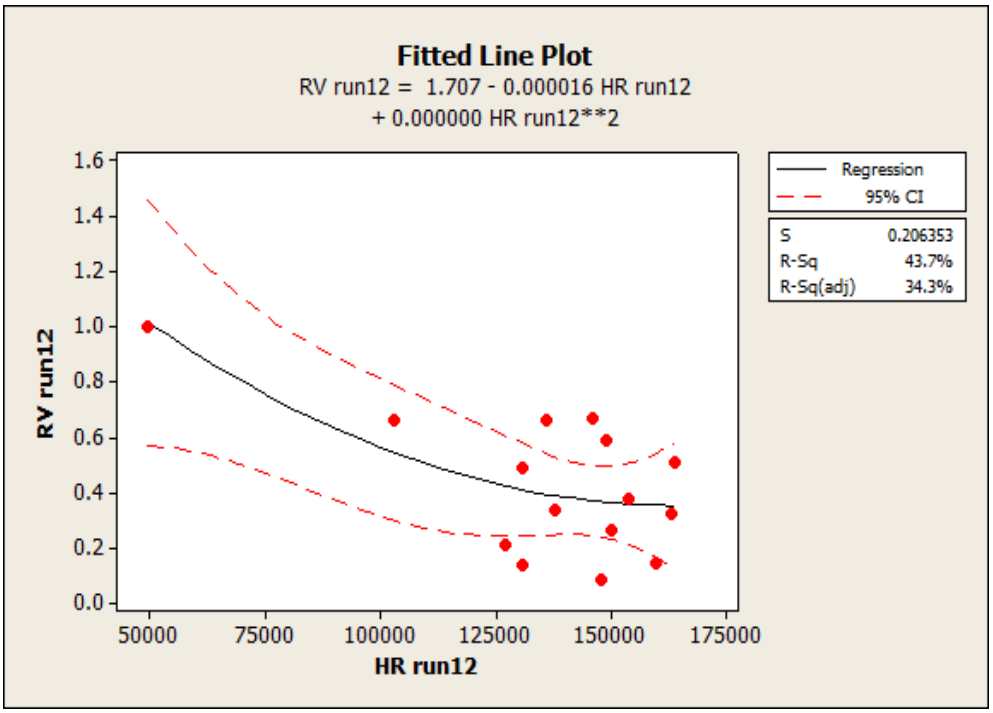


Figure 7.8: Regression Analysis of the Fractional Method vs REF for Simulation Run12.

The relatively weak correlation between the REF and Fractional Method is a positive result because of the inherent limitations of the Fractional Method. These results are not unexpected given the fact that the REF is sensitive to requirements quality, which is the ultimate objective of the requirements engineering effort, and the Fractional Method only measures the quantity of requirements changes, which can have positive or negative impact on requirements quality. The significant difference between the REF and Fractional Method may explain the limited measurement insight provided by the Fractional Method in the past.

7.2: Requirements Engineering Effort

Accurate estimation of requirements engineering effort is critical for systems engineers, program managers, contractors and decision-makers to properly allocate the resources necessary to complete the requirements engineering process and build the right system to satisfy customer needs. In DoD, each MDAP requires risk mitigation planning that includes a general estimate of the level of effort required to bring the program to a successful conclusion. In risk mitigation planning "there must be enough detail to allow a general estimate of the effort required and technological capabilities needed based on system complexity" [DAG, 2011:193] but little guidance is provided on generally accepted approaches or best practices for estimating, measuring or forecasting requirements engineering effort.

The most recent models known to be currently available to the SE community (e.g. COSYSMO) are generally empirical in nature and are based on equations with effort multipliers and adjustable parameter exponents that require custom 'calibration' from organization to organization to obtain reliable results [Valerdi, 2005:78]. Valerdi [2005] measured requirements engineering effort for thirty-five (35) DoD programs from six (6) different major defense contractors [Valerdi, 2005], and these measured data serve as the basis for comparison of the REF simulation experiments shown in Figures 7.10 through 7.14. The Valerdi [2005] data are shown by the black crosses in these Figures. The requirements engineering effort data measured by Valerdi [2005] ranges from relatively small system development programs with $R \approx 70$ to very large programs with $R \approx 20{,}000$. A regression plot of the Valerdi [2005] engineering effort data versus the number of requirements is shown in Figure 7.9. There appears to be a low to moderate correlation,

and this is consistent with the Valerdi [2005] findings. Regression analyses of all the REF simulation runs are provided in Appendix A. A few of the simulation runs suggest a relatively high degree of correlation between engineering effort and the number of requirements, which may explain why some systems engineers find the number of requirements to be a useful metric for estimating the engineering effort.

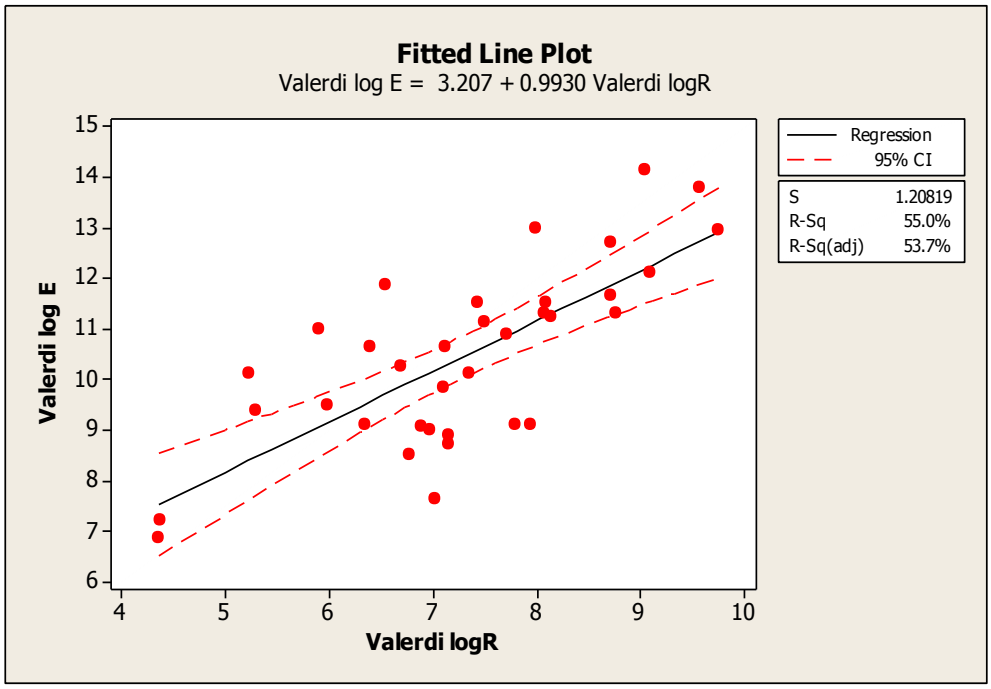


Figure 7.9: Regression Analysis of the Engineering Effort vs Number of Requirements in Valerdi [2005].

7.2.1: Experimental Results

The blue and red solid lines in Figure 7.10 are the maximum requirements engineering effort estimated by the REF using Bose-Einstein and Maxwell-Boltzmann statistics, respectively, and assuming an initial state of $H_{R\max}$. These estimates of the maximum expected requirements engineering effort assume R is constant from start to finish and correspond to the transformation of the initial state $R \rightarrow q_0$ to the desired end state $R \rightarrow$

q_{14} where ΔI is calculated using the difference between the initial state requirements entropy $H_{Ri} = H_{R\max}$ and the final state $H_{Rf} = H_{R\min}$. The REF $H_{R\max}$ assuming an information output rate of 1.0 bit per second (bps) bounds the high range of the Valerdi [2005] data reasonably well. Of the thirty-five (35) data points measured by Valerdi [2005], only two (2) exceed the upper bound estimate. The Bose-Einstein statistics appear to result in a reasonable approximation of what could be expected for a *maximum* requirements engineering effort for a given R, and discrepancies that exceed the $H_{R\max}$ estimate may indicate inefficiencies in the SE program or reflect other factors that have been correlated with SE program cost growth, such as funding instability, lack of SE expertise in program management offices, technology immaturity, lack of user involvement and support, key personnel turnover and inadequate risk management [NRC, 2008; Bottelson, 2011]. An information output bitrate of 0.1 bps bounds all the data points with margin as shown in Figures 7.11 through 7.14.

The initial state requirements quality distributions used for the REF simulations in Figures 7.10 through 7.14 were estimated by randomly generating $p(r_{reuse})$ with a maximum of 0.4 for $R < 400$ and $0.6 \leq p(r_{reuse}) \leq 0.9$ for $R > 1100$. For $400 \leq R \leq 1100$, $0 \leq p(r_{reuse}) \leq 1$. More than 40% of the Valerdi [2005] data are in the range between $400 \leq R \leq 1100$, and the engineering effort in this range varies from approximately 1.1 person-years for 1100 requirements to approximately 78 person-years for 750 requirements. The large variation in the macroscopic behavior observed for the requirements engineering effort in this range is taken into account in the REF by treating $p(r_{reuse})$ as a uniformly distributed random variable, which results in significant differences in the number of possibilities between efforts with relatively high

requirements reuse (r_{reuse}) and high initial state requirements quality versus efforts with relatively low r_{reuse}, higher r_{new}, and relatively low initial state requirements quality. After the number of new and reused requirements are either manually input or randomly generated, the R are distributed accordingly among the $N = 15$ quality levels using binomial probability distributions assuming that reused requirements will have a higher probability of a quality factor bound to it than new requirements when the requirements engineering effort starts.

It was assumed in the REF simulations that SE programs with a higher $p(r_{reuse})$ at the start also have a higher probability of r_{new} quality because the experience on previous system development programs of a similar nature result in a better understanding the new system requirements. Examples of initial state requirements quality distributions were shown previously in Figures 5.4 and 5.5. Once the initial state requirement quality distribution is determined, H_R and ΔI are calculated using equations (13) and (16), respectively. The ΔI in bits is converted into time units by estimating an information output rate using a normally distributed random variable between 1.0 bps for smaller programs and 5.0 bps for larger programs based on insight in the literature [Reed and Durlach, 1998; Mason, 1978; Tribus and McIrvine, 1971]. This bitrate is a function of the SE team's productivity and may vary significantly depending on factors such as the degree of domain expertise, requirements engineering experience, resource allocation and system complexity. More data are needed to gain insight into this bitrate because such data does not appear to be available in the SE literature.

The blue circles in Figure 7.10 through 7.14 are REF estimations of ΔE for thirty-five (35) randomly generated sets of requirements, each representing the R of a SE program.

The ΔE was estimated by randomly generating initial requirements quality distributions and calculating the ΔE necessary to transition these requirements to the desired end state using equation (15). The initial R remains constant and represents the R in the desired end state for each simulation run.

In order to run the simulation experiments and compare the requirements engineering effort measured by Valerdi [2005] with the ΔE estimated by the REF, the mean of the natural log of R (7.29) and standard deviation ($\sigma = 1.3264$) of the Valerdi [2005] data were used to randomly generate R in the REF using the lognormal distribution. The SE programs generated by the REF range in size from $R = 80$ to 15,000 for simulation experiments. The mean natural log of ΔE estimated by the REF for the simulation shown in Figure 7.10 is 9.93 ($\sigma = 1.881$) and the mean requirements engineering effort estimated from the measured data published by Valerdi [2005] is 10.447 ($\sigma = 1.775$). Statistical analysis of the simulation results suggests that there is not sufficient evidence to reject Hypothesis 2. The difference between the $\langle \Delta E \rangle$ estimated by the REF and engineering effort measured by Valerdi [2005] is not statistically significant ($p < .05$; $z = 1.72 < 1.96$). Similar results were found for the simulation runs shown in Figures 7.11 through 7.14. These findings suggest that the REF may be a viable new quantitative method for estimating, measuring and forecasting requirements engineering effort. The input parameters required for the REF can be measured or estimated at the start of the program or any other point by the systems engineer, program manager or decision-maker and can be updated continuously to monitor the status and progress of the effort.

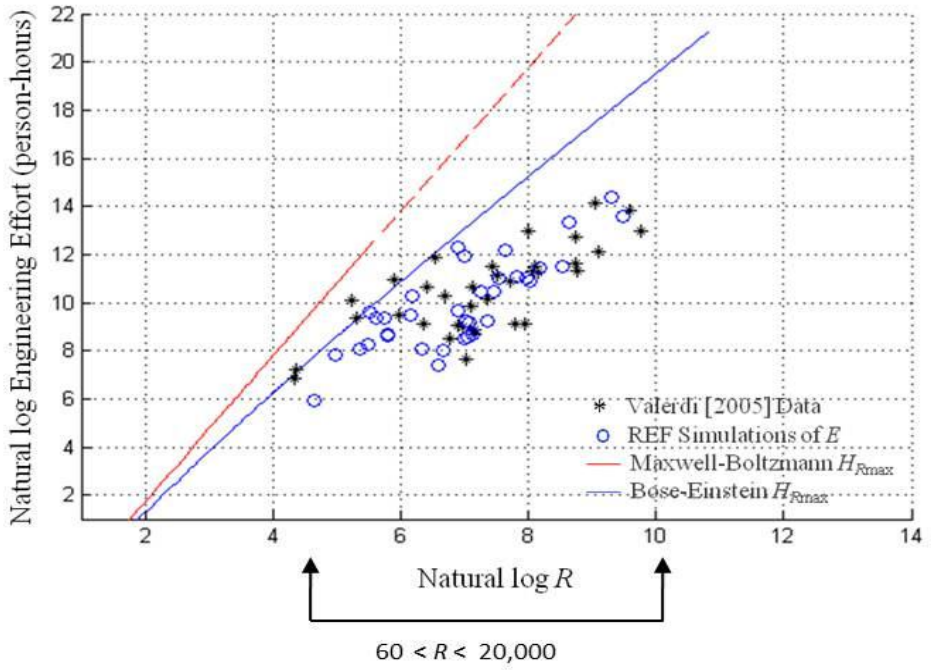


Figure 7.10: Simulation Run1 with ⟨ΔE⟩ = 9.93 and σ = 1.88.

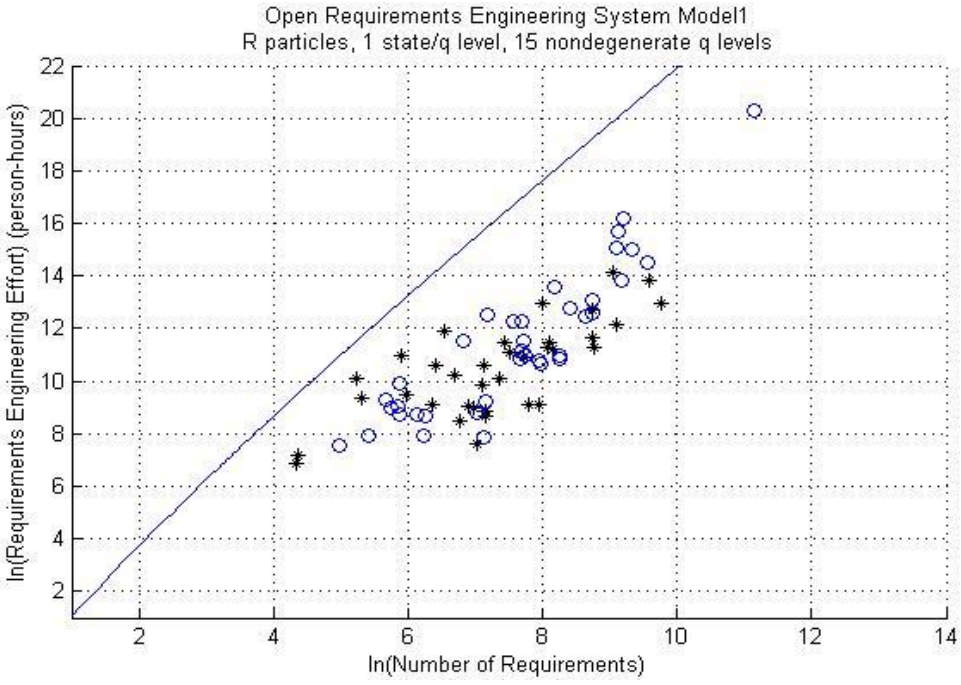


Figure 7.11: Simulation Run2 with ⟨ΔE⟩ = 11.46 and σ = 2.872.

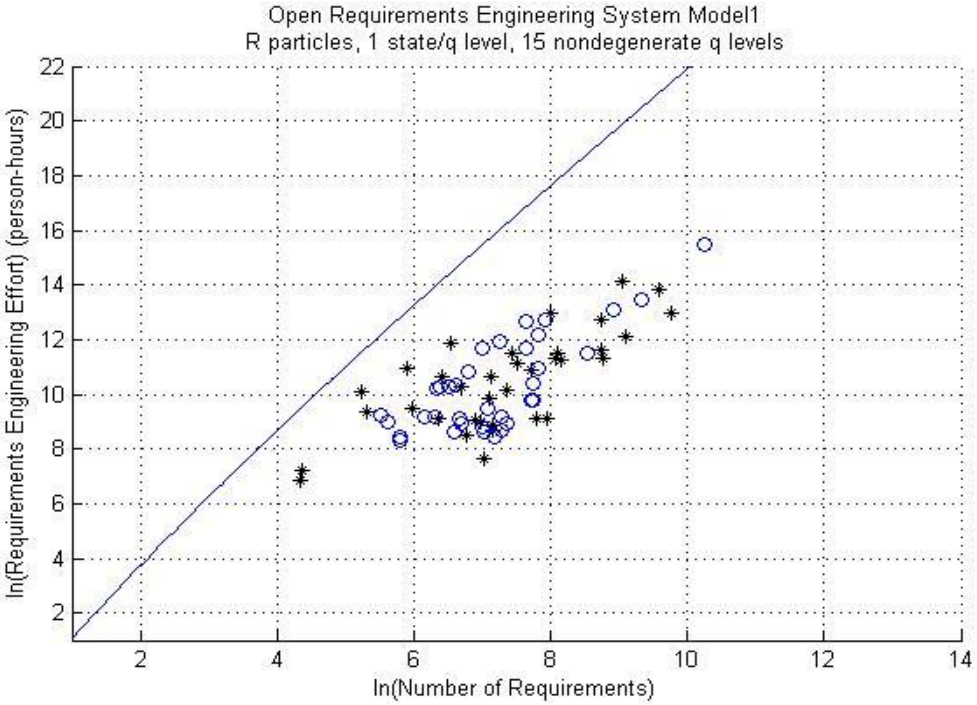


Figure 7.12: Simulation Run3 with ⟨ΔE⟩ = 10.32 and σ = 1.733.

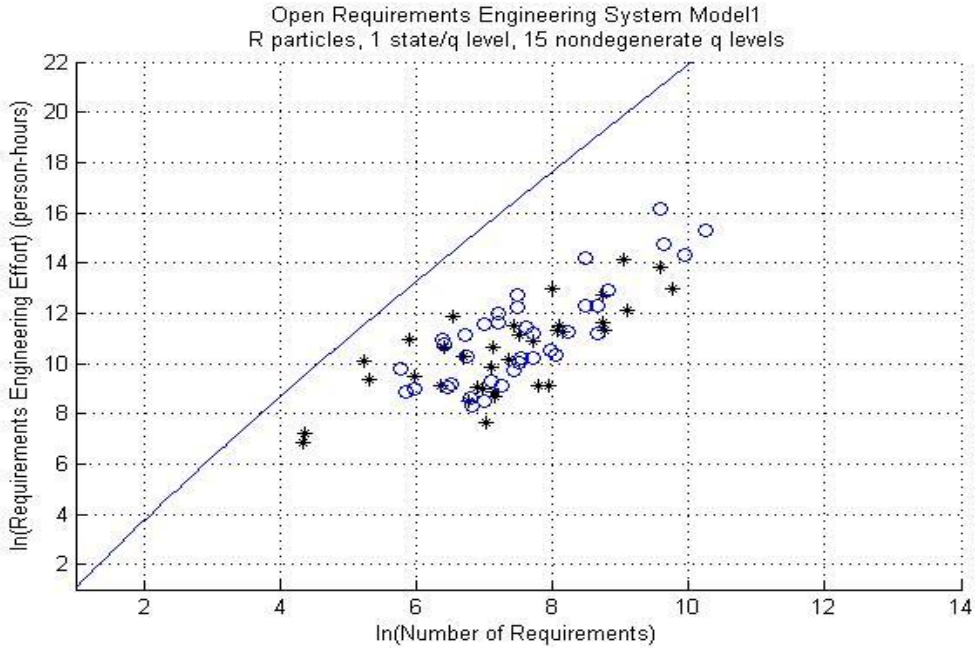


Figure 7.13: Simulation Run4 with ⟨ΔE⟩ = 10.38 and σ = 2.761.

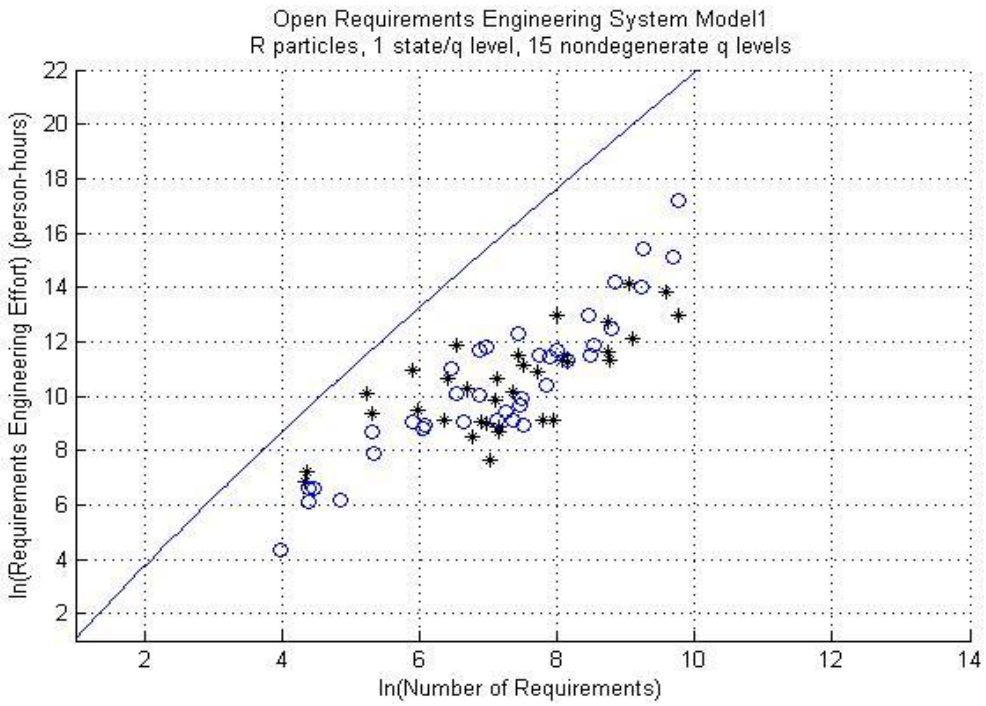


Figure 7.14: Simulation Run5 with ⟨ΔE⟩ = 11.14 and σ = 2.029.

A comparison between an estimated *lower range* of Δ*E* and the Valerdi [2005] SE program data is shown in Figure 7.15. This lower bound was estimated using a relatively high probability that a given requirement r_j will have all quality factors bound to it at the start of the program (0.86). This case may reflect an incremental improvement or a minor upgrade program with very high requirements reuse. The REF parameters used in this simulation estimate a lower bound of Δ*E* for *R* ranging from 90 to 1.63E5. The REF simulations suggest that there may be underling structure in the Valerdi [2005] data. A more detailed study of the REF and other models (e.g. COSYSMO) would provide valuable insight into the relationship between the macroscopic behavior of the requirements engineering process and important underlying properties and interactions contributing to that behavior.

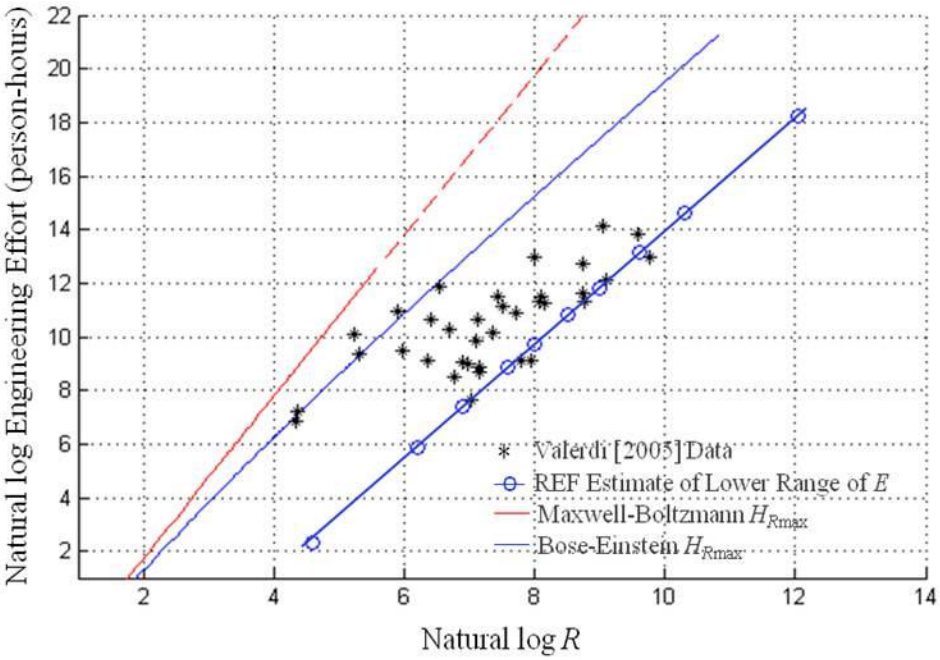


Figure 7.15: Simulation Run1 with estimation of lower entropy bound of Valerdi [2005] data.

7.2.2: Requirements Engineering Effort Data Analysis

7.2.2.1: Statistical Analysis of REF Results and Measured Data

The mean $ln\Delta E$ of the five (5) REF simulation runs is 10.65 with a mean standard deviation of 2.216 compared with the Valerdi [2005] mean engineeirng effort of 10.447 with a standard deviation $\sigma = 1.775$. Null hypothesis individual value plots for each of the five (5) simulation runs are provided in Appendix B. In each case, the null hypothesis is not rejected. There is not a statistically significant difference between the Valerdi [2005] data and the REF estimation of engineering effort using equation (15). There is no evidence to suggest that such a method or the underlying theory upon which it is based is currently available to the SE community.

Chapter 8: Conclusions and Future Research

DoD has placed greater emphasis in recent years on the incremental improvement and upgrade of existing systems versus the development of new systems that provide leap-ahead capabilities. A major driver of this approach has been the need to get systems into the field more quickly to support on-going operations and address rapidly changing threats. The Defense acquisition community has imposed tighter schedules for the delivery of systems to meet urgent operational needs, and these tighter schedules constrain new technology development, major redesigns, system innovation, and risk tolerance.

It is likely that this more conservative approach to system development has led to greater reuse of requirements for systems that have already been developed and fielded, and this requirements reuse should be reflected to some degree in the requirements engineering effort data measured by Valerdi [2005]. Requirements reuse can benefit the engineering effort if executed and managed properly, and caution should be exercised when reusing requirements [Botta et al, 2006]. Programs with higher requirements reuse can start the requirements engineering effort with a higher mean requirements quality distribution, resulting in less engineering effort to reach the desired end state. The variance in mean initial quality level for programs with low versus high requirements reuse was a factor that contributed to the good agreement between the REF simulation results and the measured data shown in Figures 7.9 through 7.13 [Valerdi, 2005]. Valerdi [2005] acknowledges that "(t)he effects of volatility and reuse "were deferred because of the difficulty in gathering data to calibrate them (in the COSYSMO model). Future work should incorporate these relevant parameters into the (COSYSMO) model" [Valerdi,

2005:116]. The REF accounts for the effects of new and reused requirements in terms of probability distributions and the likelihood of a reused requirement occupying a given quality level. The probability of *R* initially having higher quality and lower H_R, and therefore requiring less engineering effort to reach the desired end state, increases as the number of reused requirements increases. The REF is inherently flexible and can accommodate any mean quality level for new or reused requirements determined to most accurately reflect the current state in the context of the new program.

The number of bits necessary to communicate the system requirements is far less important than requirements quality. Two engineers may differ in the number of bits needed to communicate a set of *R*, but there should be general agreement when *R* meets the quality attributes, one of which is 'concise'. Assuming that an inflexible constraint on the size of the SRD does not exist, the number of bits is far less important than whether or not the SRD contains all the information necessary for the right system to be designed and built the right way. It is possible for a single requirement statement to not have several quality attributes for the same or similar reasons, and a single change to the requirement may impact several quality attributes simultaneously. For example, a requirement with a TBD or TBR is by definition incomplete. Such a requirement may also lack the information necessary to determine whether it is correct, unambiguous, realistic or consistent (or possibly others). In this case, the requirement entropy would be relatively high, and it is possible that a single change (e.g. input of a parameter value) can transition the requirement from high entropy to low entropy. A standard operating procedure (SOP) for requirements validation is recommended to ensure its effective implementation and the consistent assessment of requirements quality, whether using the

REF or not. For this reason, we include SOPs as a control input for the REF modeling process in Figure 5.6.

A system that satisfies customer needs is a quality system in the eyes of the customer, at least until their needs change or other needs are discovered during its use. Even then, the customer usually appreciates the quality of the legacy system and wants the new system to do everything the legacy system can do and more. A quality system is a reflection of good requirements, and good requirements reflect the quality attributes that have been determined to be important for a quality system to be built. The REF describes the behavior of the requirements engineering process from a quantum point of view but does not necessarily change the way we do requirements engineering, with the possible exceptions of placing a greater emphasis on requirements quality and integrating its assessment into the engineering effort.

System development projects "can be viewed as a stochastic process" and they "achieve values in accordance with as-yet-unknown probability distributions" [Honour, 2004:3]. The REF defines a probabilistic methodology for describing the stochastic behavior of the requirements engineering process. Although there are many ways for the process to fail, it is intuitive and generally understood that there is also more than one way to implement the process successfully and achieve a positive outcome. The REF accounts for the random and deterministic elements of the process by extending principles of information theory and statistical mechanics to SE. The simulation results suggest that the REF is a step toward resolving the requirements engineering problem because of the insight it provides into the stochastic nature of the open requirements engineering process and the underlying properties contributing to the observed behavior.

Particles in a physical system occupy energy levels corresponding to their accumulated energy, and the physical entropy reflects the number of ways that the particles can reconfigure themselves among the available states. Requirements in the REF occupy quality levels corresponding to their accumulated quality, and the requirements entropy reflects the number of ways that the requirements can be configured among the available states. The REF quality levels, and the cumulative attributes they represent, are analogous to higher energy levels in a physical system that become more probable as the total energy of the system increases. In our case, higher quality levels become more probable for requirements to occupy as the total cumulative quality Q increases. For the purpose of this research, we assume that requirements at a higher level of quality are not allowed to reconfigure themselves to lower quality levels randomly or arbitrarily. This redistribution may occur, however, if a new requirement is added or an existing requirement is revised such that it has a negative impact on other existing requirements, such as if a key performance parameter is increased and becomes more difficult to achieve and has a negative ripple effect on other requirements. This effect would be expressed in the entropy when the calculation is redone because of its impact on the requirements quality distribution.

The REF is a practical quantitative method for measuring requirements quality, integrating its assessment into the requirements engineering effort, and facilitating the V&V during requirements management activities. These practical considerations are important because without a straightforward implementation plan, the REF will likely not be used in SE programs and, as a result, will likely have minimal impact on resolving the real world requirements engineering problem.

8.1: Summary of Research Findings

The results presented in this dissertation suggest that the REF may be a useful quantitative method for estimating, measuring and forecasting requirements trends at any given time in the requirements engineering process. Results from requirements trend simulations demonstrate the potential value of the REF to address limitations of the Fractional Method currently described in the SE literature and improve the measurement of the status and progress of the requirements engineering effort. The simulation results show that the REF may be a more precise and accurate indicator of requirements trends because the REF distinguishes between the positive, negative or neutral effects of r_r and r_a relative to the desired end state or some other interim state that may be of interest. There is a statistically significant difference between the REF and the Fractional Method in their forecasts of when the requirements will become stable. This finding is not unexpected when one considers the purpose of the requirements engineering effort is to define a complete set of requirements with sufficient information quality for the right system to be designed and built the right way. The REF accounts for requirements quality in the estimation, measurement and forecast of requirements entropy while the Fractional Method only measures the quantity of requirements changes without any sensitivity to their relative impact on information quality.

Simulation results are also presented that suggest the REF may be a promising new method for estimating, measuring and forecasting requirements engineering effort. Results from random variable simulations show that there is not a statistically significant difference between the mean engineering effort estimated by the REF and the mean engineering effort measured for DoD programs [Valerdi, 2005]. The Valerdi [2005],

Rhodes et al [2009] and Costello [1995] data represent collective experience and observations of *macroscopic* behavior of the stochastic requirements engineering process, and the REF is a theoretical construct of fundamental underlying properties that describe this behavior.

Also introduced is a new theory of information quality that extends principles of information theory and statistical mechanics to information development processes. The special case of the requirements engineering process is considered in this dissertation. Axioms of the theory of information quality are compared with those that underpin information theory and statistical mechanics, and the similarities in the mathematical foundation of the theories are shown.

New methods, tools and techniques are needed by SE practitioners, program managers and decision-makers to improve the engineering of increasingly complex systems. The considerable effort in recent years to develop leading indicators that correlate with program outcomes and improve the effectiveness of SE activities is timely and further demonstrates the need. Improved SE methods will enable a more diverse group of practitioners to successfully implement the SE process and bring high quality complex systems to market more quickly. The probabilistic theory of information quality and its implementation using the REF are intended to serve as "inter-disciplinary connectors" [INCOSE, 2007:20] to bridge SE with well established principles of other disciplines for the purpose gaining a better understanding of SE process behavior.

Improved SE methods would benefit experienced SE practitioners as well as scientists and engineers in Research, Development and Engineering Centers (RDECs) and other R&D organizations responsible for implementing SE earlier in the System Development

Lifecycle. Scientists and engineers in R&D organizations may be unfamiliar with SE and may be unaccustomed to principles without an accepted theory [INCOSE, 2007:19] or mathematical equations to guide them. The R&D community may find quantitative SE methods and metrics more appealing than processes and heuristics. The REF can be used to facilitate any requirements engineering process that includes requirements quality among its objectives and seeks to quantify the status and progress of the effort in terms of the characteristics shared by good requirements.

Requirements trends and engineering effort are examples of macroscopic behavior that can be observed and measured in the requirements engineering process. Study of the theoretical concepts that describe the behavior of complex processes in other disciplines was instrumental in the development of the REF. The REF applies "lessons learned" from these disciplines for the purpose of exploring the possibility that similar underlying properties may be contributing to observed macroscopic behavior of SE processes that appear on the surface to be quite different.

There have been differences of opinion within the SE community regarding the degree to which the SE discipline is underpinned by "formal or theoretical background for its practices" (Sheard and Mostashari, 2009:295; Simpson and Simpson, 2010). The debate may reflect different expectations for a theory of SE and the formalism upon which it is based. INCOSE recognizes the "lack of an accepted set of theory and principles" and the "lack of tools that enable use of existing standards" in the SE process [INCOSE, 2007:19]. This dissertation introduces a new theory of the relationship between the macroscopic behavior of the requirements engineering process and underlying properties contributing to that behavior that are otherwise difficult to measure. An objective of the

REF is to improve the understanding of the relationship between requirements changes, requirements volatility, requirements quality, requirements entropy and engineering effort and, as a result, enable more effective and efficient requirements engineering in complex system development programs. Time will tell whether or not systems engineers, program managers, contractors and decision-makers who are faced with the difficult task of building our most advanced systems will find the REF useful in their practice.

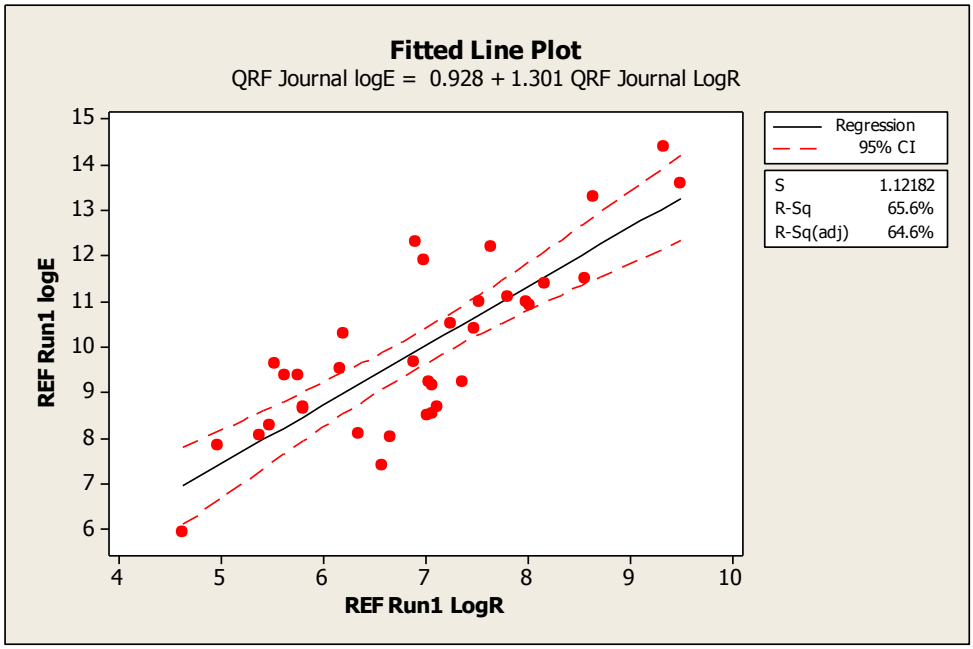
8.2: Areas of Future Research

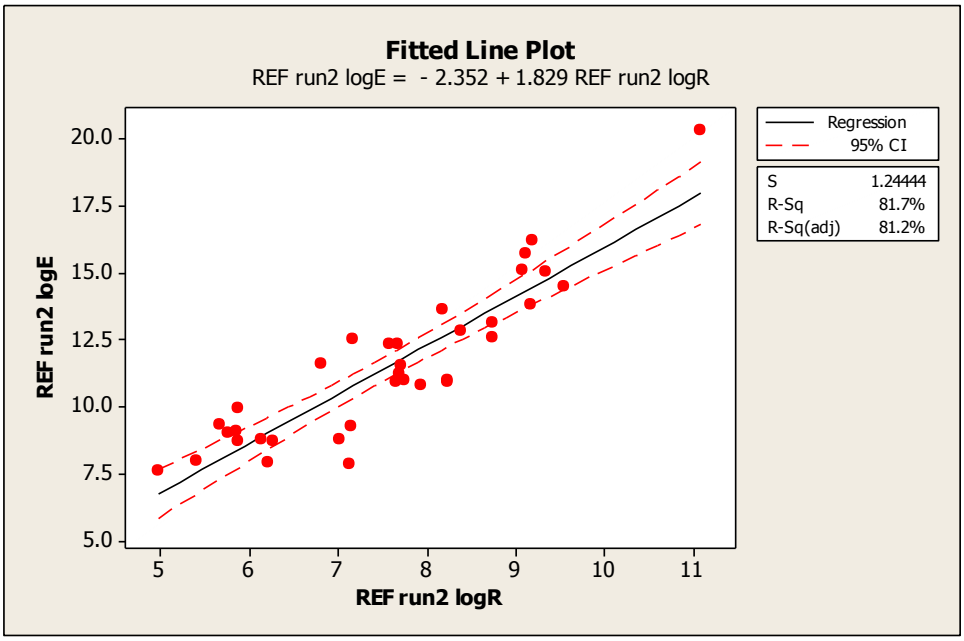
This dissertation is step toward a new complementary way of thinking about the requirements engineering problem and a new SE theory. A strong case can be made that a clear understanding of the problem is necessary before an effective solution can be identified and implemented. COSYSMO and COCOMO are models that may be used for the purpose of gaining insight into important empirical relationships between observed macroscopic behavior and various factors contributing to that behavior. The REF applies principles of information theory and statistical mechanics to SE for the purpose of gaining additional insight into the problem and important relationships that are otherwise difficult to measure. A practical framework based on these principles for estimation, measurement and forecasting of requirements trends and engineering effort is not known to exist. The REF can be readily combined with the Fractional Method or other current methods to better understand process-specific or organization-specific requirements engineering problems and, as a result, support the development of tailored solutions that improve the effectiveness and efficiency of the requirements engineering effort. Future collaborative research with organizations that have completed the calibration process for

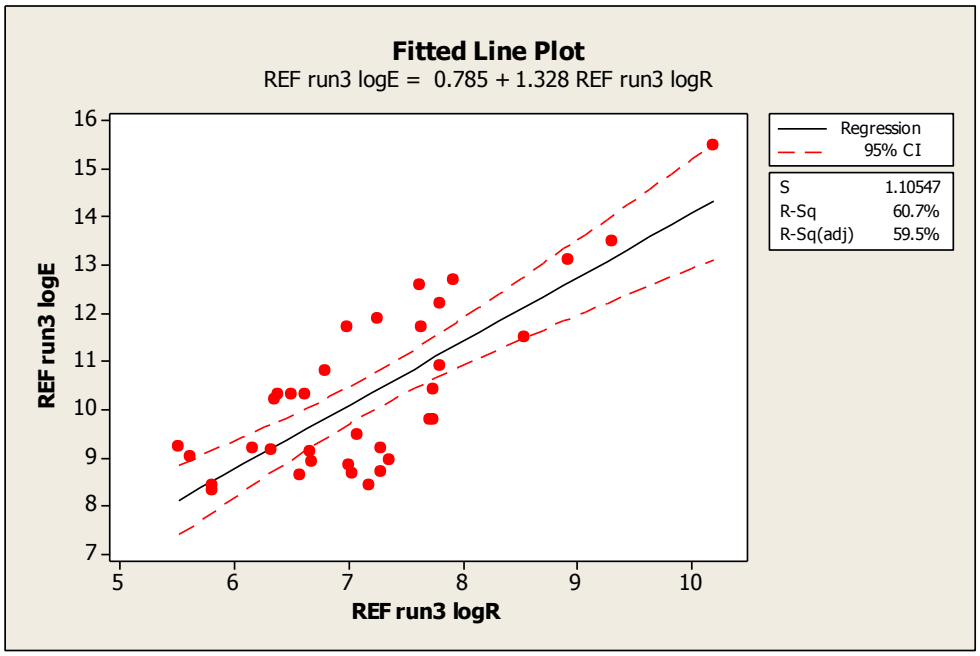
the requirements volatility extension of the COSYSMO model [Peña and Valerdi, 2010] is recommended to investigate the relationship between the REF and COSYSMO for estimating engineering effort as a function of requirements volatility. Further insight is needed into the published engineering effort data to discriminate between the requirements engineering effort and the total engineering effort. There are several mechanisms in the REF that can account for the difference.

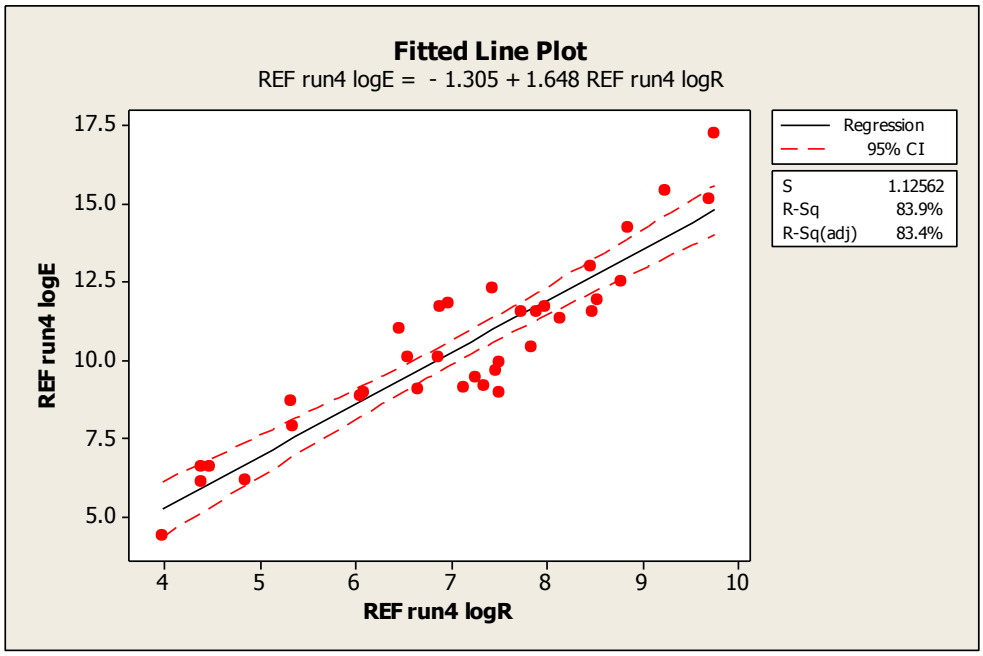
Future research includes a study of the REF estimation of ΔE for the more general case that includes varying degrees of requirements volatility and negative ripple effects that permit a change to an *existing* requirement to result in the loss of one or more quality attributes by one or more *existing* requirements. Future work also includes a study of REF implementation from a complex system acquisition perspective a case study that evaluates the practical utility and effectiveness of the REF during the requirements engineering effort of a system development program.

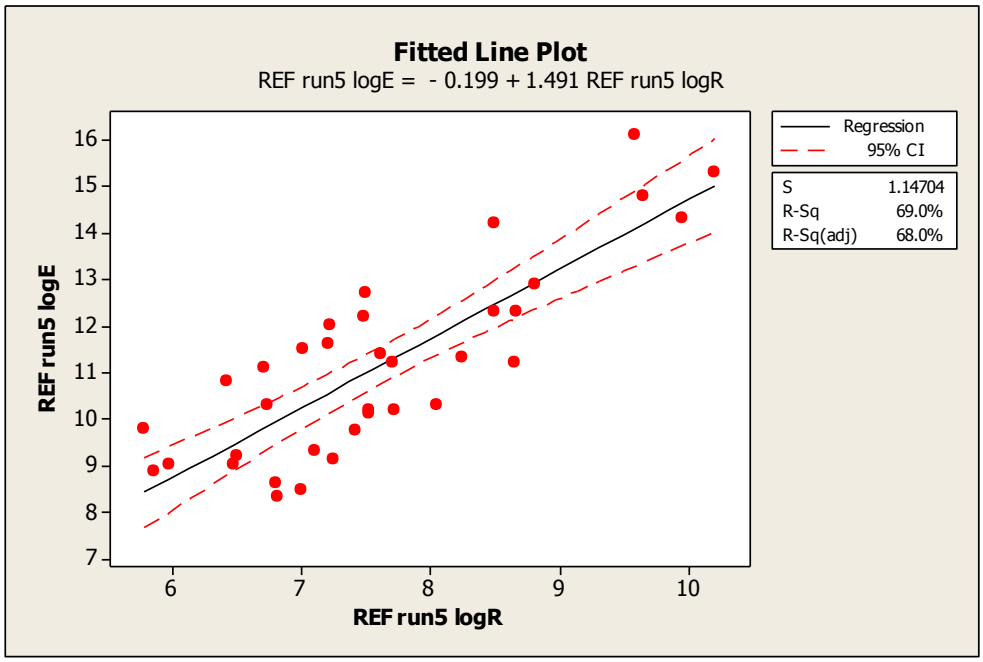
Appendix A: Regression Analysis Plots of Requirements Engineering Effort vs Number of Requirements



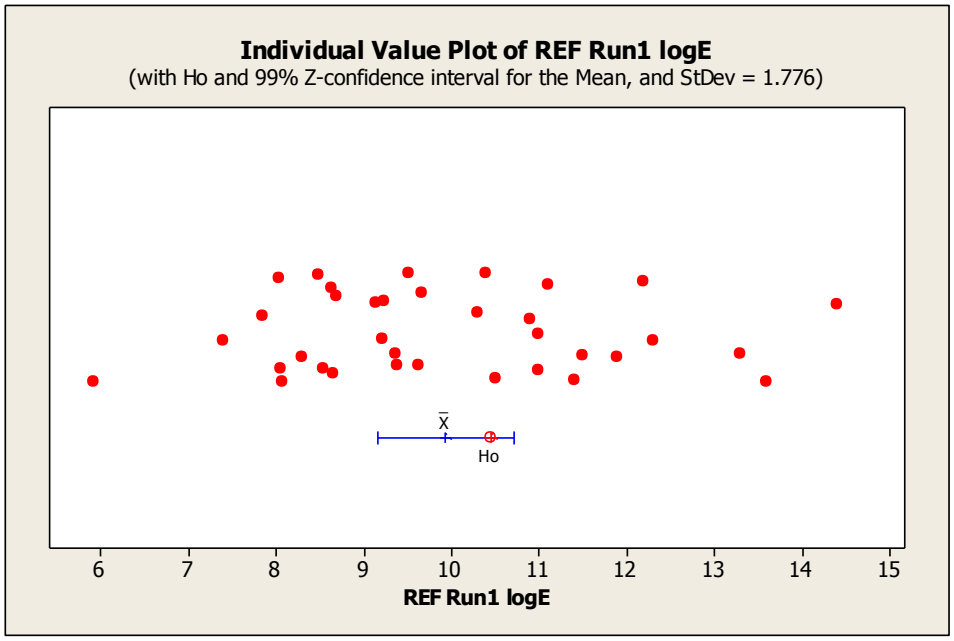


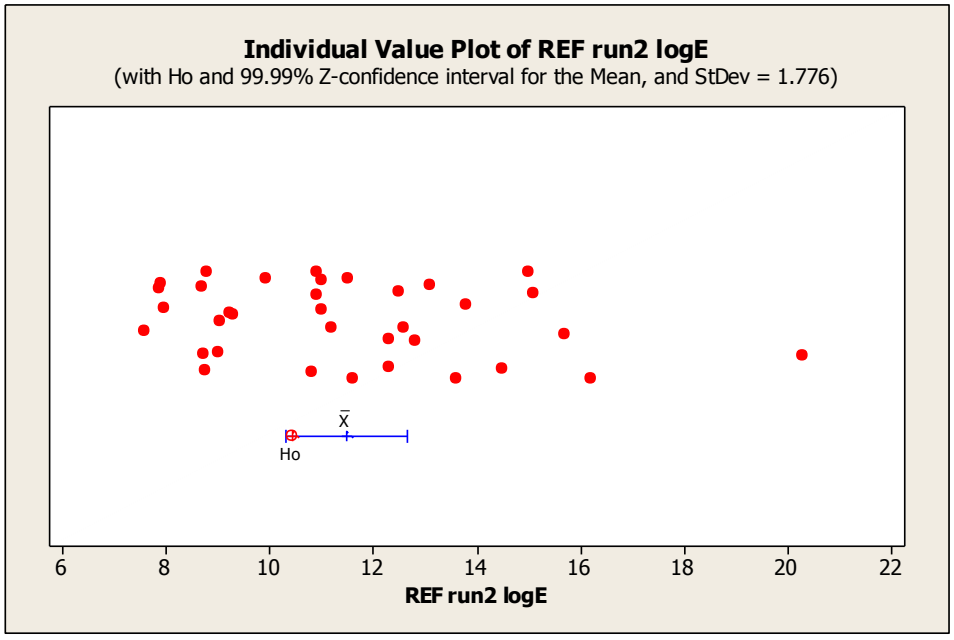

Fitted Line Plot
REF run3 logE = 0.785 + 1.328 REF run3 logR
Regression
95% CI
S 1.10547
R-Sq 60.7%
R-Sq(adj) 59.5%
REF run3 logE
REF run3 logR
16
15
14
13
12
11
10
9
8
7
5
6
7
8
9
10

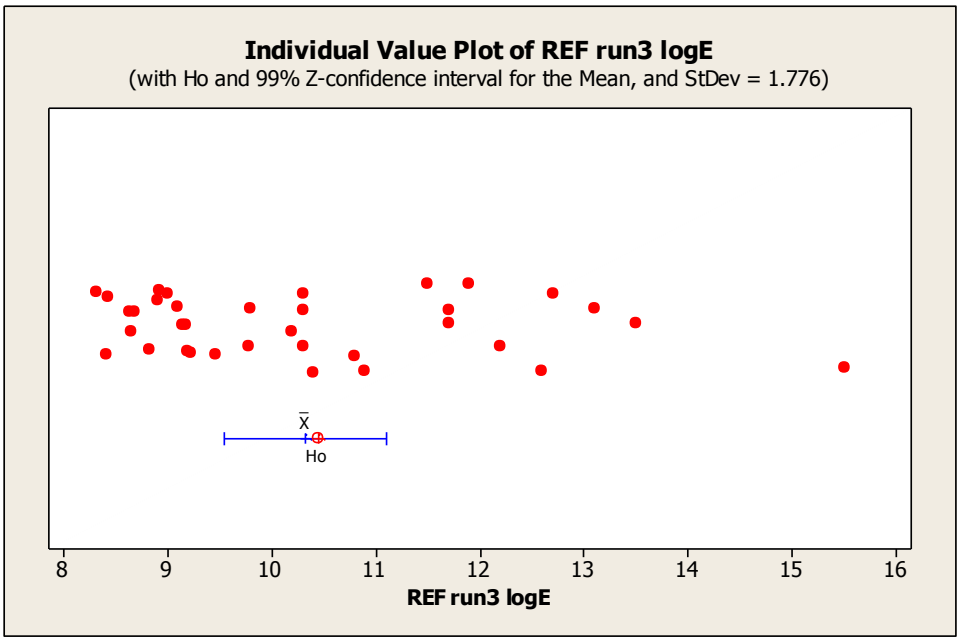

Fitted Line Plot
REF run4 logE = - 1.305 + 1.648 REF run4 logR
Regression
95% CI
S 1.12562
R-Sq 83.9%
R-Sq(adj) 83.4%
REF run4 logE
REF run4 logR
17.5
15.0
12.5
10.0
7.5
5.0
4
5
6
7
8
9
10

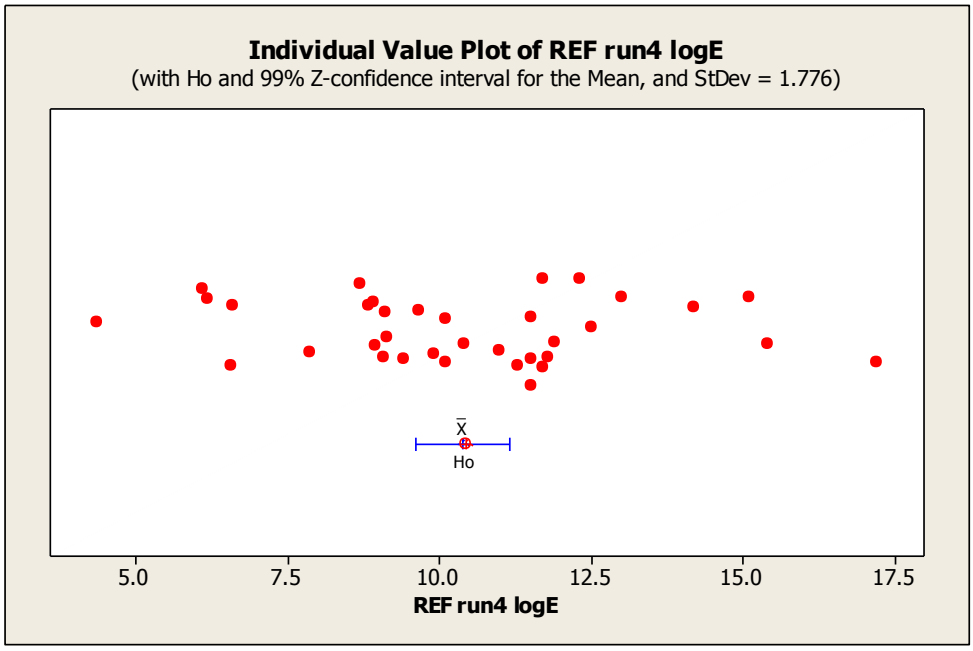

Fitted Line Plot
REF run5 logE = - 0.199 + 1.491 REF run5 logR
Regression
95% CI
S 1.14704
R-Sq 69.0%
R-Sq(adj) 68.0%
REF run5 logE
REF run5 logR
16
15
14
13
12
11
10
9
8
7
6
7
8
9
10

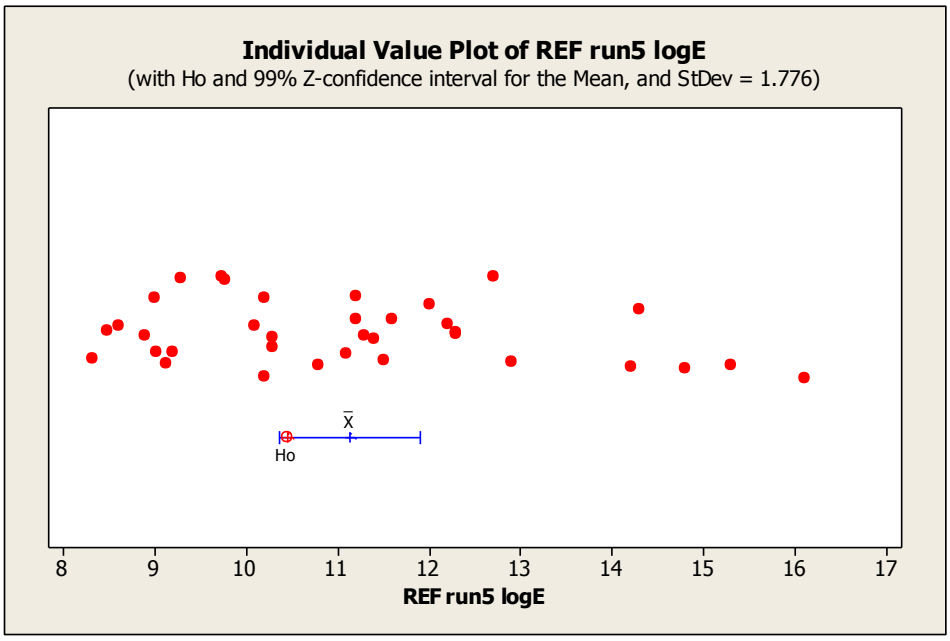
Appendix B: Null Hypothesis Individual Value Plots with Confidence Intervals



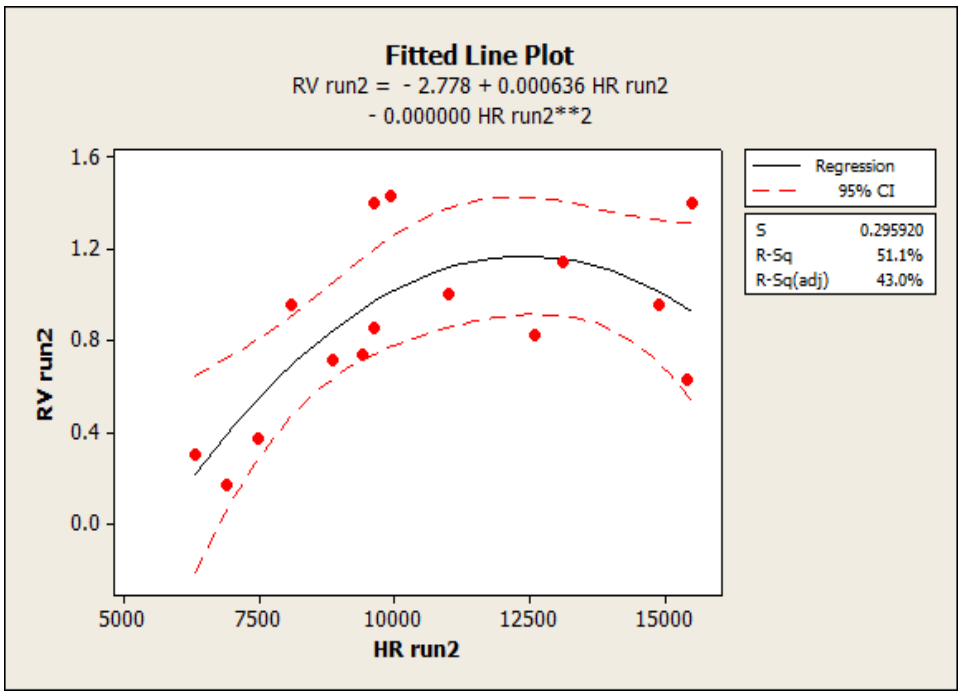


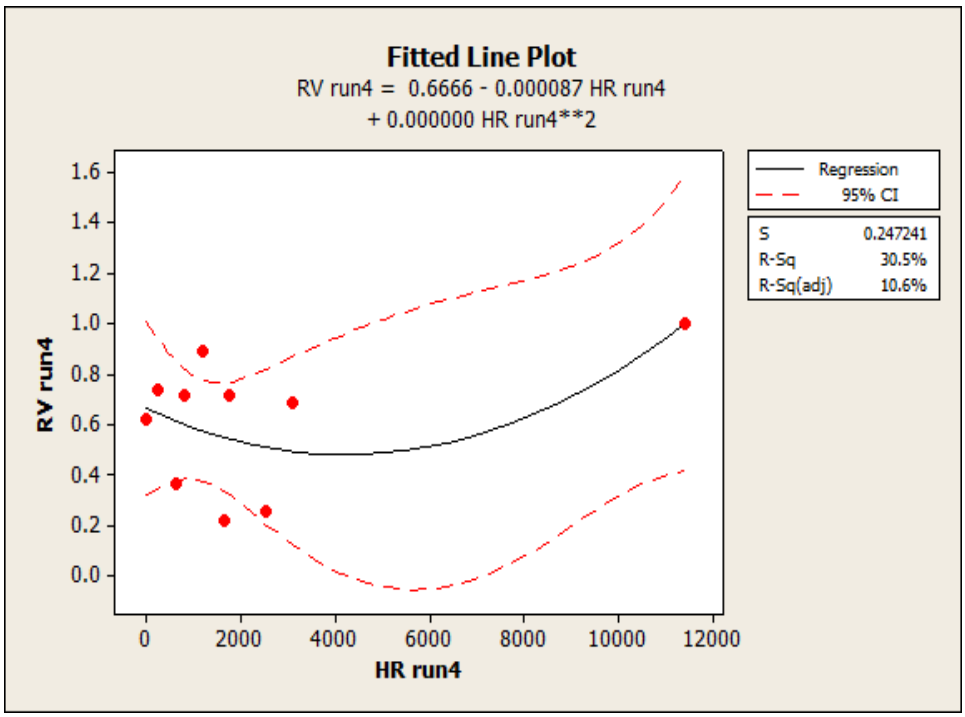

Individual Value Plot of REF run3 logE
(with Ho and 99% Z-confidence interval for the Mean, and StDev = 1.776)
X̄
Ho
8
9
10
11
12
13
14
15
16
REF run3 logE

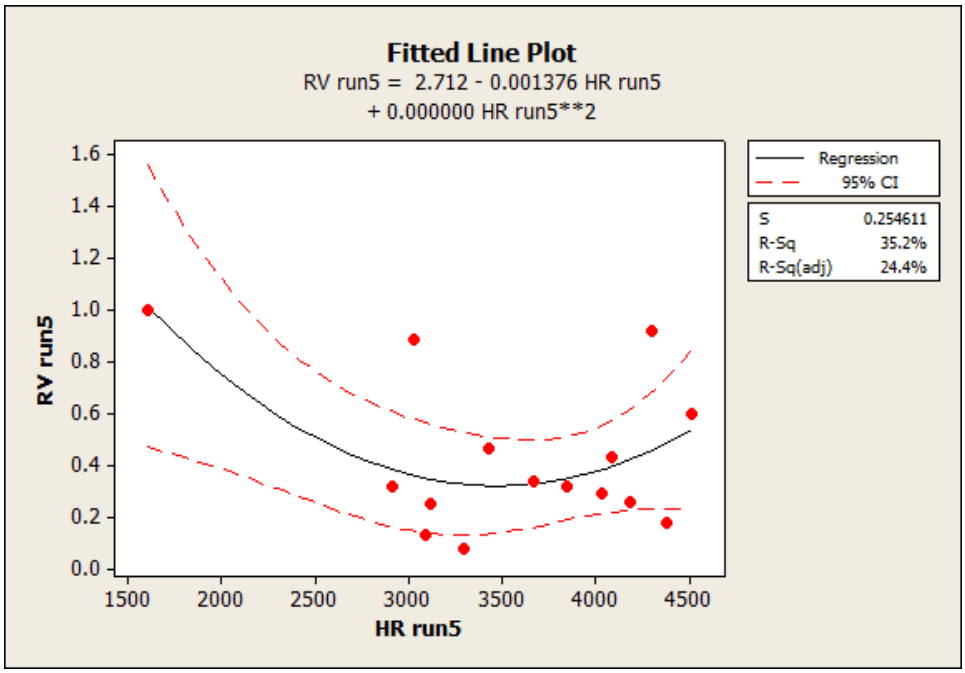

Individual Value Plot of REF run4 logE
(with Ho and 99% Z-confidence interval for the Mean, and StDev = 1.776)
X̄
Ho
5.0
7.5
10.0
12.5
15.0
17.5
REF run4 logE

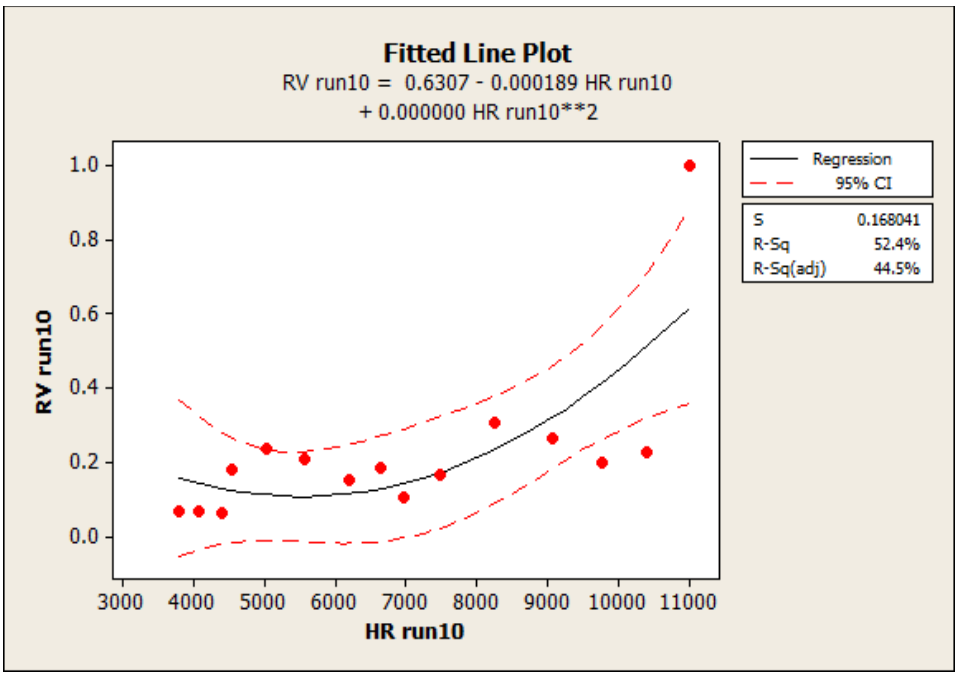

Individual Value Plot of REF run5 logE
(with Ho and 99% Z-confidence interval for the Mean, and StDev = 1.776)
X̄
Ho
8
9
10
11
12
13
14
15
16
17
REF run5 logE

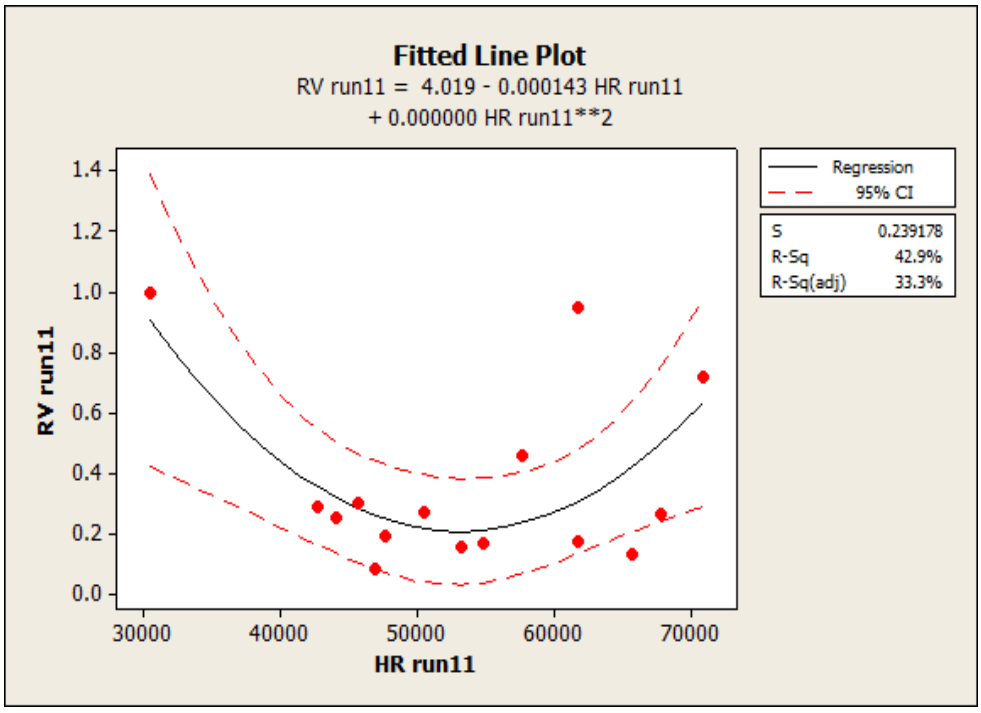
Appendix C: Regression Analysis Plots of Fractional Method Requirements Trend vs Requirements Entropy

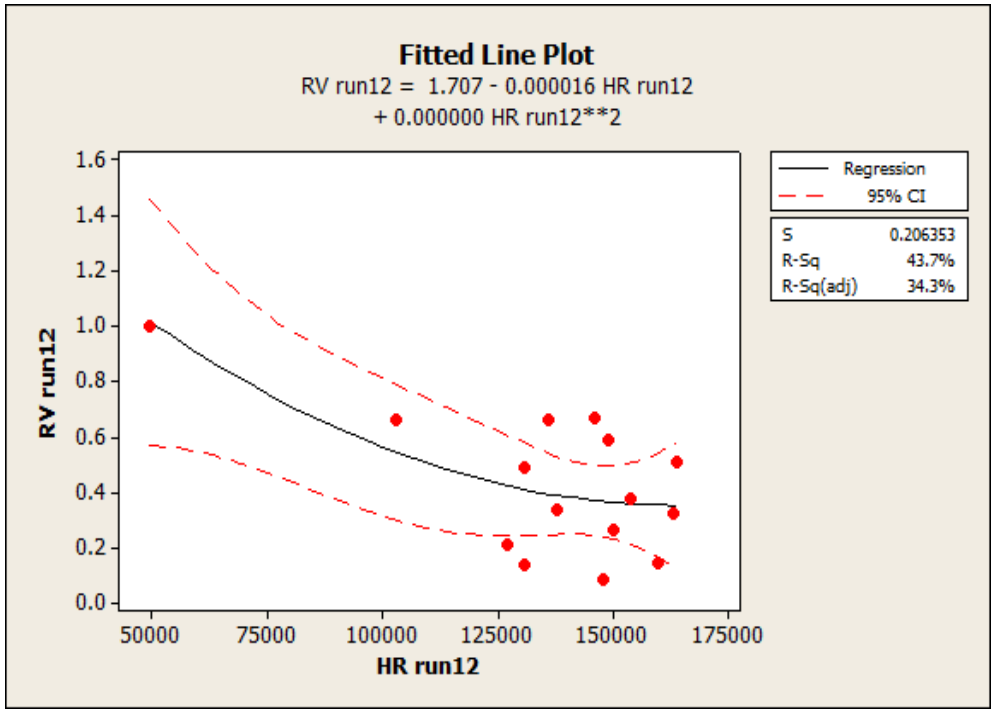





Fitted Line Plot
RV run5 = 2.712 - 0.001376 HR run5
+ 0.000000 HR run5**2
Regression
95% CI
S 0.254611
R-Sq 35.2%
R-Sq(adj) 24.4%
RV run5
HR run5
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
1500
2000
2500
3000
3500
4000
4500


Fitted Line Plot
RV run10 = 0.6307 - 0.000189 HR run10
+ 0.000000 HR run10**2
Regression
95% CI
S 0.168041
R-Sq 52.4%
R-Sq(adj) 44.5%
RV run10
HR run10
1.0
0.8
0.6
0.4
0.2
0.0
3000
4000
5000
6000
7000
8000
9000
10000
11000


Fitted Line Plot
RV run11 = 4.019 - 0.000143 HR run11
+ 0.000000 HR run11**2
Regression
95% CI
S 0.239178
R-Sq 42.9%
R-Sq(adj) 33.3%
RV run11
HR run11
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
30000
40000
50000
60000
70000


Fitted Line Plot
RV run12 = 1.707 - 0.000016 HR run12
+ 0.000000 HR run12**2
Regression
95% CI
S 0.206353
R-Sq 43.7%
R-Sq(adj) 34.3%
RV run12
HR run12
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
50000
75000
100000
125000
150000
175000

Appendix D: Summary of Rounded REF and Fractional Method Requirements Trend Forecasts of When *R* Will Reach Desired State and Become Stable (i.e. No Additional Changes Necessary)

Baseline Measurement Interval Forecast (Mixed)			
Run	REF	Fractional Method	Δ
1	12	26	14
2	18	34	16
3	10	42	32
4	9	>50	41
5	32	16	16
6	27	18	9
7	14	19	5
8	47	22	25
9	21	28	8
10	22	14	8
11	45	15	30
12	27	20	7
13	39	17	22
14	10	48	38
15	36	18	18
16	27	23	4
17	25	18	7
18	16	44	28
19	21	30	9
20	15	42	27
21	39	23	16
22	28	19	9
23	31	20	11
24	17	33	16
			Δbar = 17.33

Appendix E: Engineering Effort Simulation Run Data

Valerdi [2005] Data		Simulation Run1		Simulation Run2		Simulation Run3		Simulation Run4		Simulation Run5		
lnE	lnR	REF lnE	REF lnR	REF lnE	REF lnR	REF lnE	REF lnR	REF lnE	REF lnR	REF lnE	REF lnR	
6.850	4.350	5.921	4.635	10.960	8.252	11.682	6.992	6.581	4.382	9.131	7.247	
7.200	4.375	7.842	4.970	7.860	7.125	8.322	5.805	10.397	7.836	11.633	7.206	
9.375	5.300	8.056	5.366	13.562	8.181	8.999	5.624	11.459	7.886	12.010	7.217	
10.120	5.230	8.285	5.485	10.779	7.938	9.225	5.521	11.522	8.481	9.007	5.974	
10.975	5.900	9.620	5.521	12.253	7.680	10.922	7.809	6.587	4.477	8.894	5.858	
9.500	5.990	9.381	5.624	9.933	5.872	9.098	6.658	11.304	8.150	9.202	6.503	
9.100	6.350	9.362	5.756	11.516	7.721	9.192	6.163	9.669	7.472	8.609	6.796	
10.625	6.400	8.642	5.805	9.242	7.155	8.421	5.808	8.828	6.057	10.340	8.048	
11.875	6.550	8.665	5.808	8.763	6.142	8.649	7.031	10.997	6.455	10.777	6.418	
10.250	6.700	9.506	6.163	12.611	8.740	8.933	7.358	9.423	7.263	9.305	7.115	
8.500	6.770	10.300	6.188	9.309	5.680	8.830	7.012	4.358	3.970	10.227	7.724	
9.050	6.900	8.077	6.342	8.677	6.269	11.510	8.549	8.930	7.508	9.026	6.472	
8.990	6.970	7.404	6.585	9.052	5.846	11.666	7.637	9.078	6.652	16.130	9.584	
7.625	7.030	8.031	6.658	15.003	9.338	8.625	6.585	12.968	8.459	12.279	8.498	
9.850	7.100	9.658	6.888	8.710	5.881	10.216	6.346	17.196	9.755	10.235	7.534	
10.625	7.125	12.283	6.901	8.779	7.034	13.481	9.319	12.301	7.441	9.742	7.425	
8.875	7.150	11.905	6.992	7.956	5.403	13.073	8.928	15.389	9.252	9.773	5.778	
8.700	7.150	8.493	7.012	13.843	9.173	9.470	7.083	11.778	6.980	8.487	7.000	
10.125	7.350	9.225	7.031	7.908	6.223	12.162	7.810	7.864	5.342	14.758	9.649	
11.500	7.440	8.543	7.056	12.298	7.576	10.346	6.616	11.485	7.736	14.188	8.486	
11.125	7.500	9.145	7.075	10.877	8.248	8.905	6.683	6.178	4.852	15.282	10.244	
10.900	7.720	8.681	7.117	12.523	7.181	10.402	7.754	10.055	6.871	11.436	7.616	
9.100	7.800	10.466	7.244	12.759	8.405	8.434	7.178	11.871	8.538	11.193	7.710	
9.100	7.950	9.222	7.358	11.185	7.693	9.153	6.315	10.063	6.550	10.296	6.735	
12.975	8.000	10.430	7.469	14.527	9.561	10.801	6.796	8.940	6.084	12.285	8.655	
11.300	8.080	11.026	7.533	15.094	9.087	15.469	10.247	9.123	7.134	11.541	7.008	
11.500	8.100	12.165	7.637	16.197	9.196	10.295	6.507	8.704	5.323	11.131	6.709	
11.250	8.150	11.061	7.809	20.309	11.147	12.714	7.916	12.463	8.787	12.933	8.807	
12.720	8.730	10.987	7.980	15.709	9.119	9.779	7.724	14.206	8.847	10.051	7.524	
11.650	8.730	10.858	8.023	13.096	8.747	12.640	7.631	9.916	7.496	14.299	9.947	
11.300	8.770	11.424	8.166	11.559	6.821	8.695	7.290	11.692	6.880	8.318	6.812	
14.125	9.050	11.506	8.549	10.860	7.664	9.793	7.754	9.137	7.351	11.177	8.655	
12.125	9.100	13.342	8.641	10.965	7.754	10.302	6.375	15.113	9.696	12.699	7.496	
13.800	9.580	14.362	9.315	7.573	4.970	11.917	7.258	11.682	7.989	11.266	8.236	
12.950	9.770	13.574	9.486	9.013	5.756	9.171	7.292	6.111	4.394	12.248	7.481	
10.447	7.290	9.927	6.920	11.465	7.559	10.323	7.182	10.382	7.096	11.140	7.605	**Mean**
1.775	1.326	1.881	1.173	2.872	1.422	1.733	1.024	2.761	1.535	2.029	1.133	**Sigma**

References

Air Force Center for Systems Engineering (AFCSE), Systems engineering case studies: synopsis of learning principles, 17 May 2010.

Y. Akao, Quality function deployment: integrating customer requirements into product design, Productivity Press, Cambridge, MA, 2004.

Y. Akao and G. Mazur, The leading edge in QFD: past, present and future, The International Journal of Quality & Reliability Management 2003, 20(1).

G. Anthes, No more creeps! Are you a victim of creeping user requirements? Computerworld 1994, 28(18):107-110.

W. Ashby, Requisite variety and its implications for the control of complex systems, Cybernetica 1958, 1(2):83-99.

N. Azizian, T. Mazzuchi, S. Sarkani, D. Rico, A framework for evaluating technology readiness, system quality, and program performance of U.S. DoD acquisitions, Systems Engineering 2011, 14(4):410-426.

A.T. Bahill and B. Gissing, Re-evaluating systems engineering concepts using systems thinking, IEEE Trans on Systems, Man and Cybernetics, Part C: Apps and Revs 1998, 28(4):516-527.

A.T. Bahill and S.J. Henderson, Requirements development, verification, and validation exhibited in famous failures, Systems Engineering 2005, 8(1):1-14.

K. Bailey, Restoring order: relating entropy to energy and information, Sys Research 1987, 4(2):83-92.

Y. Bar-Yam, Unifying principles in complex systems, in: converging technology (NBIC) for improving human performance, Roco, M.C., Bainbridge, W.S., Eds. 2003.

H.A. Bashir and V. Thomson, Estimating design complexity, Journal of Engineering Design 1999, 10(3):247-257.

B. Blanchard and W. Fabrycky, Systems Engineering & Analysis, Prentice Hall, 2005.

B. Boehm, Some future trends and implications for systems and software engineering processes, Systems Engineering 2006, 9(1):1-19.

B. Boehm, R. Valerdi and E. Honour, The ROI of systems engineering: some quantitative results for software-intensive Systems, Systems Engineering 2008, 11(3):221-234.

L. Boltzmann, Vorlesungen uber Gastheorie, 2 vols. Leipzig: Barth, 1896, 1898. Translated into English by S. G. Brush, Lectures on Gas Theory, Cambridge Univ. Press, London, 1964.

J.D. Bottelson, Requirements and cost stability: a case study of the F/A-18 Hornet program, Acquisition Research Journal, DAU, October 2011.

R. Botta, Z. Bahill, and T. Bahill, When are observable states necessary? Systems Engineering 2006, 9(3):228-240.

R. Botta and A.T. Bahill, A prioritization process, Engineering Management Journal 2007, 19(4):20-27.

J.H. Brill, Systems engineering – a retrospective view. Systems Engineering 1998, 1(4):258-266.

L. Brillouin, Physical Entropy and Information II, Jour of Applied Physics 1951, 22(3):338-343.

L. Brillouin, Science and information theory, Academies Press, Inc. London, 1962.

J.B. Brissaud, The meanings of entropy, Entropy 2005, 7(1):68-96.

C.N. Calvano and P. John, Systems engineering in an age of complexity. Systems Engineering 2004, 7(1):25-34.

D.J. Campbell, Task complexity: A review and analysis, Academy of Management Review 1988, 13(1):40-52.

M.G. Ceruti and S.H. Rubin, Infodynamics: analogical analysis of states of matter and information. Information Sciences 2007, 177:969-987.

Chairman of the JCS Manual for Operation of JCIDS, CJCSM 3170.01C, May 2007.

P. Checkland, Systems thinking, systems practice, John Wiley & Sons Ltd 1999.

D. Clark, D. Howell, C. Wilson, Improving naval shipbuilding project efficiency through rework reduction. Master's Thesis, Naval Postgraduate School, NSN 7540-01-280-5500, 2007.

D. Clausing and L. Cohen, Recent developments in QFD in the United States, presented at "Gaining Competitive Advantage by Design", Institution of Mechanical Engineering Conference, Coventry, UK, March 23 and 24, 1994.

R.J. Costello, Metrics for requirements engineering, M.Sc. Thesis, CSU, Long Beach, 1994.

R.J. Costello and Dar-Baiu Liu, Metrics for requirements engineering. Journal of Sys Software 1995, 29:39-63.

J.J. Cristiano, J.K. Liker, C.C. White, Key factors in successful application of Quality Functional Deployment (QFD), IEEE Trans on Eng Management 2001, 48(1):81-95.

E. Curado and F. Nobre, On the stability of entropic forms, Physica A 2004, 335:94-106.

DAG, Defense Acquisition Guidebook 2011, https://acc.dau.mil.

DAU, Systems engineering fundamentals, Defense Acquisition University Press, Fort Belvoir, Virginia 22060-5565, 2001.

DAU, Directive-Type Memorandum (DTM) 09-027 Implementation of the Weapon Systems Acquisition Reform Act of 2009 (Public Law 111-23), 4 December 2009.

DAU, Committed life-cycle costs versus time, 1993.

A. Davis, S. Overmyer, K. Jordan, J. Caruso, F. Dandashi, A. Dinh, G. Kincaid, G. Ledeboer, P. Reynolds, P. Sitaram, A. Ta, M. Theofanos, Identifying and measuring quality in a software requirements specification, Proc 1st Int'l SW Metric Symp, IEEE, Balt MD, May 21-22, 1993.

A. Davis, Software requirements: objects, functions, and states, Prentice Hall, Englewood Cliffs, NJ, 1993.

H. Davidz, Enabling systems thinking to accelerate the development of senior systems engineers. Dissertation, 2006.

E.B. Dean, Quality function deployment for large systems, Proceedings of the 1992 International Engineering Management Conference, Eatontown, NJ, 25-28 October.

R. Dewar, Information theory explanation of the fluctuation theorem, maximum entropy production and self-organized criticality in non-equilibrium stationary states, J. Phys. A: Math. Gen 2003, 36:631-651.

I. Dincer and C. Yunus, Energy, entropy and exergy concepts and their roles in thermal engineering. Entropy, 2001, 3:116-149.

DOD Instruction 5000.02, Operation of the Defense Acquisition System, 2008.

J.P. Dowling and G.J. Milburn, Quantum technology: the second revolution, Phil Trans R Soc Lond A 2002, 361:1655-1674.

H. Eisner, Essentials of project and systems engineering management, 2nd Edition John Wiley & Sons, 2002.

S. Ferreira, J. Collofello, D. Shunk, G. Mackulak, Understanding the effects of requirements volatility in software engineering by using analytical modeling and software process simulation, The Journal of Systems and Software 2009, 82:1568-1577.

M. Frank, Engineering systems thinking and systems thinking, Systems Engineering 2000, 3(3):163-168.

A.D. Franklin, Extrapolation errors in linear regression, Ancient TL 1986, 4:31-35.

S.G. Gambhir, An investigation of facilitator-assisted and CONOPS-based requirements elicitation methods using a 2×2 factorial experimental design, Systems Engineering 2001, 4(4):272-286.

GAO, Defense acquisitions: assessments of selected weapons programs, report to congressional committees, 2009, GAO-09-326SP.

GAO, Defense acquisitions: assessments of selected weapons programs, Report to Congressional Committees, 2011, GAO-11-233SP.

K. Gill, 7th Annual Conf on Systems Engineering Research 2009 (CSER 2009) Loughborough University – 20th – 23rd April 2009, Systems Thinking or Systems Engineering

D. Ginn and M. Zairi, Best practice QFD application: internal/external benchmarking approach, International Journal of Quality & Reliability Mgmt 2005, 22(1):38-58.

A. Hari, J.E. Kasser, M.P. Weiss, How lessons learned from using QFD led to the evolution of a process for creating quality requirements for complex systems, Systems Engineering 2007, 10(1):45-63

R. Hartley, Transmission of information, Bell System Technical Journal 1928, 7:535-563.

T.L. Hill, An introduction to statistical thermodynamics. Addison-Wesley Pub. Co, Inc., 1960.

E.C. Honour, Understanding the value of systems engineering, Proc INCOSE Int'l Symp, Toulouse, France, 2004.

E.C. Honour and R. Valerdi, Advancing an ontology for systems engineering to allow for consistent measurement, Proceedings of the Conference on Systems Engineering Research, Los Angeles, CA 2006, pages 1-12.

E. Hull, K. Jackson, J. Dick, Requirements engineering, Springer 3rd Edition 2011.

INCOSE, INCOSE Systems Engineering Handbook v3.2.1 2011.

INCOSE, SE Leading Indicators Guide v2.0 INCOSE-TP-2005-001-03, 2010.

INCOSE, SE Leading Indicators Guide v1.0, 2007.

INCOSE, SE Handbook v.2.a TP-2003-016-02, 2004.

INCOSE, Systems Engineering Vision 2020 TP-2004-004-02 September, 2007.

INCOSE, Systems Engineering Measurement Primer, INCOSE-TP-2010-005-02, 2010.

E.T. Jaynes, Information theory and statistical mechanics, Phys Review 1957, 106(4):620-630.

E.T. Jaynes, The minimum entropy production principle, Ann. Rev. P. Chem 1980, 31:579-601.

J.N. Kapur, Four families of measures of entropy, Indian J. Pure Appl. Math. 1986, 17(4):429-449, April 1986.

J. Kim and D. Wilemon, Sources and assessment of complexity in NPD projects, R&D Mgmt 2003, 33(1):15-30

T. Kivinen, Applying QFD to improve the requirements and project management in small-scale project. M.Sc. Thesis, U. of Tampere, 2008.

A. Kludze, Engineering of complex systems: the impact of systems engineering at NASA. Dissertation UMI Number: 3144615, 2004.

G.P. Kulk and C. Verhoef, Quantifying requirements volatility effects, Science of Computer Programming 2008, 72:136-175.

A. Lamsweerde, Requirements engineering. John Wiley & Sons Ltd., 2009.

J.A. Lane and B. Boehm, When does requirements volatility stop all forward progress? Practical software and systems measurement user's group conference, Golden Colorado, July 2007.

A. Loconsole and J. Börstler, An industrial case study on requirements volatility measures, Proc of the Asia Pacific software engineering conference, Taipei, Taiwan, Dec 2005, 249-256.

S. Maadawy and A. Salah, Measuring change complexity from requirements: A proposed methodology 2012, IMACST 3(1):1-25.

R.O. Mason, Measuring information output: a communication systems approach, Information and Management 1978, 1:219-234.

S.E. Massen and C.P. Panos, A link of information entropy and kinetic energy for quantum many-body systems, Physics Letters A 280.1 2001:65-69.

Mathworks Inc. MATLAB® 32-bit student version R2010a.

J.C. Maxwell, On Faraday's Lines of Force. Cambridge Philosophical Society Transactions 10 1864, 27-82, Delivered on 10 December 1855 and 11 February 1856, from Howard, 2007.

G. Mazur, President QFD Institute, 2011, Private communication.

Miller, G.A., The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information, Psychological Review 1956, 63:81-97; reprinted in Psychological Review 1994, 101:343-352.

R. Miller, S. Floricel, D.R. Lessard, The strategic management of large engineering projects, MIT Press, 2000.

S. Mizuno and Y. Akao, Quality functional deployment: a company-wide quality approach, JUSE Press, Tokyo, 1978.

P. Mohagheghi and R. Conradi, An empirical investigation of software reuse benefits in a large telecom product, ACM TOSEM 2008, 17(3):1-34.

P. Mohagheghi, R. Conradi, O. Killi and H. Schwarz, An empirical study of software reuse vs defect-density and stability, Proceedings of the 26th International Conference on Software Engineering (ICSE '04), IEEE 2004, 11 pages.

NRC, Pre-milestone A and early-phase systems engineering: a retrospective review and benefits for future Air Force systems acquisition, National Academies Press, Washington, D.C., 2008.

H. Nyquist, Certain factors affecting telegraph speed, Bell System Technical Journal 1924, 3:324-346.

Oxford English Dictionary (OED) 2013, http://oxforddictionaries.com

M. Peña and R Valerdi, Characterizing the impact of requirements volatility on systems engineering effort, COSYSMO constructive systems engineering cost model, 25th Annual COCOMO Forum, November 2010.

K. Pohl, Requirements engineering fundamentals, principles and techniques, Springer-Verlag Berlin Heidelberg 2010.

QFD Institute, 2011, www.qfdi.org website information.

R. Rajnish, H. Dev, R. Vyas, Writing quality requirements (SRS): an approach to manage requirements volatility, Indian Jour of Computer Science and Engineering 2010, 1(1):28-37.

E. Rechtin and M. Maier, The art of systems architecting, CRC Press, New York, 1997.

C.M. Reed and N.I. Durlach, Note on information transfer rates in human communication, Presence: Teleoperators and Virtual Environ 1998, 7(5):509-518.

A. Renyi, On measures of entropy and information, J. Neyman, ed. 4th Berkeley symposium on mathematical statistics and probability. Berkeley, 1961:547-561.

D. Rhodes, R. Valerdi, G. Roeddler, Systems engineering leading indicators for assessing program and technical effectiveness, Systems Engineering 2009, 12(1):21-35.

F. Richtmyer, E. Kennard, J.N. Cooper, Introduction to modern physics, McGraw-Hill, 1969.

M.M. Ricker, Requirements specification understandability evaluated with cohesion, context, and coupling, Ph.D. Dissertation, George Mason University, 1995.

Rouse W.B., Complex engineered, organizational and natural systems, Systems Engineering 2007, 10(3):260-271.

A.P. Sage, Complex adaptive systems, in AccessScience@McGraw-Hill, DOI 10.1036/ 1097-8542. YB020185, 2002.

A.P. Sage and W.B. Rouse, Handbook of Systems Engineering and Management 2009, Wiley 2nd Edition 2009.

A.P. Sage, Systems Management for Information Technology and Software Engineering, John Wiley & Sons, 1995

T. Sawada, Quality deployment and product planning, Proceedings of the 9th Annual Convention of the Japanese Society for Quality Control 1979, pp. 27-30.

M.J. Schroeder, An alternative to entropy in the measurement of information, Entropy 2004, 6:388-412.

SEI CMMI-DEV v1.3, Improving processes for developing better products and services, 2010.

C. Seife, Decoding the universe: how the new science of information is explaining everything

in the cosmos, from our brains to black holes, Viking Press, 2007.

P.M. Senge, The fifth discipline: the art and practice of the learning organization. Currency Doubleday, New York, 1990.

C.E. Shannon, A Mathematical Theory of Communication, Mobile Comp & Com Rev 1948, 5(1):3-55.

C.E. Shannon and W. Weaver, The mathematical theory of communication, U. Ill. Press, 1949.

C.E. Shannon, Zero error capacity of noisy channels, *IEEE Trans* Information Theory 1956, IT-2, pp. 8-19.

S.A. Sheard, Principles of complex systems for systems engineering, INCOSE Symposium, San Diego, CA 24-28 June 2007.

S.A. Sheard and A. Mostashari, Principles of complex systems for systems engineering, Systems Engineering 2009, 12(4):295-311.

M.J. Simpson and J.J. Simpson, Formal, theoretical aspects of systems engineering: comments on "principles of complex systems for systems engineering" Systems Engineering 2010, 13(2):204-207.

G.E. Stark, P. Oman, A. Skillicorn, A. Ameele, An examination of the effects of requirements changes on software maintenance releases, Journal of Systems Software Maintenance: Research and Practice 1999, 11:293-309.

L. Szilard, On the decrease of entropy in a thermodynamic system by the intervention of intelligent beings, 1929 translated Behav Sci 1964, 9(4):301.

L.D. Thomas and R.A. Mog, A quantitative metric of system development complexity: validation results, INCOSE International Symposium, Vancouver, BC, 1998.

M. Tribus and E. McIrvine, Energy and Information, Scientific American 1971, 179-184.

R. Valerdi, The Constructive SE Cost Model (COSYSMO), Dissertation, 2005.

R. Valerdi, B. Boehm and D. Reifer, COSYSMO: A constructive systems engineering cost model coming of age, INCOSE 2003 Annual International Symposium Proceedings, 70-82.

H. Walach and N. von Stillfried, Generalised quantum theory, basic idea and general intuition: a background story and overview, Axiomathes 2011 doi: 10.1007/s10516-010-9145-5.

Weapon Systems Acquisition Reform Act (WSARA), Public Law 111-23, 2009.

A. Wehrl, General properties of entropy, Reviews of Mod Phys 1978, 50(2):221-260.

K.E. Wiegers, More About Software Requirements, Microsoft Press, Redmond, WA, 2006.

R. Xu, Y. Xue, P. Nie, Y. Zhang, D. Li, Research on CMMI-based Software Process Metrics, IEEE Proc 1st IMSCCS'06 0-7695-2581-4/06, 2006.

W.H. Zurek, Complexity, entropy and the physics of information, Addison-Wesley Publishing Company, 1990.