Value Centric Approach to Target System Modularization Using Multi-Attribute Tradespace Exploration and Network Measures of Component Modularity

by

Henry H. Roark, III

B.S., Physics, Georgia Institute of Technology (1995)

Submitted to the System Design and Management Program
in partial fulfillment of the requirements for the degree of

Master of Science in Engineering and Management

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2012

© Henry H. Roark, III, MMXII. All rights reserved.

The author hereby grants to MIT permission to reproduce and distribute publicly paper and electronic copies of this thesis document in whole or in part.

Author .....
System Design and Management Program
May 1, 2012

Certified by .....
Adam M. Ross
Research Scientist, Engineering Systems
Thesis Supervisor

Accepted by .....
Patrick C. Hale
Director, System Design and Management Program

Value Centric Approach to Target System Modularization Using Multi-Attribute Tradespace Exploration and Network Measures of Component Modularity

by

Henry H. Roark, III

Submitted to the System Design and Management Program
on May 1, 2012, in partial fulfillment of the
requirements for the degree of
Master of Science in Engineering and Management

Abstract

Deciding where to modularize a system can have long-term impact on that systems value over its entire lifecycle. The modularity of a system can impact the systems flexibility, evolvability, scalability, mass, costs, and development schedule. Making these modularization decisions is a key job of the system architect. There exists a need to provide the system architect tools that will help focus modularization efforts on the areas of the system that are most likely to provide value to stakeholders of the system. Using a terrestial vehicle as a case study, an approach is developed that links component modularity to system design variables which are likely to change levels. The approach utilitizes Multi-Attribute Tradespace Exploration, Multi-Epoch Analysis, and network measures of component modularity to identify components which are most likely to need to change as well as the components ability to make a modularity change. It is found that the tools utilized can be successfully linked to provide early development phase information about value-centric system modularizations; the approach does require a network representation of the system earlier in the design cycle than it may be typically available. Using explicit knowledge, the approach developed can focus designers modularization efforts on the elements of the system that may need to change to accommodate changes in decision makers preferences, use contexts, or epoch contexts. The approach developed can aid the system architect modularizing a system so as to create a high amount of leverage for a systems stakeholders.

Thesis Supervisor: Adam M. Ross
Title: Research Scientist, Engineering Systems

Acknowledgments

I first want to thank my wife Stephanie for unwaveringly supporting me in this SDM endeavor. Various people from work and school have commented on how well I seem to be doing juggling family, work, and school. I know that in reality you absorbed most, if not all, of the variability over the last two years and, indeed, you are the one that has been doing all the juggling. I do not know how I will ever make up for all the lost time, but I look forward to starting now. Along with Stephanie I want to let my three children, Brandon, Tristan, and Emma know that you have all done an incredible job dealing with my absence, even when I was physically there. You were most supportive, patient, and loving and I thank you from the deepest of my heart. I love all of you.

Next I would like to thank my support network of fellow SDMers. You have been exceptional for being there to bounce ideas off, for doing problem sets, and in general being there when it came time to get over challenges and share successes. I especially want to thank Candice Engler, (soon to be Dr.) John Helfrich, and Cdr. Dennis Evans (USCG). Best of luck to all of you.

For this thesis work, thank you to Adam Ross for all the support you have given me. The feedback, drawing out assumptions and implicit statements, and helping me move my research forward is greatly appreciated. You are the shoulders I am standing upon. Thank you to Donna Rhodes and Adam for providing me a great place, both in terms of physical comfort and intellectual stimulation, at SEAri to do this work.

I would like to thank Alice, Howard, and Mano, for sponsoring my SDM journey and helping me balance work and school. At last but not least, thank you to John Deere for sponsoring this research.

Contents

1 Introduction **9**
- 1.1 Motivation . . . 9
- 1.2 Value . . . 10
- 1.3 Scope . . . 11
- 1.4 Methodology . . . 11

2 Literature Review **12**
- 2.1 Overview . . . 12
- 2.2 Multi-Attribute Tradespace Exploration . . . 12
  - 2.2.1 MATE Overview . . . 12
  - 2.2.2 RSC as a process formalization . . . 15
  - 2.2.3 Role in enabling evolutionary acquisition . . . 17
- 2.3 Design Structure Matrix . . . 17
  - 2.3.1 DSM Overview . . . 17
  - 2.3.2 DMM and MDM Overview . . . 18
- 2.4 Modularity and Modularization . . . 19
  - 2.4.1 Definitions of Modules, Modularity, and Modularization . . . 19
  - 2.4.2 Proposed benefits of and challenges to modularity . . . 20
  - 2.4.3 Existing modularization approaches . . . 20
  - 2.4.4 Prior MATE studies . . . 21
- 2.5 Research Objectives / Questions . . . 22

3 Design Approach Proposal **24**
- 3.1 Approach Goals . . . 24
- 3.2 Approach Roadmap . . . 25
  - 3.2.1 Component Modularity Analysis . . . 25
  - 3.2.2 Delta-Design Vector Development . . . 27
  - 3.2.3 Approach Flow . . . 28
  - 3.2.4 Connection to Architecture Synthesis . . . 30
- 3.3 Summary . . . 31

4 Case Example **32**
4.1 Overview . . . 32
4.2 Target System - LifeTrac . . . 33
4.3 Structural Model . . . 35
4.3.1 Component DSM Model . . . 35
4.3.2 Component Modularity Metrics . . . 38
4.4 Value Driven Design Formulation . . . 40
4.4.1 Decision Makers, Utility Functions, and Costs . . . 40
4.4.2 Design Variables Mappings . . . 45
4.5 Design Variable to Component Mapping . . . 47
4.6 Tradespace Exploration . . . 50
4.6.1 Design to Utility and Cost Mapping Models . . . 50
4.6.2 Resulting Tradespaces . . . 51
4.7 delta-DV Development . . . 60
4.8 Modularity Analysis . . . 66
4.8.1 Component by component analysis . . . 67
4.8.2 Connection to Tradespace Network Analysis . . . 72
4.9 Alternative Approach Flow . . . 75
4.10 Discussion . . . 76
4.11 Case Summary . . . 79

5 Discussion and Implications **81**
5.1 MATE output to target modularization . . . 81
5.2 Research Objectives / Questions . . . 82
5.3 Limitations . . . 83
5.4 Future Research . . . 84
5.5 Final Thoughts . . . 85

A LifeTrac Design Structure Matrix **86**

B LifeTrac Modularity Metrics **91**

C LifeTrac Modularity Metrics for Selected Components **93**

D LifeTrac System Model **97**

Bibliography **101**

List of Figures

2-1 High-level flow of a MATE analysis, from (McManus et al., 2005) 13
2-2 High-level flow of a RSC analysis, from (Ross et al., 2009a) 16
2-3 Taxonomy of DSM types, from (Browning, 2001) 18

3-1 Proposed additions to RSC for value-centric component modularity design . 26
3-2 Proposed approach steps to value centric approach to modularization 29
3-3 Proposed connection to architecture synthesis 30

4-1 Photograph of LifeTrac, from (Open-Ecology-Project, 2011d) 34
4-2 Component DSM of LifeTrac 37
4-3 Single attribute utilities $U^i(X^i)$ for skid steer epoch 42
4-4 Moldboard plowing, from http://tinyfarmblog.com/breaking-new-ground/ . 43
4-5 Single attribute utility $U^i(X^i)$ for tractor epochs 44
4-6 Skid steer epoch 1 tradespace 52
4-7 Skid steer epoch 1 tradespace, by hydraulic power 53
4-8 Skid steer epoch 1 tradespace, by bucket width 54
4-9 Skid steer epoch 2 tradespace 54
4-10 Skid steer epoch 2 tradespace, by hydraulic power 55
4-11 Skid steer epoch 2 tradespace, by bucket width 55
4-12 Tractor epoch 3 tradespace 56
4-13 Tractor epoch 3 tradespace, by engine power 57
4-14 Tractor epoch 4 tradespace 58
4-15 Tractor epoch 4 tradespace, by engine power 59
4-16 Normalized Pareto Traces by design 61
4-17 fuzzy-Normalized Pareto Traces by design 63
4-18 Key to interpreting modularity plots 67
4-19 Modularity of skid steer bucket width modules 69
4-20 Modularity of skid steer engine power modules 69
4-21 Modularity of tractor width modules 71
4-22 Modularity of tractor engine power modules 72
4-23 Plaform, variants, users, and available transitions for modular LifeTrac . . . 80

A-1 Component DSM of LifeTrac - Upper-left Quadrant 87
A-2 Component DSM of LifeTrac - Upper-right Quadrant 88
A-3 Component DSM of LifeTrac - Lower-left Quadrant 89
A-4 Component DSM of LifeTrac - Lower-right Quadrant 90

List of Tables

4.1 Context Variables 41
4.2 Skid steer decision maker single attribute utilities - epochs 1 and 2 42
4.3 Tractor decision maker single attribute utilities, epochs 3 and 4 43
4.4 Tractor, epoch 3 and 4, constraints 44
4.5 Skid steer attributes to design vector mapping 45
4.6 Tractor attributes to design vector mapping 46
4.7 LifeTrac design vector to component mapping 49
4.8 Basic metrics from LifeTrac tradespaces (designs considered $n = 592$) 51
4.9 Selected LifeTrac designs 64
4.10 LifeTrac transition rules 73
4.11 LifeTrac Rule-Effects Matrix 75
4.12 Example designs reachable through modularity path enablers 77
4.13 Product offerings for construction epochs (epochs 1 and 2) 78
4.14 Product offerings for agriculture epochs 79

B.1 Centrality and modularity metrics for LifeTrac 92

C.1 Degree modularity, $M(D)_i$ and $(Rank)$, of select LifeTrac components . . . 94
C.2 Distance modularity, $M(T)_i$ and $(Rank)$, of select LifeTrac components . . 95
C.3 Bridge modularity, $M(B)_i$ and $(Rank)$, of select LifeTrac components . . . 96

D.1 Retail Engine Prices 99
D.2 Skid Steer Specifications 100

Chapter 1

Introduction

1.1 Motivation

It is during the conceptual design phase of a system's project that much of the lifecycle cost is committed. One of the architecture choices which can be committed during conceptual design is how a system's components are interconnected and hence the level of modularity of the components. This modularity level can have long term impacts on the changeability of a system over its lifespan.

One of the difficulties with the previous statement is that modularity has many definitions, introducing ambiguity into how to measure and value modularity. MIT's Engineering Systems Division define modularity as "the degree to which the components of a system can be designed, made, operated, and changed independently of each other" (Allen et al., 2002); another definition is as a measure of the lack of technical interface connectivity between components of a system (Sosa et al., 2007), and yet another definition is modularity as a series of design rules placed upon a system (Baldwin and Clark, 2000). Further, modularity has been proposed to provide many lifecycle benefits to systems and products including economic value (Baldwin and Clark, 2000), aiding in system flexibility and evolvability (de Weck et al., 2011), increased product variety (Johnson and Broms, 2000), and

as an aid to complexity management (Browning and Eppinger, 2002). Some of the research on modularity assume that designers are able to find design rules that split the system into modules without degrading system performance (Baldwin and Clark, 2000), while others argue that modularity has been observed to have significant negative cost, weight, and performance implications in high-energy density systems (Whitney, 2003).

With so many definitions, potential lifecycle benefits, and potential trade-offs, design engineers are faced with many objectives and challenges when deciding how to modularize components within a system. Some work has been done on this front, including the development of clustering algorithms that take into account the endogenous system structure to determine component clusters (Yu et al., 2007) and the engineering system matrix that uses qualitative knowledge of endogenous and certain exogenous system factors as a screening for potential areas of high system change (Bartolomei, 2007). The author's experience is that the decision as to the organization of system components into modules during conceptual design currently seems to be based on expert experience, rules of thumb, heuristics, and iteration. Stakeholders and system designers need tools to aid in making design decisions regarding where to focus modularity efforts during conceptual and early design.

1.2 Value

The dynamic tradespace approach proposes tools for helping make design decision regarding incorporating changeability into a system, based on changes in customer utility over time and physics-based models of system performance (Ross, 2006). This research effort proposes that a dynamic tradespace framework can be coupled with a network representation of a system's components to aid in value-based decisions about how and if to modularize a system so as to make the system more robust to stakeholder requirement changes. This new approach will provide design engineers with the tools necessary to make decisions regarding modularization of a system in the support of product variety and evolvability.

The anticipated outcome from this research are tools that designers could use to make design decisions about what system components should be made into modules under specific situations.

1.3 Scope

The scope of this thesis includes the development of an approach to focus system modularization efforts and its application to a case. The Multi-Attribute Tradespace Exploration (MATE) method will be used, as it develops value focused designs and indicators of potential design changes. Design Structure Matrices (DSM) will be used to model the system architecture and the connectivity between the components and network centrality measures based on the design structure matrix will be used to measure modularity of the components. These two different techniques, MATE and DSM, will be combined into one approach.

The case study will be a particular mobile terrestrial machine; this particular machine was used to demonstrate the approach because it is one that has multiple use scenarios and is available as open source and is hence easily accessible to other researchers.

1.4 Methodology

The research approach will include a description of the current literature on modularity and methods and motivations for grouping system components into modules. Enhancements and/or additional inputs to the tradespace exploration process will be proposed, mostly in the form of connecting a network representation of a system as input into the tradespace exploration process. A motivating case study will be used to demonstrate the proposed approach connections/enhancements. The case study will include a new application of the tradespace exploration approach to a terrestrial vehicle.

Chapter 2

Literature Review

2.1 Overview

This section will step through the existing literature of the tools, frameworks, and concepts employed during the course of this research effort. The purpose of this section is twofold: place the research in the context of the existing published research and to provide enough background information that a reader can understand the remainder of the thesis, assuming they are familiar with the references provide. As such, it is not meant as a comprehensive study of these subject areas. The areas that will be covered include Multi-Attribute Tradespace Exploration (MATE), the design structure matrix (DSM), and an overview of structural modularity and modularization. With this background in place, at the end of this chapter the research questions and objectives will be presented.

2.2 Multi-Attribute Tradespace Exploration

2.2.1 MATE Overview

MATE is a decision analysis framework for making understanding the trade-offs between (sometimes) widely different design alternatives and to make those decisions as to which

alternatives to pursue in the face of uncertainty. Some of the first publications of the usage of MATE include Ross, Hastings, Warmkessel, and Diller (2004), with further development by Ross (2003). Later a method for considering multiple future scenarios was developed (Ross, 2006). The framework's key strengths are in its ability to map from different systems architectures and design variants to utility in the decision makers eye and to engage decision makers and system designers, through interactive exploration, in the decision making process. According to McManus et al. (2005), MATE's "intent is to allow informed upfront decisions and planning, so that the detailed design process which follows is aimed at the right solution, and is forewarned of potential problems and forearmed to seize potential opportunities."

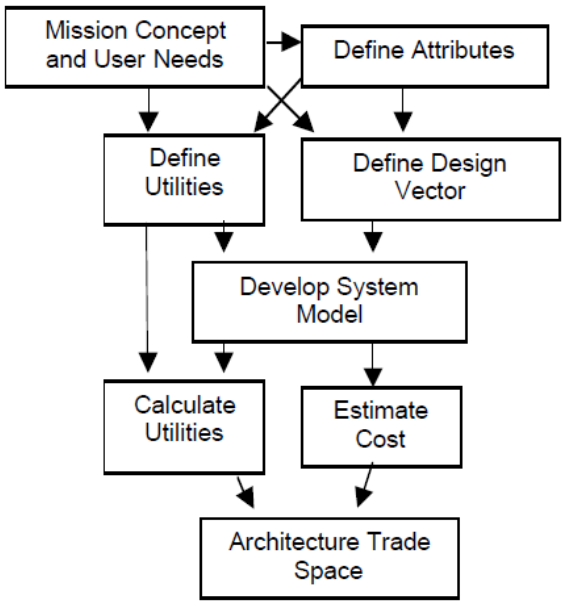


Figure 2-1: High-level flow of a MATE analysis, from (McManus et al., 2005)

The conceptual flow of the MATE process is shown in figure 2-1.Provided here is a brief description of each step of the flow.

Mission Concept and User Needs MATE starts with the synthesis of a concept of operations and a list of articulated user needs. This is the standard input into most

systems engineering activities.

Define Attributes Attributes are a measures of how decision makers value the outcome of the design process. In MATE, these are single-attribute utility attributes (Keeney and Raiffa, 1993). These are determine from the mission statement and user needs, as well as through other methods such as interviews with the decision makers.

Define Utilities Once the targeted set of attributes is elicited, utility functions are developed for each attribute. The first step is to work with the decision makers to determine the single attribute utilities and then how those are combined into a multi-attribute utility function (MAUF). This MAUF later used to determine the amount of utility a given design variant provides to the decision maker.

Define Design Vector This step of MATE uses as input the mission concept and user needs, attributes, and the utility functions to determine a set of design alternatives that might best satisfy the decision maker needs. The design alternative space under consideration is enumerated through a series of design vectors (DVs). One can think of a design vector as the dimensions of the system architectures that will be explored and the particular values of each dimension. The design vectors are often generated algorithmically because of the large number of design alternatives being considered during a MATE analysis.

Develop System Model System models are used in MATE to map from the DV space to a system performance space. The system performance space dimensions are the attributes based, gathered earlier in the process. The system model for MATE is usually based on the physics of the underlying system under consideration, and can either be theoretical based or determined through empirical models or a combination of both. The system model needs trade model fidelity with model performance; because a MATE analysis might consider hundreds or thousands of different design

alternatives, this trade in model fidelity and accuracy and computation speed can be important to the usability of a MATE analysis.

Estimate Cost Coupled with building the system performance model, a cost model of the system is determined. The cost model is built such that decision makers can make trade off decisions between system cost and performance as measured in utility. The cost model can take many forms and should be appropriate to the system architectures under consideration.

Calculate Utilities This step maps the system performance attribute values to the decision makers multi-attribute utility (MAU). Utility and estimated cost make up the two dimensions of the trade space.

Architecture Trade Space The final step of a MATE analysis is for the decision makers and system designers to explore the space of utility-cost trades. The purpose of this exercise is to make a decision on which, if any, designs to move forward into the next stage of systems development. Alternative outcomes could be to revisit the system architecture alternatives because too few or none of the designs are considered promising, confirm with the decision makers the absoluteness of their utility requirements, or abandon or shelve the project. Many visualization, analysis, and interaction tools have been brought to bear on this step of a MATE analysis.

Much of the MATE case studies has been used for decision analysis (Ross, 2003, 2006). Some cases include the use of the output from MATE as an input into the design process (Shah, 2004). In this research, as specified in later chapters, MATE will be used in both capacities.

2.2.2 RSC as a process formalization

Later versions of MATE incorporate considerations for uncertain futures, under the name dynamic-MATE (Ross, 2006). Dynamic MATE was further codified and formalized as

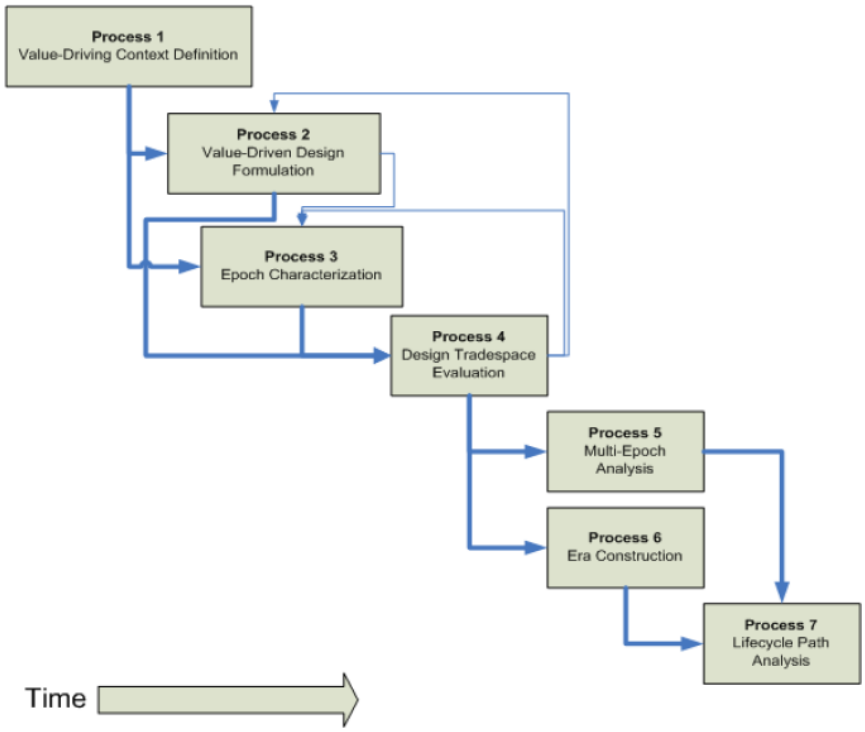


Figure 2-2: High-level flow of a RSC analysis, from (Ross et al., 2009a)

Responsive Systems Comparison (RSC) method (Ross et al., 2009a). RSC supports mechanisms to construct future possible system context situations into epochs, construct the epochs into multiple epochs to consider full system lifecycles, analysis tools to select potentially valuable designs, and mechanisms to consider transitions from one design vector to another and the impact that has on value delivery, A flow chart view of RSC is shown in figure 2-2. This work will utilize several constructs from RSC, and in particular will make a connection between modularity as discussed in this work and the tradespace network analysis.

2.2.3 Role in enabling evolutionary acquisition

Derleth (2003) applies MATE to evolution acquisition by presenting a case of small diameter bombs. In this work, three spirals through the acquisition process are considered, both with changing decision maker utility functions as well as changing design vectors and design dimensions. Over the three evolutionary acquisition spirals different architectures emerge as being potentially viable and design decisions made in earlier spirals may limit (or make not cost effective) transitions to later designs. Derleth claims that "...tracking different architectures through different spirals can help the systems engineer to understand why architectures should or should not be chosen, where flexibility is important, and what the system and utility drivers are." This notion of focused flexibility will be used by this research to suggest areas to focus the modularization efforts of system architectures.

2.3 Design Structure Matrix

2.3.1 DSM Overview

The design structure matrix (DSM) is a system modeling tool used to highlight the interactions between elements (Browning, 2011). DSM's use a square matrix to represent system elements along the each of the axis, and interactions between the system elements

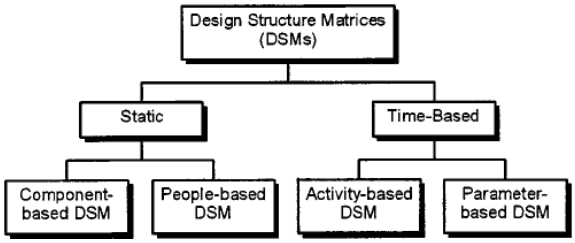


Figure 2-3: Taxonomy of DSM types, from (Browning, 2001)

as marks in the matrix (Browning, 2001). Several types of DSM's have been proposed, with a basic split being made between DSM's representing statics and dynamics of the system under consideration (Browning, 2001). In the static DSM branch, these are further broken down into the two major application areas of products and organizations. This taxonomy is shown in figure 2-3. DSM's are but one system analysis tool, and this this research the application of static DSM's will be used because they lend themselves to modularity and design option analysis.

2.3.2 DMM and MDM Overview

While a DSM provide a concise mechanism to show the interaction of system elements in one domain, the domain mapping matrix (DMM) (Danilovic and Browning, 2007) and the multiple domain matrix (MDM) (Lindemann et al., 2009) provide mechanisms to map between domains. MDM's build upon the DMM, adding allowed mathematically operations between single domain DSMs and multiple domain DMMs. A classic example of the multiple domain matrices include several matrices from the house of quality (Hauser and Clausing, 1988; Lindemann et al., 2009). Additional examples of DMM's existing in MATE, such as mapping between the DV and the decision makers value generating attributes (Ross, 2003). In this research, DMM's will provide formalized mapping between the system component architecture domain and the design vector domain.

2.4 Modularity and Modularization

2.4.1 Definitions of Modules, Modularity, and Modularization

One of the more difficult aspects of this research was to settle on an operational definition of modules, modularity, and modularization. MIT's Engineering System Division defines a module as "a part of a system that is constructed to have minimal, standardized interactions with the rest of the system" and modularity as "the degree to which the components of a system can be designed, made, operated, and changed independently of each other" (Allen et al., 2002). Several authors have defined modularity as something along the lines of the one-to-one mapping of a system function to the system form and including certain component interface characteristics ((Ulrich, 1995), for example), while others have defined modularity as a series of design rules that enable various operations, called modularity operators, on the system (Baldwin and Clark, 2000). While these definitions suffice as conceptual definitions, there is a need in this research to find a definition which is lends to an operational measure of modularity.

This research is primarily concerned with increasing the changeability of a system based on determining how well certain components are connected together as modules and their connectivity to the other elements in the system. With this desire, and the need for an operational measure, the definition provided by Sosa et al. (2007) will be adopted by this research: modularity is "a measure of the lack of technical interface connectivity between components of a system". Based on network analysis and graph theory, Sosa et al. develop three metrics for measuring component modularity and their impacts on component redesign. The three metrics, degree modularity $M(D)$, distance modularity $M(T)$, and bridge modularity $M(B)$ are a normalizing of existing network centrality metrics (Degree, Freeman Closeness, and Freeman Betweenness) to a range of $[0, 1]$ where 1 is the highest level of modularity for each metric. From these metrics the authors find a correlation between the set of outdegree modularity and outdistance modularity and the likelihood of

planned redesign of a component to change the systems performance level. These metrics and this correlation finding will be used for the modularity analysis of components needed to change a system's performance.

2.4.2 Proposed benefits of and challenges to modularity

Modular systems have been proposed to provide many lifecycle benefits to systems and products including but not limited to increasing overall system economic value (Baldwin and Clark, 2000), aiding in system flexibility and evolvability (de Weck et al., 2011), increasing product variety (Johnson and Broms, 2000), and as an aid to complexity management (Browning and Eppinger, 2002). McManus and Hastings (2006) propose that modularity as a mitigation strategy for certain types of uncertainties. In this regard, modularity can be viewed as a means to achieve desired goals, and may be critical for achieving other lifecycle properties (Ross, 2008; de Weck et al., 2011).

While modular architectures have many potential benefits, these benefits do not come without costs. Whitney (2003) claims that for systems with relatively high energy density, modularity increases weight, increases cost, and/or results lower performance. Hölttä-Otto and de Weck (2007) demonstrate that for systems with business and performance constraints that in the case where stakeholders value lower weight, smaller size, or higher performance one tends to find more integral architectures; conversely when stakeholders value commonality and reuse across products, in order to achieve cost savings, one tends to find more modular architectures.

2.4.3 Existing modularization approaches

The primary methods to aid a design engineer seemed to be focused on clustering or heuristics. Yu, Yassmine and Goldberg develop an clustering approach, using DSM representations of a system, to cluster a proposed system into blocks where the interactions within a module cluster are maximized and between clusters are minimized (Yu et al.,

2007). The gaps with this approach is that it only looks endogenously at the system under consideration and lacks tools necessary to identify areas where modularization may aid in achieving desired lifecycle properties and system variability across decision makers or through time. Function-cluster was proposed as a way to consider potential changes in mass, energy, and information flow between components and to cluster a system into modules and suggested system cleavage points to introduce interfaces between modules that will minimize the likelihood of propagation of changes should a module need to change (Hölttä-Otto, 2005). This approach by Hölttä-Otto takes into consideration exogenous factors that could require changes in the system, but leaves it up to the system architect to recognize potential changes that may emerge due to changing or different stakeholder needs.

On the other end of the modularization approach spectrum are heuristics, from sources such as Maier and Rechtin (2009), Stone et al. (2000), and classes such as ESD.34 at MIT (Crawley, 2007). The gap with these approaches is that they suggest extensive experience is required by the system architect in order to make modularity choices, leaving the developing organization to rely on tacit knowledge and the designer with little experience to learn through trial and error.

2.4.4 Prior MATE studies

It is important to note that at least two prior MATE studies have incorporated modularization and modularity in their analysis. The first was a study of Joint Direct Attack Munition (JDAM) done after the JDAM had been deployed (Ross, 2006). In this study modularity was considered as a path enabler to transition between designs. Alone modularity contributed to decreasing transition time for some of the transition rules and combined with use of commercial off the shelf (COTS) components contribute to a decrease in transition time and costs. In this regard, modularity was considered as a tool to enable the lifecycle property of changeability for the JDAM. The second study used MATE to inform

the development of modularity design rules for a satellite cluster that would enable evolutionary acquisition (Shah, 2004). This study treats the satellite as a module and considers the interactions between the modules during operations; this modularity has recently been referred to as "modularity 2" (de Weck et al., 2011) and is not the scope of this research.

2.5 Research Objectives / Questions

With so many proposed benefits and trade-offs of modularity, it seems important to focus design efforts on the components of the system most likely to benefit from being modular. Modularity for the sake of modularity could lead to unnecessary product development costs and schedule and/or systems not meeting stakeholders expectations in cost and performance. On the other hand, modularity may be a key enabler for achieving lifecycle system properties, such as evolvability and flexibility, and the lack of appropriate modularity may inhibit those properties. Systems and design engineers need better tools to focus modularization efforts on the key components of the system that may benefit from modularity, as well as a way to measure the extent to which a particular component arrangement within a system architecture is modular. MATE provides the tools to determine which design variables may need to change over time or between stakeholders, while DSM and modularity metrics give the designer tools to measure the degree of modularity of a component.

With that in mind, the guiding questions for this research are:

1. How could the MATE framework be used to inform modularization decisions in support of changing and/or differing stakeholder requirements?

2. What is the relationship between component modularity, system path enablers, and transition rules within the dynamic MATE framework?

3. How can the findings from above be used to inform existing approaches to modularization?

This research plans to join MATE and DSM modularity metrics into an unified approach that system architects and engineers can use to make informed decisions about where to focus modularization efforts. Anticipated outcomes from this research include tools that designers could use to make design decisions about what system components should be made into modules under specific situations.

The remainder of this thesis is the proposed design process, the use of the proposed design process on an existing terrestrial off-road vehicle, and a discussion of the results and conclusions from the research.

Chapter 3

Design Approach Proposal

The purpose of this chapter is to outline a set of goals for an approach to design that will enable value-centric decisions about component modularity within a system, propose an approach that will achieve those goals, and show the connection of the approach to system development activities downstream from the proposed approach.

3.1 Approach Goals

In this section an approach is proposed that will aid designers in making value-centric decisions on component modularity. The goals of the approach are as follows:

1. Provide a mechanism to understand potentially desired changes to design variables, based on decision maker preferences, and link those changes to components that may need to be altered in response. The purpose of this goal is to provide designers a value-centric approach to target modularization efforts.

2. Build upon the Responsive Systems Comparison (RSC) method by making connections to RSC's existing process steps. The purpose of this goal is to use the decision analysis tools from RSC to make decisions as to the trade-offs in terms of decision maker utility, system cost, and modularization.

3. Utilize the component modularity metrics by Sosa et al. (2007) to evaluate the component modularity of a proposed system architecture. The purpose of this is to be able to measure component modularity during early stage architecture synthesis or to be able to reverse engineer existing systems and quantify the modularity of components in that existing system.

3.2 Approach Roadmap

With these goals in mind, the approach in figure 3-1 is proposed to achieve these goals. The approach additions to RSC process, with black and white portions representing RSC Ross et al. (2009a), and the additions that aid in making value-centric decisions regarding component modularity shown in red.

3.2.1 Component Modularity Analysis

Component modularity analysis consists of two activities: quantification of the modularity of the components of a system and linkage of the components to the design variables used in RSC.

Quantification of the components is done by representing the system in a way that is susceptiable to network centrality measures, such as a DSM or network representation, and then calculating the component modularity metrics. This is the same method as used in Sosa et al. (2007).

It should be noted that constructing the form-form interaction network requires either development of a DSM or network representation of the form-form interactions of the proposed system. Because it is often the case that the proposed new system is an evolution of an existing system, for many systems development efforts this assumption is deemed achievable. There are other scenarios, such as green-field conceptual level design, where representation of the form-form interactions of the underlying system cannot be built due to lack of necessary inputs. This thesis did not attempt to address these latter types of

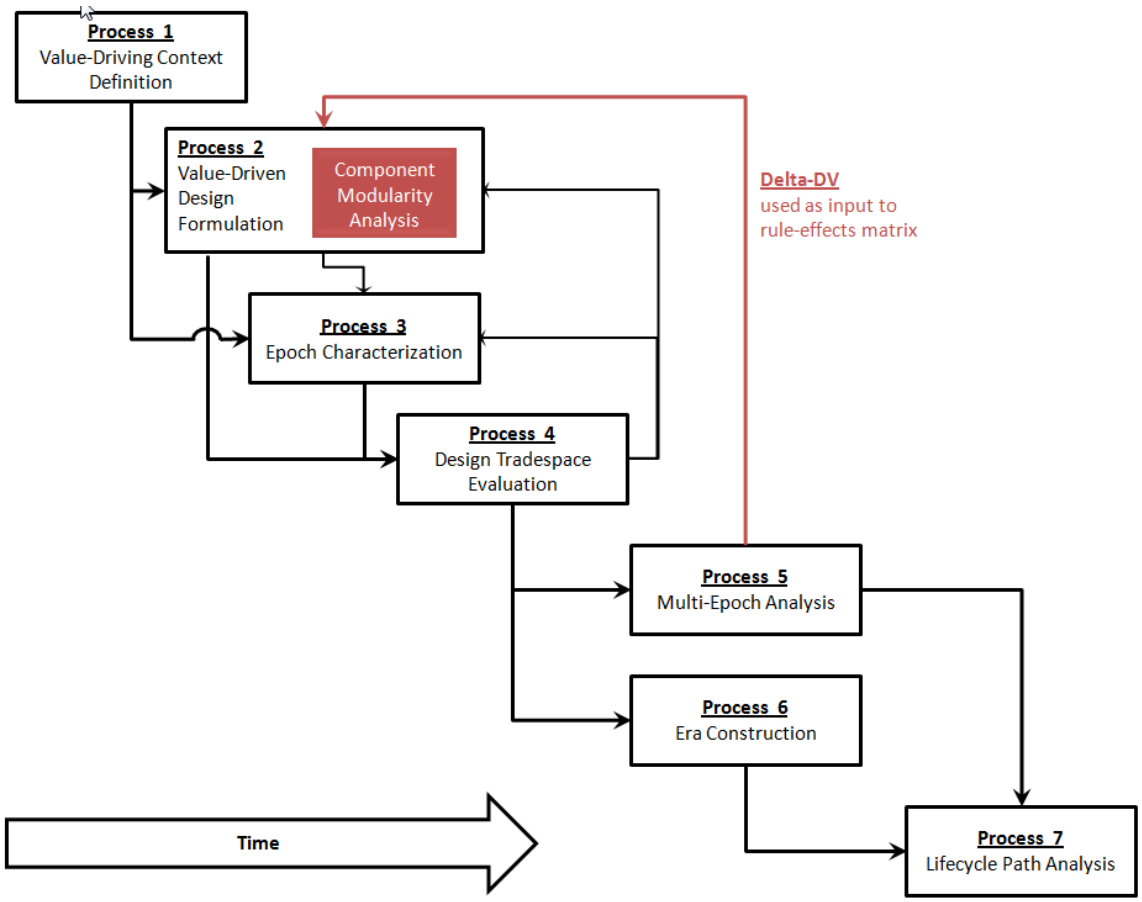


Figure 3-1: Proposed additions to RSC for value-centric component modularity design

scenarios and it could be an area of future research; a potential approach is suggested in section 4.9 and section 3.2.4.

The second activity is to provide a linkage between components of the system and design variables. This linkage is an indication of the system components that are likely to need to change as a result of a change in level of a design variable. This linkage is represented as a multiple domain mapping table between the components and the design variable.

If an existing system is in place, the output from these prior activities can be used to build an initial rule-effects matrix that will include the appropriate modularity path-enablers. For example, let one assume that a set of components is likely to need to change because of a change in a design variable and that same set of components have relatively high modularity metrics values. In this example one can reasonably introduce into the rule-effects matrix a modularity path enabler and the appropriate impacts to system cost and performance into the resultant tradespace network analysis. These impacts may be represented as a path enabler which lowers the time and/or cost to transition from one design to another.

3.2.2 Delta-Design Vector Development

The next few paragraphs discuss the feedback loop from Process 5, Multi-Epoch Analysis, to Process 2, Value Driven Design Formulation (see figure 3-1).

The delta design vector (delta-DV) is a construct where potentially valuable designs are selected and evaluated for variability in design variables between potentially valuable designs. The variability is analyzed because it can be an indicator that multiple designs, with different levels of that design variable, could be of value to decision makers. The inputs into this analysis determines if the analysis represents value to dynamically to a single decison maker or statically across multiple decision makers: if the epochs are constructed to represented dynamically varying attributes to a single decision maker then

the former type of analysis would result whereas if the epochs are constructed to represent the decision attributes of multiple decision makers then the latter type of analysis would result. If a designer chooses to support changes to a particular design variable over the life of a system, then the designer can take appropriate action in terms of modularity requirements on the components that may need to be altered in order to reduce the time and/or costs to execute the change in designs.

The output from the delta-DV is used as input into the rule-effects matrix in the tradespace network analysis portion of RSC to synthesize modularity path-enablers for consideration (an example of this will be given during the case study, in section 4.8.2). Because this may introduce modularity path enablers that were not previously included and because the path enablers may impact system performance and cost, this may require re-executing Process 2 and Process 4, Tradespace Evaluation, in order to understand the impacts of the modularity path-enabler.

The delta-DV is tied to the component modularity analysis via a table that links the design variables to the components that are likely to change as a result of a design variable changing. This linkage would only be in place if there is a system architecture being proposed at this stage, as may be the case in incremental changes to existing products or systems. Because of this, the modularity requirements should also be carried into downstream architecture synthesis activities as covered in the next section.

3.2.3 Approach Flow

In an attempt to better describe the approach activities, they have been diagramatically represented in figure 3-2. It should be noted that this is not a formal process, per se, and one could perform other steps as one finds appropriate. This diagram will be used throughout the case example in chapter 4 as a road-map to help the reader understand which step of the process is being executed.

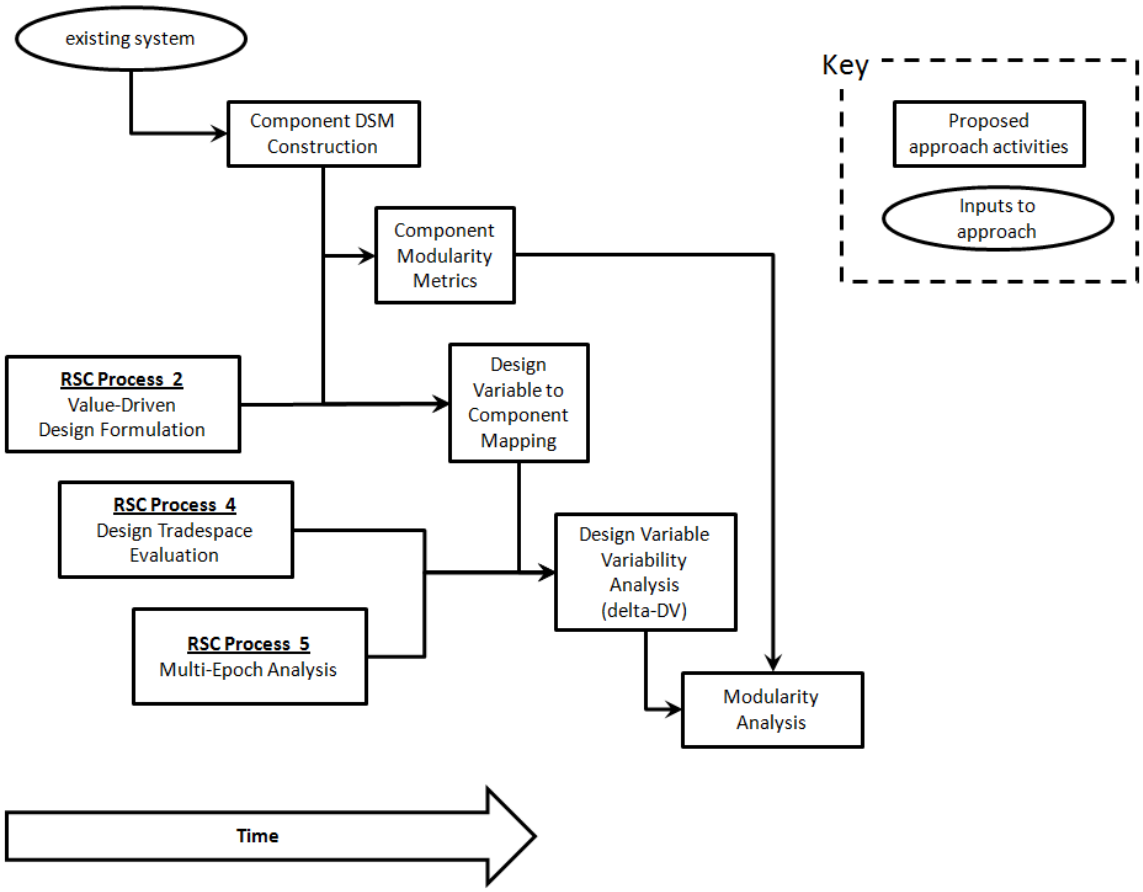


Figure 3-2: Proposed approach steps to value centric approach to modularization

3.2.4 Connection to Architecture Synthesis

An output from the proposed approach could be a required level of modularity in support of changes to certain design variables. The introduction of the component modularity evaluation in the downstream architecture synthesis activity allows quantitative evaluation to determine if those goals have been achieved based on a certain design. The system architecture would be represented as form-form interactions of the system components. From that the component modularity metrics will be executed against the proposed design and compared to the modularity requirements flowing out of the RSC process to determine if the modularity requirements have been achieved. This is shown graphically in figure 3-3.

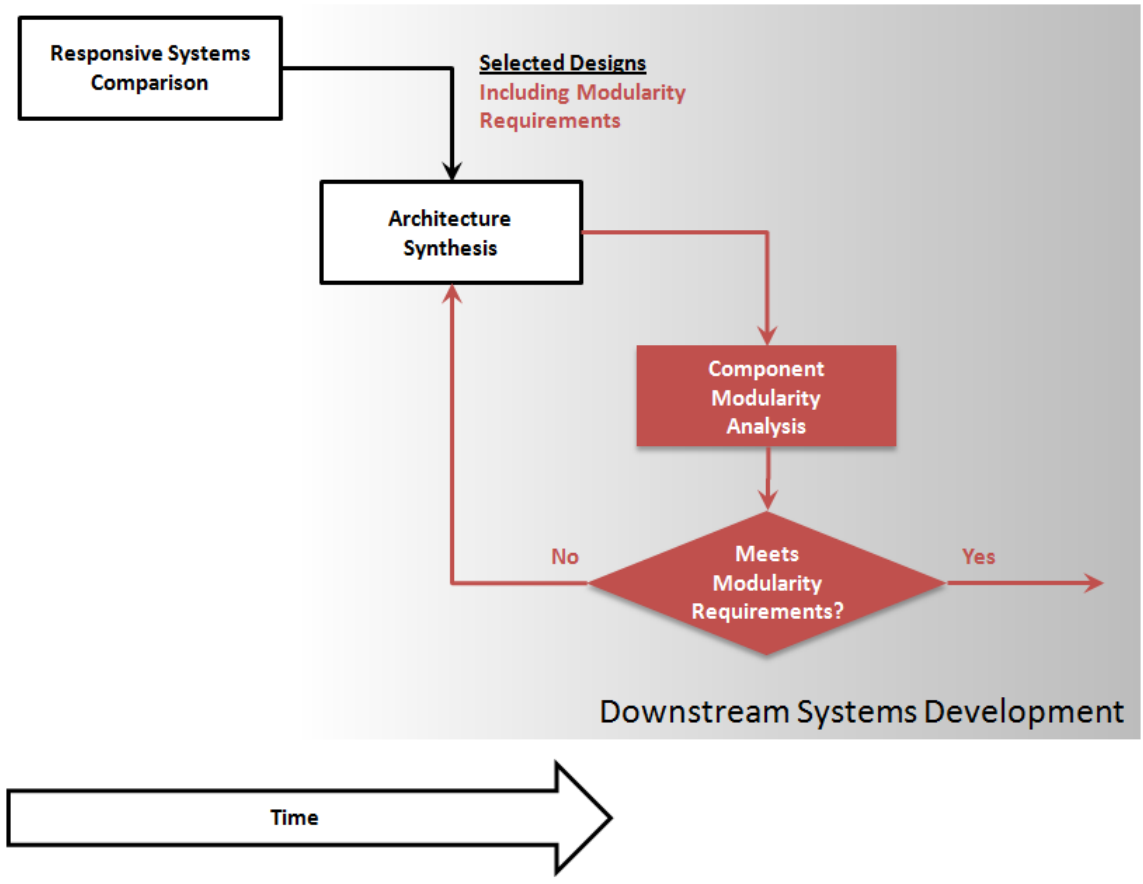


Figure 3-3: Proposed connection to architecture synthesis

3.3 Summary

This chapter has covered the goals of the proposed design approach and made a proposal to achieve those goals. The scenarios starting with an existing design has been covered. Also, a connection to activities downstream from RSC has been proposed. In the next chapter a case study will utilize this design approach.

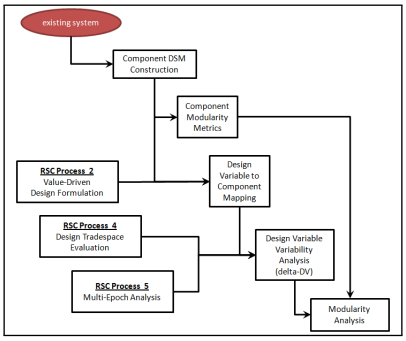
Chapter 4

Case Example

4.1 Overview

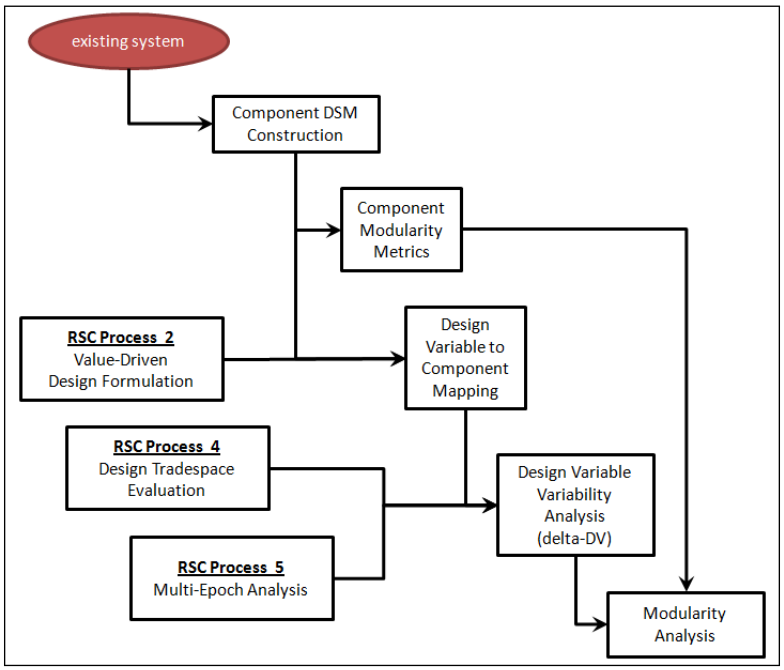
This section of the research applies the design approach from the previous section to a case study of an open-sourced, terrestrial vehicle that is intended to act as an agricultural tractor or a wheeled skid-steer.

Throughout this chapter, as each activity of the proposed process is executed, figure 3-3 will be shown so as to provide sign posts for the reader. The approach activity being executed in a given section will be highlighted in red in the diagram. An example is shown below.



Example process activity roadmap

4.2 Target System - LifeTrac



The target system is the OpenEcology Project's LifeTrac tool (Open-Ecology-Project, 2011a) (see figure 4-1). The LifeTrac tool is designed for two functions: acting as a simple wheeled skid steer and as a simple agricultural tractor. The purpose of the skid steer function is to push, lift, and move material around a work site. The purpose of the agricultural tractor function is to provide tractive energy for pulling agricultural implements, such as tillage or seeding equipment, through a field. It is interesting to note that the LifeTrac tool is described as a "modular" design, but little is given on the website to backup that claim.

The LifeTrac tool was chosen for this case study for the following reasons:

- The LifeTrac tool is targeted for use by two types of users. The first type is the farmer that would like a tool to aid in field operations; the second type is a construction work site operator that needs a tool for moving material around a job location. With

Figure 4-1: Photograph of LifeTrac, from (Open-Ecology-Project, 2011d)

these two types of users and use cases, one might expect two different measures of utility that then could result in different desirable system properties.[1]

- The LifeTrac tool is an existing system, demonstrating the use of the proposed modularization design approach in the evolution of a system. This is considered an acceptable starting scenario as many design efforts are incremental in nature.

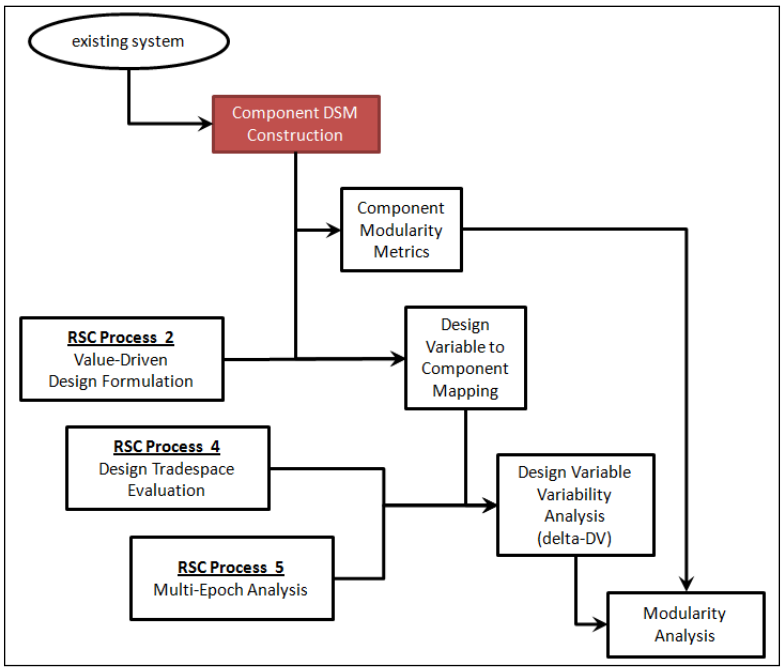
- While a simple design, it is sufficiently complex to demonstrate the proposed modularization design approach. Sufficiently complex was determined by the number of elements ($n = 47$) and interactions ($i = 218$) between the components based on the analysis of the component design structure matrix model of the system. By keeping with a simple design it is hoped that the research is more approachable.

[1]For the remainder of this case study, the term 'decision maker' is used to denote the person responsible to deciding what equipment will be used; in marketing terms they might be considered the 'purchase decision maker'. Where the term 'user' is utilized, it denotes the person operating the equipment. These may or may not be the same person.

- The designs and costing information is covered under an open source license, allowing ease of research both now and in the future (Open-Ecology-Project, 2011e).

4.3 Structural Model

4.3.1 Component DSM Model



The first activity in the proposed approach is to construct a component DSM model of the underlying system. This was done for the LifeTrac tool by reverse engineering open source computer aided design (CAD) models of LifeTrac (Open-Ecology-Project, 2011c). The component DSM model in figure 4-2 considers four types of dependencies: spatial (P), mass flow (M), information flow (I), and energy flow (E). Overall, there are 47 DSM elements in the LifeTrac DSM, with 218 element-to-element dependencies, leading to a

interaction density of 0.10.[2] A version of this DSM, zoomed-in to be more readable, is available in appendix A.

These interaction types were simplified to binary, single type (there is either an interaction between two elements or there is no interaction) interactions for the remainder of the case study. The decision to do use binary, single type interactions is a trade off in model fidelity versus execution time for later analysis. Further, (Sosa et al., 2007) suggests that physical and information dependency may be more easily determined than energy or mass flow dependencies; by collapsing to one type it is suggested that at least the connectivity between components will be more completely recorded for use in the subsequent network centrality analysis. The lowest fidelity form of a DSM shows interactions as one type and binary, whereas many alternatives are available for increasing the fidelity of the model of interactions, including types of interactions as well as strength of interactions (as done by (Sosa et al., 2007)).

At this point in analysis it is not uncommon to apply clustering algorithms to the DSM to group components together into modules. However, that step is not necessary here as we are not interested in the grouping of components into modules because clustering does not change the connectivity between the components. Instead, we want to explore how connected each component is to all other components in the system as an indication of the "cost" of propagation of change. As such, we will skip the effort associated with DSM clustering and move onto the next step.

[2]Calculated as $D = I/(E^2 - E)$ where D is the interaction density, I is the count of the number of off-diagonal elements in the DSM where there is an element-to-element interaction, and E is the number of elements in the DSM.

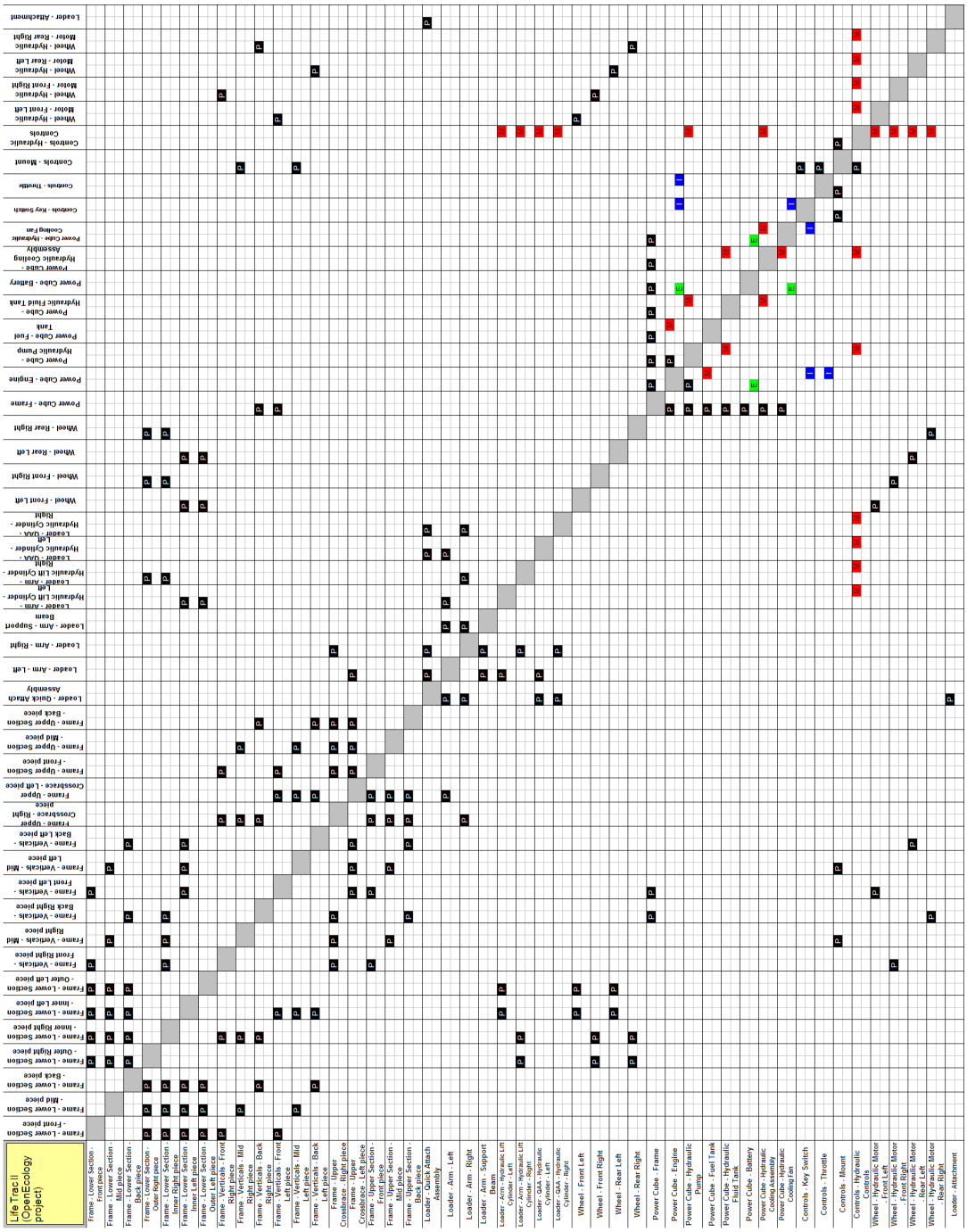
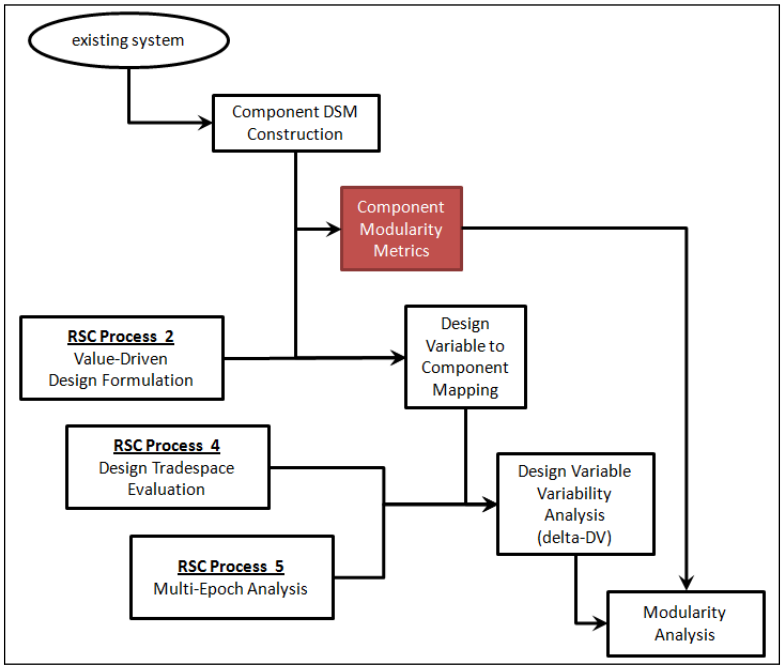


Figure 4-2: Component DSM of LifeTrac

4.3.2 Component Modularity Metrics



The next activity in the proposed design approach is to calculate the metrics for degree, distance, and bridge centrality as given in (Sosa et al., 2007). Each of these metrics is in the range of $[0, 1]$, with higher values corresponding to a higher level of modularity for that metric. Just as a review, these metrics provide the following insights into the connectivity of a component to other components:

Degree modularity tells one about the number of other components that have direct design dependencies with a given component. The less direct design dependencies component i has with other components, the higher the value of $M(D)_i$. This is the simplest of the three metrics.

Distance modularity tells one about how far away (or how close) a given component is to other components. This is built on the concept of farness (or its inverse, closeness) in network theory. This measure captures the concept that design changes may propagate not just to/from immediate neighbors (as measured in degree modularity) but

also through the network of the design dependencies from components. If component i has high distance modularity $M(T)_i$ then changes to that component would have a longer distance to traverse to reach other components in the system.

Bridge modularity tells one about how many design dependency paths a component lives on between other components. This is built on the network theory concept of centrality. The idea with this measure is to capture the degree to which a component is on design dependency paths between other components. If component i has high bridge modularity $M(B)_i$ then it lies on fewer design dependency paths between all other components.

These metrics have been calculated for all LifeTrac components: for those so inclined, the full enumeration is available in appendix B. These metrics, owing to the symmetry and binary interactions of the DSM, are modified for this case analysis[3] and defined as

$$M(D)_i = 1 - \frac{D_i}{n-1}$$

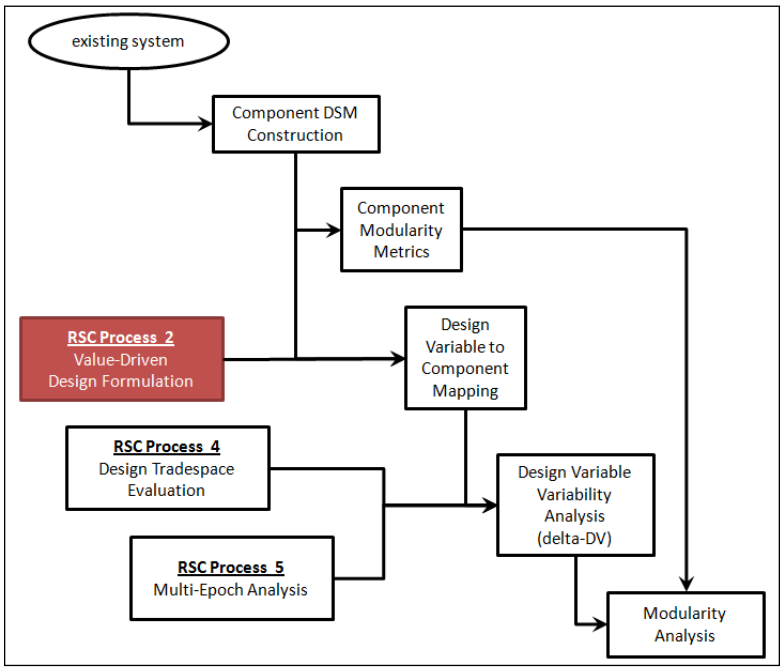
and

$$M(T)_i = \frac{T_i}{n(n-1)}$$

where $M(D)_i$ is degree modularity for component i, D_i is the Freeman degree for the component i, T_i is the Freeman farness for component i, and n is the number of components. Owing to the symmetry of the LifeTrac DSM, if in-degree and out-degree modularity, as specified by (Sosa et al., 2007) were each calculated for this DSM the values would be the same for each component (that is, $M(OD)_i = M(ID)_i = M(D)_i$); the same is true for out-distance and in-distance modularity. Bridge modularity, $M(B)_i$, maintains the same definition as given by (Sosa et al., 2007).

[3]Sosa et al. (2007) provides equations for the modularity metrics that account for a DSM where the strength of the interaction between the components can be specified and where the interactions are asymmetrical.

4.4 Value Driven Design Formulation



This section covers the activities which take place during RSC Process 2. RSC Process 2 includes identifying decision makers, eliciting decision maker attributes, building multi-attribute utility functions based on the elicited attributes, developing a cost model, and developing a set of design variables that map to the decision maker's attributes.

4.4.1 Decision Makers, Utility Functions, and Costs

For this case, two decision makers and concepts of operations will be considered for the LifeTrac: one is a consumer that will use the LifeTrac as a skid steer to move materials around a work site, another is a farmer that will use the LifeTrac for agricultural field work (in particular, work associated with preparing a seed bed for the soil as in a tillage operation and/or seed planting operation). In practice this can be represented as three use scenarios, one for construction and two for agricultural usage, as given in table 4.1.

Table 4.1: Context Variables

Category	Context Variable Name	Unit	Range
Construction usage	Solution requirement	type	1 scenario
Agricultural usage	Farm tractor, row spacing	inches	(24, 30)

In order to develop the utility functions for the skid steer operations, a skid steer purchaser and user was interviewed (Engler, 2011). It should be noted that the interviewee is also an MIT Systems Design and Management student and had awareness of MATE and the need to build utility functions; this made the interview go very quickly. The interview did not attempt to completely elicit complete single attribute utilities and build a multi-attribute function, but instead was used to determine the critical attributes, some notion of the range of acceptable attribute values, and if the function was smaller is better (SIB) or larger is better (LIB). Other methods for determining attributes and utilities would be used in a more rigorous project, but is justified in this case because the focus of this research is connecting a MATE analysis with modularity analysis and not the accuracy of the utility functions. To test sensitivity of the decision maker's preference in this first scenario, two preference sets will be considered: one that will be represented by Epoch 1 and one that will be represented by Epoch 2.

The results of the interview are shown in table 4.2 and the function of each attributes utility is given in figure 4-3. A brief description of each attribute is called for here:

Material capacity - maximum load is a measure of the maximum weight that the vehicle has the capability to vertically lift and carry around a work site,

Maneuverability - vehicle width is a measure of the ability of the vehicle to fit through openings and passages, and

Lifting capacity - breakout force is a measure of the force that can be applied to break material apart, as when pulling an embedded stone out of the ground.

Table 4.2: Skid steer decision maker single attribute utilities - epochs 1 and 2

Attribute	Measurement	Units	$U=0$	$U=1$	Epoch 1 k_i[a]	Epoch 2 k_i[b]
Material capacity	Maximum load	pounds	1200	2000	0.0	0.5
Manuverability	Vehicle width	inches	96	72	0.0	0.2
Lifting capacity	Breakout force	pounds	1500	2500	0.0	0.3

[a] Because $\sum_i k_i = 0$ the MAUF will simplify to the multiplicative form $U(\underline{X}) = \prod_i U^i(X^i)$, indicating a preference set where the decision maker will not accept partial solutions.
[b] Because $\sum_i k_i = 1$ the MAUF will simplify to the additive form $U(\underline{X}) = \sum_i k_i U^i(X^i)$, indicating a preference set where the decision maker will accept partial solutions.

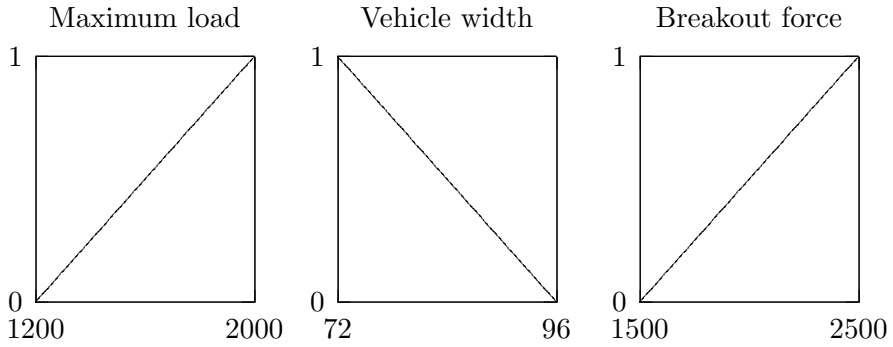


Figure 4-3: Single attribute utilities $U^i(X^i)$ for skid steer epoch

These are then combined into a multi-attribute utility function using the formalization by :

$$KU(\underline{X}) + 1 = \prod_{i=1}^{M} \left[Kk_i U^i(X^i) + 1 \right], \text{ where } K + 1 = \prod_{i=1}^{M} \left[Kk_i + 1 \right]$$

where $U(\underline{X})$ is the multi-attribute utility function, K is a constant of normalization, and k_i is the weight for the single attribute utility function $U^i(X^i)$ Keeney and Raiffa (1993) as provided in (Ross, 2006).

As stated earlier, two preference sets as two epochs will be considered: epoch 1 represents a decision maker that will not be satisfied with partial solutions and epoch 2 will represent a decision maker that is satisfied by partial solutions. This is done because

a full MAUF was not solicited and becomes a sensitivity test to the assumptions. In the epoch 1 case, because $\sum_i k_i = 0$, the MAUF simplifies to the multiplicative function $U(\underline{X}) = \prod_i U^i(X^i)$. In the epoch 2 case, because $\sum_i k_i = 1$, the function simplifies to the linearly additive form $U(\underline{X}) = \sum_i k_i U^i(X^i)$.

To determine the the utility function for the tractor operation and epochs, a review of existing literature was used with a primary source being Goering and Hansen (2004). From this the primary function measure of tractor performance was determined to be the rate at which the machine could complete one of the more demanding tasks typical in a farming operation: moldboard plowing. A photo of a small tractor performing this operation is shown in figure 4-4.

Figure 4-4: Moldboard plowing, from http://tinyfarmblog.com/breaking-new-ground/

Table 4.3: Tractor decision maker single attribute utilities, epochs 3 and 4

Attribute	Measurement	Units	$U = 0$	$U = 1$	k
Efficiency	Work rate	acres/hour	2	6	1

With this information a single attribute utility function for two epochs is considered and given in table 4.3 and the single attribute is graphically shown in figure 4-5. Because there is only a single attribute, there is no need to invoke the multi-attribute function formalization and this epoch's utility function is the single attribute function.

As stated earlier, two different epochs are considered for the agricultural scenario: one that utilizes 30 inch row spacings and one that utilizes 24 inch row spacings. This is done

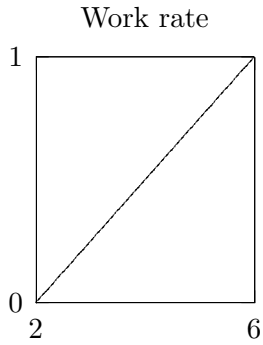


Figure 4-5: Single attribute utility $U^i(X^i)$ for tractor epochs

because row crops in the United States are typically planted on 30 inch or 24 inch row spacings in order to maximize plant productivity; it is represented as two epochs because of the cost of switching all equipment from one configuration to another means a farmer only rarely make changes between the two field configuration. Between these two epochs, the utility measures are the same, but the performance of the vehicle in the epochs is different as the field implements which are used for each field spacing configuration may result in different system performance. The constraints between the two epochs is given in table 4.4.

Table 4.4: Tractor, epoch 3 and 4, constraints

Epoch	Epoch variable name	Units	Constraint
3	Farm tractor, row spacing	inches	30
4	Farm tractor, row spacing	inches	24

For both agricultural and construction usage, lifecycle cost is measured as the acquisition cost of the underlying system plus the fuel usage over the period of use. The period of use is considered to be 5 years with 300 hours of engine time per year, typical for usage patterns and the lifespan of products of this type and size. It is recognized that this is a simple costing model, but it is sufficient for this case study. The acquisition cost model

for LifeTrac was built up from information at (Open-Ecology-Project, 2011b) and (Small-EngineSuppliers.com, 2011). For operating expenses, costs were assumed to be USD1 per horsepower-hour.

4.4.2 Design Variables Mappings

In this section, the mapping of decision makers attribute's to possible design variables is considered. This is done in a simplified version as compared to (Ross, 2003), but given the simplicity of the system and the follow-up tradespace analysis it is felt there is no loss of the model's fidelity. The goal of this mapping is to help ensure that all design variables related to achieving stakeholders goals are considered in subsequent tradespace development.

Table 4.5: Skid steer attributes to design vector mapping

Design Vector		Attributes		
Variable	Range	Material capacity	Maneuverability	Lifting capacity
Bucket width	[56-84] inches	X	X	
Available hydraulic power	[4-40] HP	X		X

The mapping for the skid steer epoch is presented in table 4.5. For each element, ranges were largely determine by the availability of off the shelf components; in a clean sheet design these dependencies would be relaxed. A description and justification for each design vector element is called for:

Bucket width represents the width of the attachment on the front of the machine. Width of the attachment bucket minimum was set in a range typical available from commercial suppliers (this was based on a review of attachment bucket sizes from Deere, Bobcat, and Caterpillar websites). For the sake of simplicity, only bucket width is

determine to affect maneuverability, because as width of the vehicle is fixed in this epoch and is considered not tradeable.

Available hydraulic power represents the amount of hydraulic power, as measured by the brake power of the engine. The base design of the LifeTrac has an engine power of 28 horsepower (HP). The minimum and maximum power of this design variable is set at 4 HP and 40 HP, as it is the limit of available air-cooled internal combustion engines available off the shelf (based on a review of engines from (SmallEngineSuppliers.com, 2011) on 20-November-2011, the same source the LifeTrac team sourced the current engine). In a more pure tradespace analysis this would not be limited by commercial off the shelf considerations; in this case we are keeping with design goals of LifeTrac by limiting to relatively simple, available, low cost engines.

Table 4.6: Tractor attributes to design vector mapping

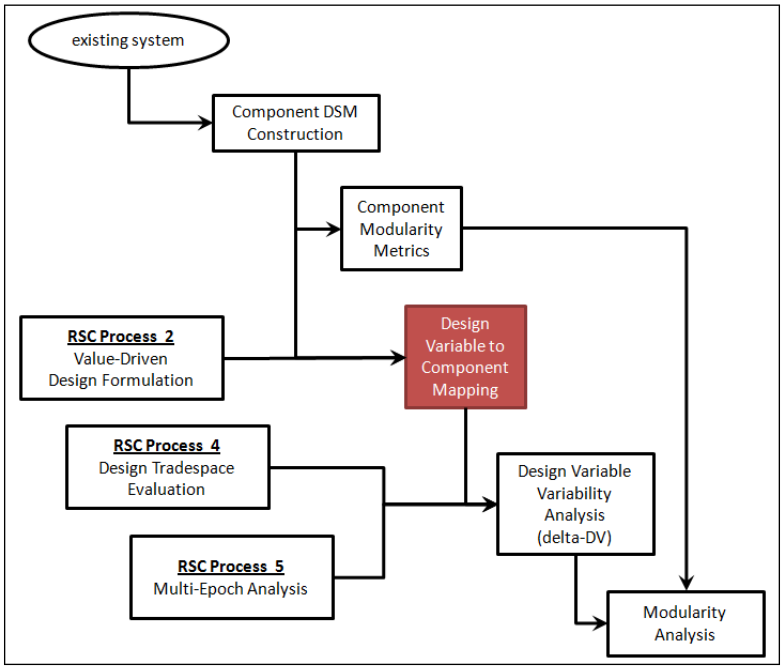
Design Vector		Attribute	Constraint
Variable	Range	Work rate	Row spacing
Engine power	[4-40] HP	X	
Vehicle width	72,90 inches		X

The mapping for the tractor epochs is presented in table 4.6. A difference here is that a design constraint in each epoch is also mapped to a design variable of the target system. A description and justification for each design vector element is called for:

Engine power represents the amount of engine power available, as measured by the brake power of the engine, that will then be translated to tractive force. The base design of the LifeTrac has an engine power of 28 horsepower (HP). The ranges were selected for the same reasons as stated earlier in the description of hydraulic power.

Vehicle width represents the tire-center to tire-center spacing. Because this is a constraint, depending on the epoch only one of the two levels will provide a feasible design in each epoch. That is, if considering the 24 inch row spacing epoch, only the 72 inch vehicle width will result in feasible designs (because 72 mod 24 = 0 and 90 mod 24 $\neq$ 0) in that epoch. The variable is listed here as it will need to have different values in different epochs to produce feasible designs, and is hence a variable that could affect the design of components within the system, making those components potential candidates for modularity analysis.

4.5 Design Variable to Component Mapping



The next activity of the design approach involves mapping from the design variables to the primary system components that are determined by a chosen design variable value. That is, for example, if the MATE analysis determines that certain engine power levels are Pareto optimal and likely to change between epochs, this mapping would provide

information about which components should be investigated for their level of modularity as compared to other components because it is these components that are likely to change in response to the change in design variable. This mapping is provided in table 4.7 with each component affected by the particular design variable marked as an 'X'. It is recognized that this is not a complete list of potentially affected components; it is desired to use the network centrality metrics from before to help one understand how likely it is for changes to these components to propagate through the system. As such, this is kept at a list of components that is relatively easy to deduce as being potentially affected. The justification for selecting each component is provided now:

Bucket width The main component affected by this design variable is the 'Loader - Attachment' component, the DSM entity that is for the attachment element of the LifeTrac.

Hydraulic power All components associated with generating and transmitting hydraulic power to the loader arm are determined by the setting of this design variable. Also affected is the length of the loader arm as it is a lever that transmit the hydraulic force to the attachment.

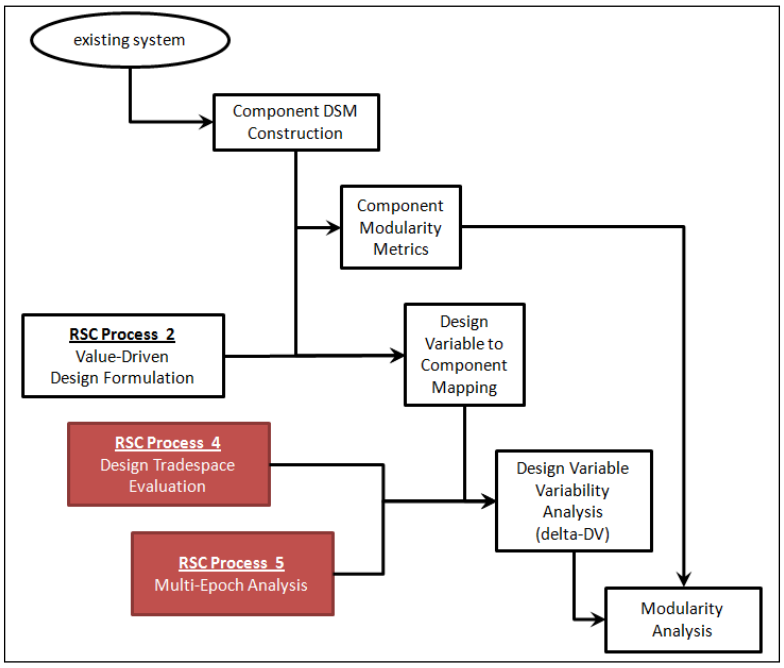
Engine power This list contains all the components that are responsible for generating and transmitting power to the wheels. These largely determine the ability to create tractive force required for pulling implements through the ground.

Vehicle width These are the main frame structural elements that determine the overall width of the vehicle.

Table 4.7: LifeTrac design vector to component mapping

		Design Variable			
		Skid Steer		Tractor	
i	Determined Component	Bucket Width	Hydraulic Power	Vehicle Width	Engine Power
1	Frame - Lower Section - Front piece			X	
2	Frame - Lower Section - Mid piece			X	
3	Frame - Lower Section - Back piece			X	
20	Loader - Arm - Left		X		
21	Loader - Arm - Right		X		
23	Loader - Arm - Hydr. Lift Cyl. - Left		X		
24	Loader - Arm - Hydr. Lift Cyl. - Right		X		
25	Loader - QAA - Hydr. Cyl. - Left		X		
26	Loader - QAA - Hydr. Cyl. - Right		X		
32	Power Cube - Engine		X		X
33	Power Cube - Hydr. Pump		X		X
42	Controls - Hydr. Controls		X		X
43	Wheel - Hydr. Motor - Front Left				X
44	Wheel - Hydr. Motor - Front Right				X
45	Wheel - Hydr. Motor - Rear Left				X
46	Wheel - Hydr. Motor - Rear Right				X
47	Loader - Attachment	X			

4.6 Tradespace Exploration



4.6.1 Design to Utility and Cost Mapping Models

The next step in the MATE process is to develop models that map each enumerated combination of the design variable options to both a multi-attribute utility value and a cost value. These models can be quite advanced or simple, depending on the complexity of the underlying system and the target level of fidelity. The models used in this case were kept purposefully simple and low fidelity, because the focus of the research is to connect MATE with modularity analysis. The performance models of LifeTrac were developed in Microsoft Excel 2010 and are a combination of physics-based and empirical models for system performance; the cost models were developed in the same fashion. Details of the portions of the models are provided in appendix D.

Table 4.8: Basic metrics from LifeTrac tradespaces (designs considered $n = 592$)

Tradespace	Feasible Designs	Designs $U(\underline{X}) \geq 0$
Skid steer, full soln. (epoch 1)	592	256
Skid steer, partial soln. (epoch 2)	592	256
Tractor, 30 in (epoch 3)	296	256
Tractor, 24 in (epoch 4)	296	272

4.6.2 Resulting Tradespaces

Four epoch tradespaces were developed for each of three scenarios: (epoch 1) the skid steer decision maker that is not satisfied with partial solutions, (epoch 2) the skid steer decision maker that is satisfied with partial solutions, (epoch 3) the tractor decision maker in the 30 inch row spacing epoch, and (epoch 4) the tractor decision maker in the 24 inch row spacing epoch. For each of these scenarios some basic metrics were calculated to determine the number of designs considered, the number of feasible designs, and the number of designs where $U(\underline{X}) \geq 0$ (see table 4.8). A brief discussion of these findings is in order at this time:

- All the designs in epochs 1 and 2 were considered feasible, but only a subset were $U(\underline{X}) \geq 0$. This largely can be owed to the fact that below a certain power range the model showed that the machine would not be able to perform to minimum needs. This may be attributed to the empirical nature of the model; a more detailed physics based model and a larger number and range of design variables could possibly lead to additional insights here. Nonetheless, for the modularity analysis this is not considered a threat to validity for the approach.

- Half of the designs in epoch 3 and 4 were considered infeasible; this is due to both ranges of vehicle width being considered in each epoch and that designs did not meet that epoch's constraint were considered infeasible. Of the designs remaining, the ones that resulted in $U(\underline{X}) < 0$ were due to the machine failing to generate enough tractive energy to meet minimum stakeholder requirements. The reason the number

of designs with $U(\underline{X}) \geq 0$ differed between epoch 3 and epoch 4 is because in epoch 4 less tractive force is required, owing to smaller plow sizes and lower resistance force being generated by the implement.

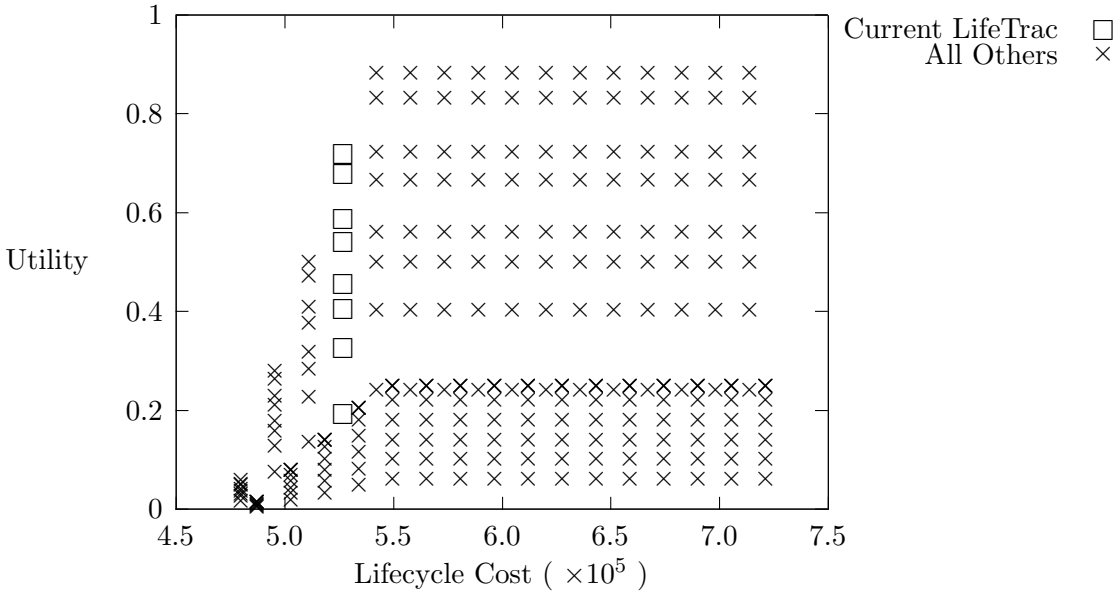


Figure 4-6: Skid steer epoch 1 tradespace

Owing to the low number of tradespaces, all four tradespaces will now be explored. The first tradespace to be considered will be the skid steer tradespace epoch 1 (full solutions), with the full tradespace shown in figure 4-6; noted in this diagram are the designs that represent the current LifeTrac design as well as all those explored. The existing design is represented as multiple design points in epoch 1: this is because the LifeTrac bill of materials does not specify a bucket size and, as such, all designs with engine power the same as existing LifeTrac (28 HP) are shown as the current design.

Inspecting the overall tradespace can provide high level insights, such as the location of non-dominated designs and general patterns. In figure 4-7 one sees that as the hydraulic energy increases (measured as a function of the engine brake power) that the cost and utility of the dominant designs increases; this continue up until a certain point where the

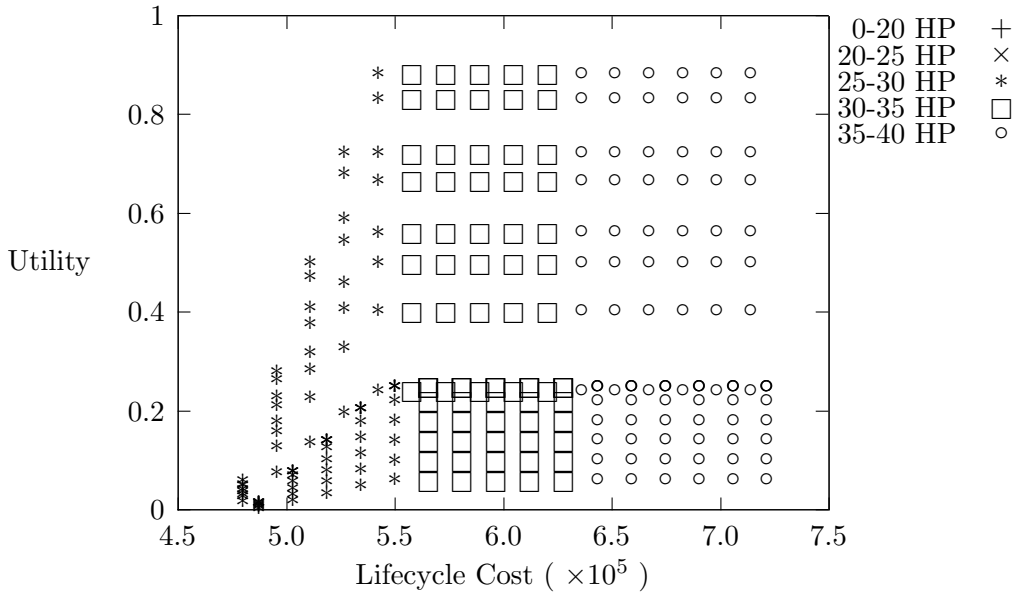


Figure 4-7: Skid steer epoch 1 tradespace, by hydraulic power

additional power does not provide additional utility. This would be an indication that additional analysis could be performed as to the sensitivity of the decision maker's utils.

In figure 4-8 one sees how the size of the bucket affects the utility-cost pairs. At the lower utility-cost levels several bucket sizes present the same optimal utility, largely because the limiting factor in achieving the ability to carry or lift material is limited by the overall hydraulic power and not by the volume of the bucket. In the higher utility-cost designs one sees a few bucket sizes dominate or nearly dominate (72 inch and 76 inch width) as the utility of maneuverability becomes important.

The next tradespace to be considered is epoch 2, the skid steer tradespace where partial solutions are acceptable. The full tradespace is given in figure 4-9, and the tradespace by hydraulic power is given in figure 4-10. These tradespaces have no interior points because of a combination of two factors: the presence of only one attribute for utility and the analytic formula for tractor performance having only one solution for each power level. Notice also that the cost ranges for the tractor epochs to achieve $U(X) \geq 0$ is lower for

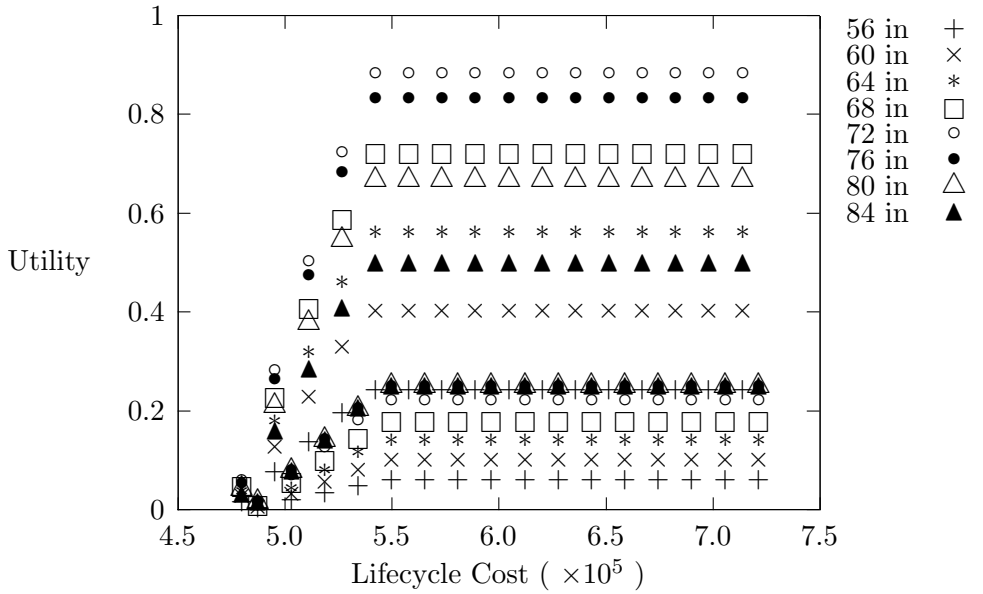


Figure 4-8: Skid steer epoch 1 tradespace, by bucket width

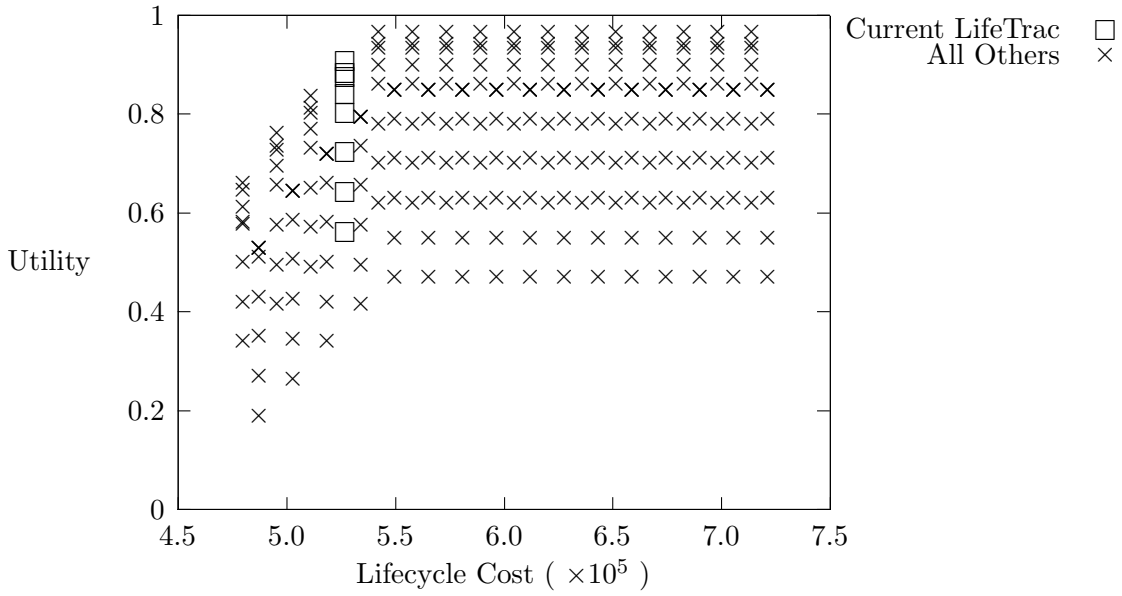


Figure 4-9: Skid steer epoch 2 tradespace

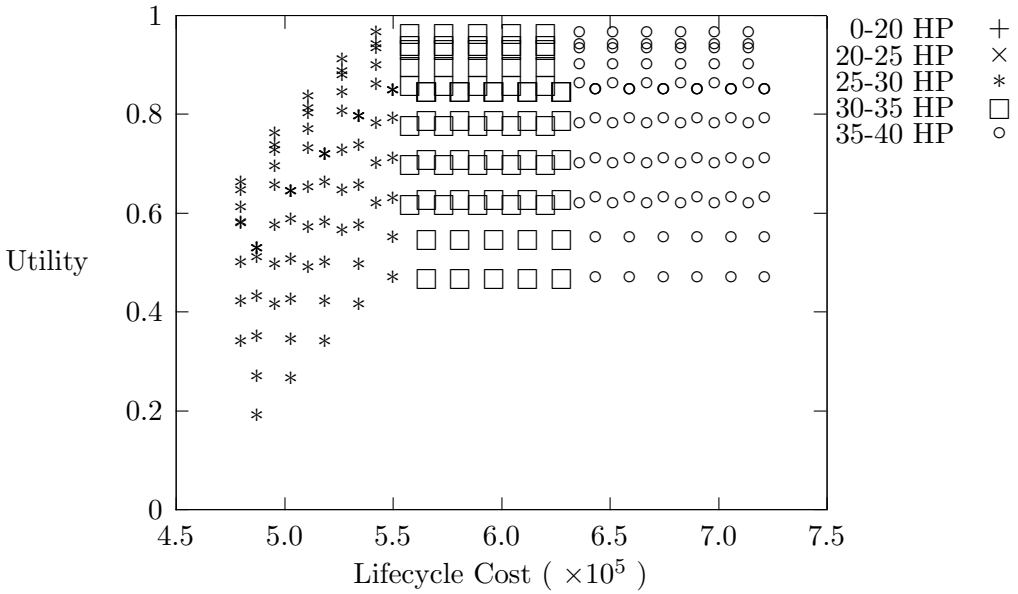


Figure 4-10: Skid steer epoch 2 tradespace, by hydraulic power

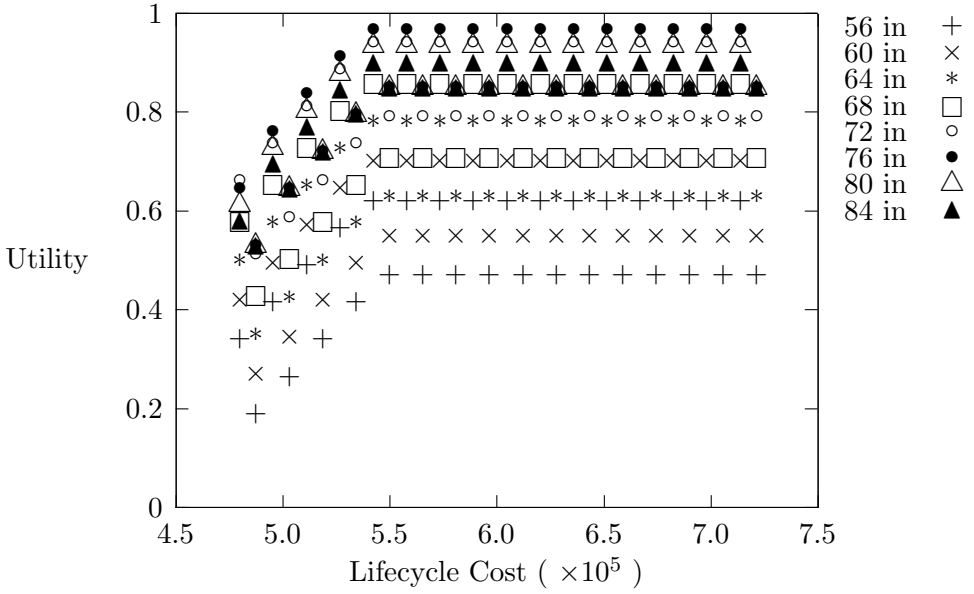


Figure 4-11: Skid steer epoch 2 tradespace, by bucket width

the tractor epochs than the skid steer epochs: this is because the tractor decision maker's preferences allowed for a lower engine power in order to provide utility as compared to the skid steer scenario, leading to a lower lifecycle costs. For these parameters there is not much information to be gained from this compared to the same figures from epoch 1. Insight is to be gained by inspecting the tradespace by bucket size as shown in figure 4-11. In this case we see a larger range of bucket sizes (72-80 inches) near the Pareto front; this is owing largely to the additional load that can be carried in the larger bucket and the decreased trade-off requirements of partial solutions. This analysis, again, speaks to the need for additional sensitivity testing if one were completing a more robust tradespace analysis.

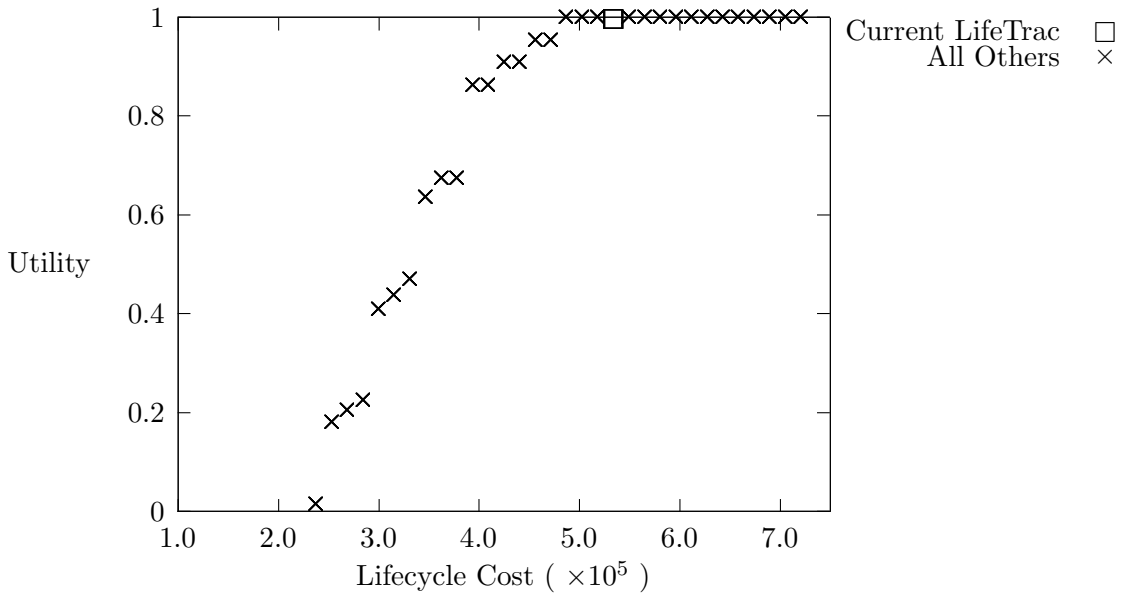


Figure 4-12: Tractor epoch 3 tradespace

The next tradespace to be considered will be the tractor tradespace during the 30 inch row centers epoch, shown in figure 4-12; this figure shows all of the considered designs and highlights the current LifeTrac design. The current LifeTrac design does not support this width, but if it were changed to support the field configuration of this epoch it would

have $U(\underline{X}) = 1.0$ and cost of USD53430 and is dominated by other design options in this tradespace. Finding an existing design not on the Pareto optimal could be attributed to several things, such as performing value based exploration of the universe of designs instead of selecting a point design and optimizing around it, or this decision makers attribute curves; due to this last issue some sensitivity analysis might be appropriate in a more full complete tradespace analysis.

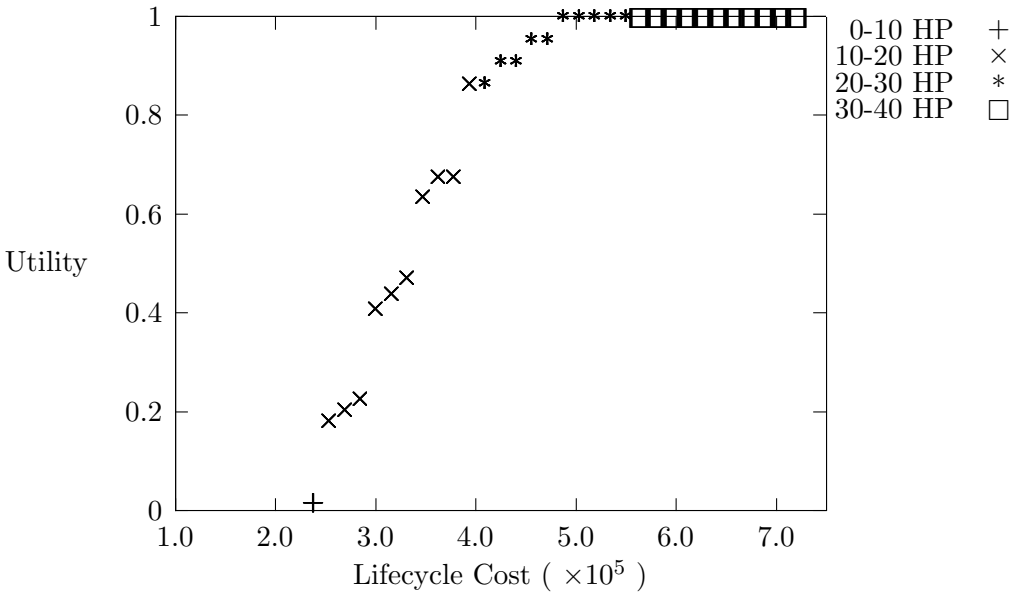


Figure 4-13: Tractor epoch 3 tradespace, by engine power

The next look at the tradespace is by ranges of engine power shown in figure 4-13. Width is not considered a design variable because it is constrained to a single value based on the epoch, with epoch 3 constraining the width to be a multiple of 30 inches and epoch 4 constraining the width to be a multiple of 24 inches.[4]. This has some steps structures in the model, owing to the underlying physics associated with performing the intended

[4]Alternatively width could have been considered a design variable with two possible values, one at 90 inches and one at 72 inches (multiples of 30 and 24 inches), instead of a per epoch constraint. In this scenario all designs of 72 inches would have been categorized as infeasible in epoch 3 and all designs of 90 inches would have been categorized as infeasible in epoch 4.

operation (that is, it takes a certain increment of additional power to be able to achieve a higher work rate). Some designs have $U(\underline{X}) < 0$ and these are due to not being able to achieve the minimum required work rate. In general, as one goes up the cost-utility Pareto frontier, one finds higher engine power systems. This continues until the maximum work rate utility is achieved at which point additional engine power only increases cost, not utility.

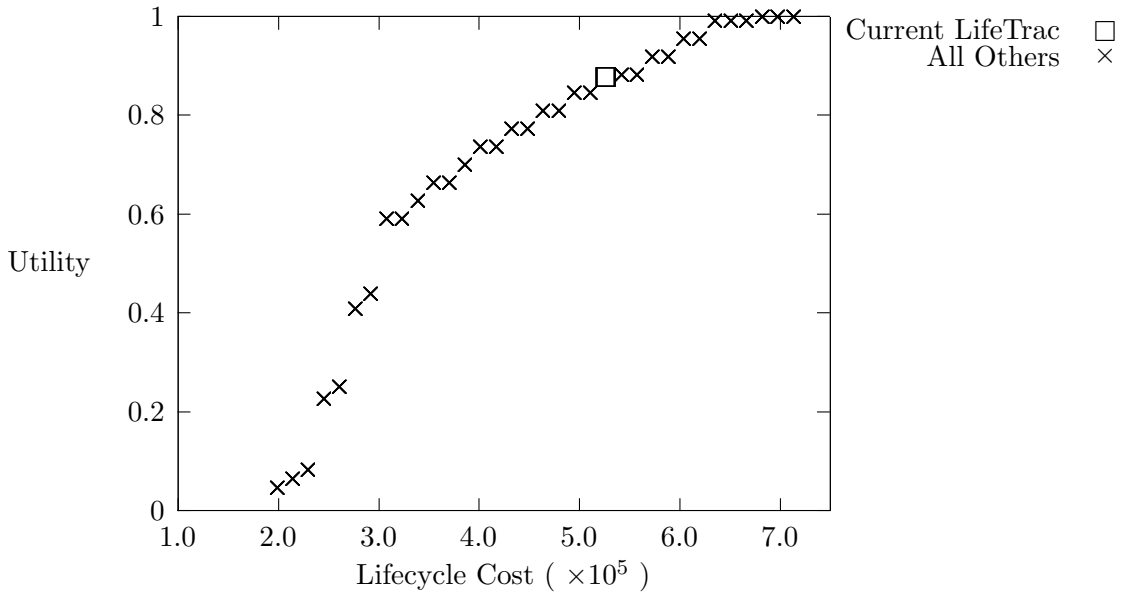


Figure 4-14: Tractor epoch 4 tradespace

The last tradespace to be considered will be the tractor tradespace during the 24 inch row centers epoch, shown in figure 4-14; this figure shows all of the considered designs and highlights the current LifeTrac design. The current LifeTrac design has $U(\underline{X}) = 0.92$ and cost of USD57360 and is non-dominated design in this tradespace.

The next look at the tradespace is by ranges of engine power (the only design variable), shown in figure 4-15. Again, we see the structures owing to the physics of the nature of plowing. Some designs have $U(\underline{X}) < 0$ and these are due to not being able to achieve the minimum required work rate. In general, as one goes up the cost-utility Pareto frontier,

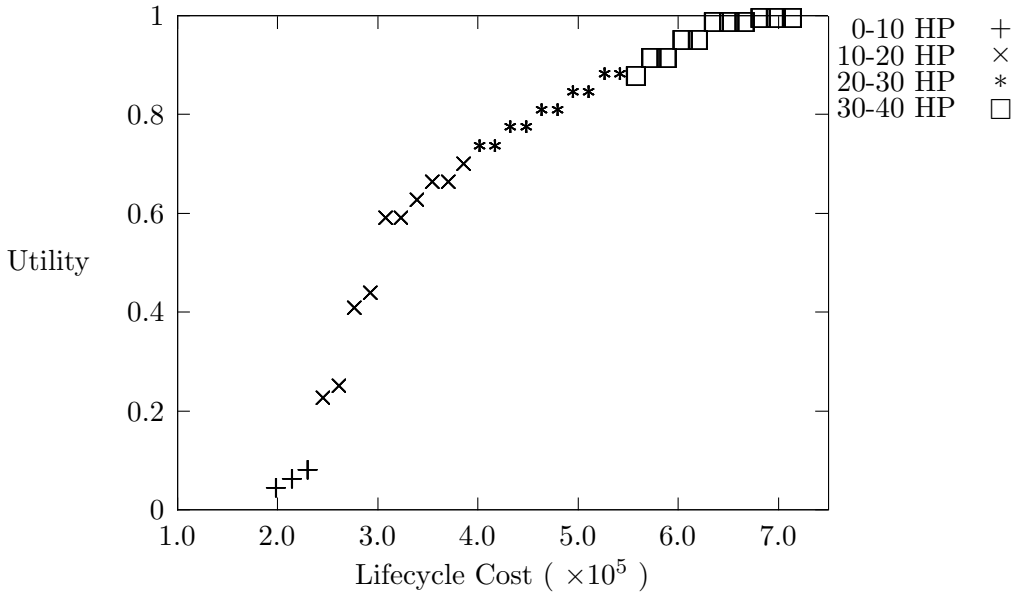
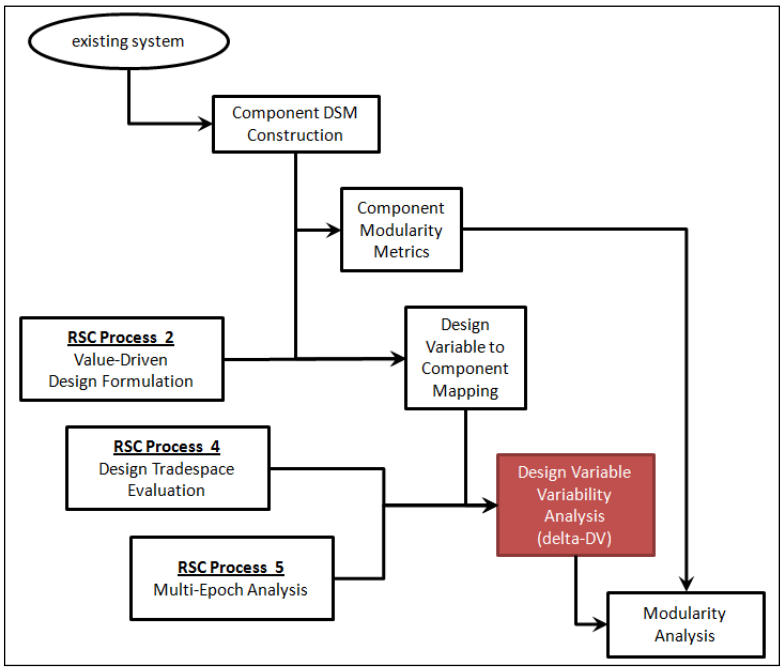


Figure 4-15: Tractor epoch 4 tradespace, by engine power

one finds higher engine power systems. As compare to the epoch 3 tradespace, less designs here achieve $U(\underline{X}) = 1$; this is owed to the lower area of the field covered per unit time during operation and the resultant inability of this configuration to achieve the maximum desired work rate for the consider design variable range.

4.7 delta-DV Development



The purpose of this section is to develop a list of system components that are likely to change as a result of design variables changes; these system components will then be carried forward in the design approach for further investigation. In this case, the technique of using insights from the (fuzzy) Normalized Pareto-trace technique from Ross et al. (2009b) will be used to aid in the down selection to a set of promising designs; these metrics are used to screen all designs down to a set of designs which are most widely considered valuable and from that set determine how to connect the selected designs via component modularity. From the set of down selected designs we will investigate changes in design variables across the selected designs. The design variables that are likely to change are traced back to the linked components done during the design variable to component mapping earlier (see table 4.7) to determine if those components exhibit relatively high- or low-levels of modularity as an indication of the system's ability to support the changes.

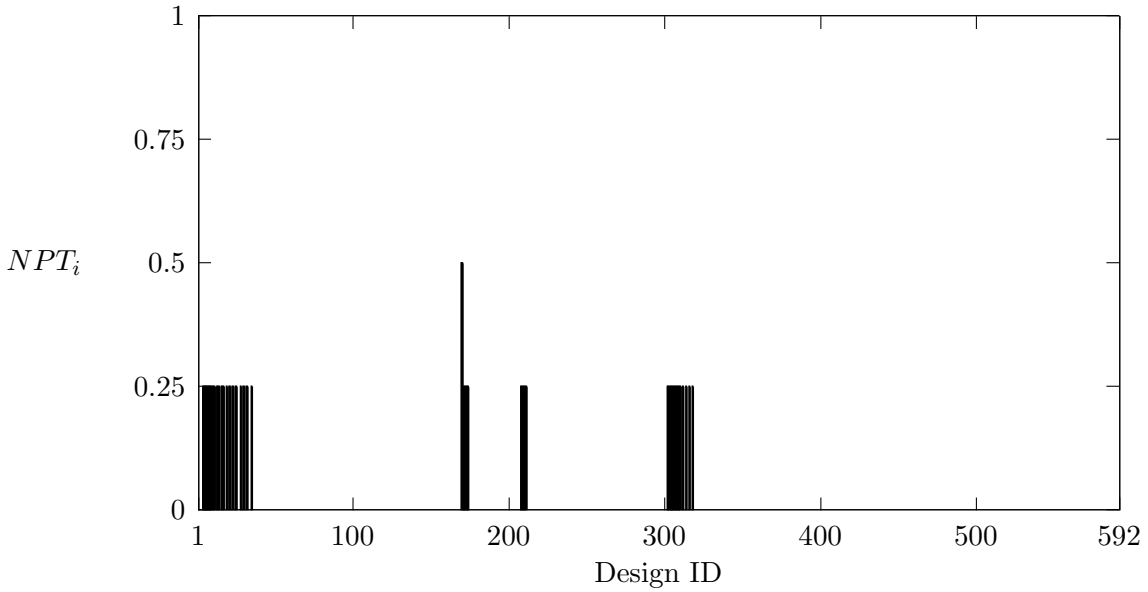


Figure 4-16: Normalized Pareto Traces by design

Our first step will be to use the Normalized Pareto Trace (NPT_i) metric from (Ross et al., 2009b).[5] This technique is used to determine if there are any counters to modularity needs; if one can find a design that is always "good" then there may be no reason to create a design that is changeable. In this technique one is searching for designs that are non-dominated across all epochs; a design with $NPT_i = 1.0$ would be non-dominated, that is, on the Pareto front across all the considered epochs and would always be considered a "good" design by exhibiting passive value robustness. The results of the Pareto trace is shown in figure 4-16. The biggest item of note is that no designs achieve the goal of being non-dominated across all epochs, which would be indicated by $NPT_i = 1.0$. One reason for this is because no designs are common between epoch 3 and 4 owing to the width design constraint. One design, design number 170, is non-dominated across two epochs, Epoch 1 and Epoch 2 (as a reminder, these are the two skid steer use case epochs). Design 170 has

[5]While this case uses (fuzzy) Pareto Trace to help select the designs, this is not the only technique available and one should use selection criteria that fits one's goals.

engine power of 25 HP, vehicle width of 72 inches, and bucket width of 72 inches.

Additional analysis, fuzzy Normalized Pareto Trace ($fNPT$), will be used in an attempt to reveal some designs that, while not being strictly non-dominated, could be potentially valuable across epochs. The fuzziness factor K will be used to denote a tolerance to uncertainty and will be varied from 0.00 to 0.10 in steps of 0.02 (the situation of $K = 0$ is the same as the strict Pareto Trace completed earlier) (Ross et al., 2009b). The results of this are shown in figure 4-17. Again, we would expect to find no designs achieving a $fNPT_i = 1.0$ because of the constraint on width across epochs 3 and 4. As K is increased we see more and more designs within an acceptable range of being a potentially valuable design. Upon inspection, it is when $K = (0.02, 0.04]$ that we see designs that, across the Epoch 3 and Epoch 4 constraints, have a fuzzy Normalized Pareto Trace value of 0.75; this means that a design is good in both construction use epochs and one agricultural use epoch.

Other criteria could be used as a screening metrics, but for this case the set of designs such that $fNPT_i \geq 0.75$ when $K = 0.04$ will be initially selected and inspected. These are presented in table 4.9. The column label Fuzzy Pareto Number in this table indicates the minimum required K value for the design to be have $fNPT_i \geq 0.75$ (the maximum number achieved by any design); this is an indicator of the amount of compromise required for value robustness across epochs. A handful of descriptive statistics (minimum, maximum, and variance) are included for each design variable across the selected designs; these are included to provide the designer with basic information about the amount of variation of the design variables across the selected designs. It is suggested that the designer could use such descriptive statistics to gain more insight into the amount of variability (or lack thereof) that may need to be supported in that design variable.

By calculating the variance for each design variable across the designs and seeing all variances > 0, one can determine that all design variables could change across the selected, potentially valuable designs, as well as gain insights into which design variables exhibit

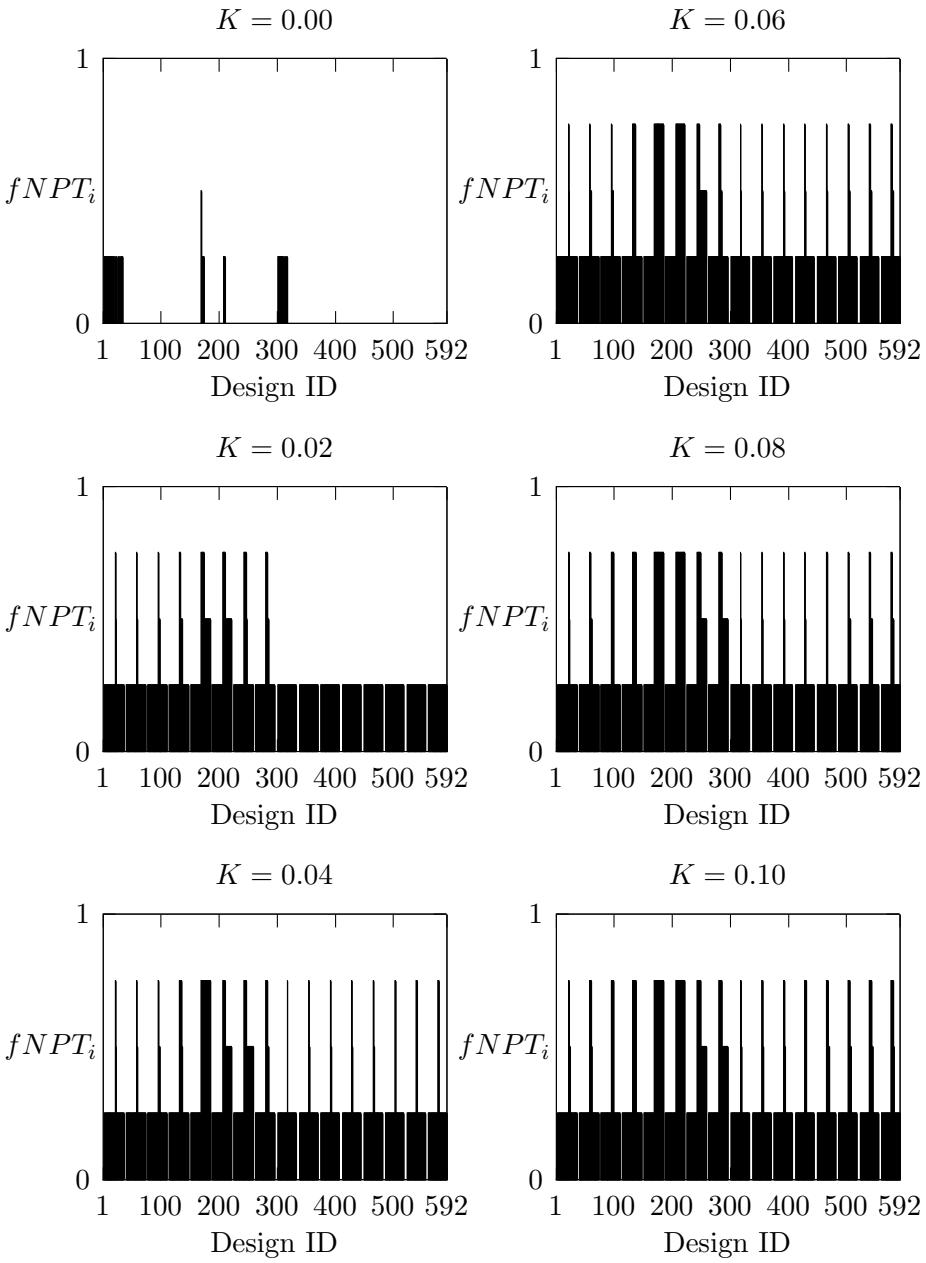


Figure 4-17: fuzzy-Normalized Pareto Traces by design

Table 4.9: Selected LifeTrac designs

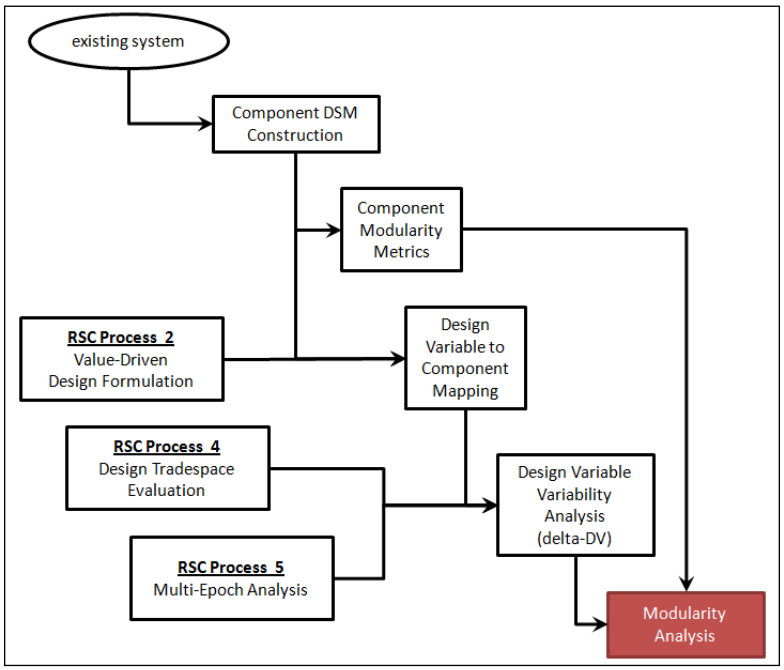
Design ID	Fuzzy Pareto Number	Hydraulic / Engine Power (HP)	Vehicle Width (in)	Bucket Width (in)
22	0.01	25	72	56
59	0.01	25	72	60
96	0.01	25	72	64
133	0.01	25	72	68
134	0.01	26	72	68
135	0.04	27	72	68
136	0.04	28	72	68
170	0.01	25	72	72
171	0.01	26	72	72
172	0.01	27	72	72
173	0.01	28	72	72
174	0.01	29	72	72
175	0.04	30	72	72
176	0.04	31	72	72
177	0.04	32	72	72
178	0.04	33	72	72
179	0.04	34	72	72
180	0.04	35	72	72
181	0.04	36	72	72
182	0.04	37	72	72
183	0.04	38	72	72
184	0.04	39	72	72
185	0.04	40	72	72
207	0.01	25	72	76
208	0.01	26	72	76
209	0.01	27	72	76
210	0.01	28	72	76
211	0.01	29	72	76
244	0.01	25	72	80
245	0.01	26	72	80
246	0.01	27	72	80
247	0.01	28	72	80
281	0.01	25	72	84
282	0.01	26	72	84
283	0.01	27	72	84
318	0.04	25	90	56
355	0.04	25	90	60
392	0.04	25	90	64
429	0.04	25	90	68
466	0.04	25	90	72
503	0.04	25	90	76
504	0.04	26	90	76
540	0.04	25	90	80
541	0.04	26	90	80
577	0.04	25	90	84
578	0.04	26	90	84
Min		25	72	56
Max		40	90	84
Variance		18.4	60.3	48.5

variance across these potentially valuable designs. Therefore all components in the mapping table 4.7 will be used in the modularity analysis in the next section. If a design variable had not varied across the selected designs, or had some relatively low level of variability as defined by the decision makers, then there would be no need to inspect those modularity metrics in the next step.[6]

Another way to view the previous analysis is as a way to layer changeability onto the most passively robust designs identified through the fuzzy Pareto Tracing. The goal of layering changeability onto the most passively robust designs is to achieve a high effective value robustness, one that leverages both passive and active robustness. Modularity is the approach to changeability focused on in this thesis, with the implication being that one wants to link together the mostly passively robust designs into a completely robust (passive and active) design through component modularity. This is the same as attempting to find or synthesize, through component modularity, designs with an unitary effective fuzzy Normalized Pareto Trace, $efNPT = 1$ (Fitzgerald et al., 2012a).

[6]The selection of designs to further consider determines the variation in the design variables, leading to a decision as to which components will be further analyzed. A different selection of potentially valuable designs could have led to different variables to be considered. This case, as presented, assumes that one is only interested in transitioning between high fuzzy Pareto Trace designs. Another criterion that could have been used is that one wants to transition between non-dominated designs within a given epoch. This additional criterion could have been used but was not in this case study.

4.8 Modularity Analysis



As a review, at this point in the design approach we now have a set of potentially valuable designs, as well as insights into which design variables exhibit variance across these potentially valuable designs. If this were a product for commercial markets, the providing organization might want to offer all the designs to the market as a options for different consumers (for example, in the language of (Meyer and Lehnerd, 1997), good, better, best product options). Or, the scenario in question could be for a single decision maker that desires a level of skid steer functionality at some time $t = 0$ but could foresee needing a higher level of functionality (utility) at $t > 0$. An alternative scenario could be that a single tractor decision maker is in epoch 1 (30 inch rows), they may want to switch to epoch 2 (24 inch rows) in the future. With each of these change scenarios we now want to understand the relative effort of supporting those different designs and design changes inside of the given product architecture. Now that we understand the design variables that may need to be changed either now or in an uncertain future, the next step focuses on answering the

question, "What might component modularity tell the designer about the design's ability to support these changes?"

4.8.1 Component by component analysis

The first step is to review the modularity metrics developed in section 4.3.2 and provided in detail in appendix C for each component affected by each changing design variable. The mapping between design variable and components that are determined by that design variable were originally given in table 4.7. Because all considered design variables across the selected designs have some level of variability, each design variable and its determined components will be considered in turn in the subsequent analysis.

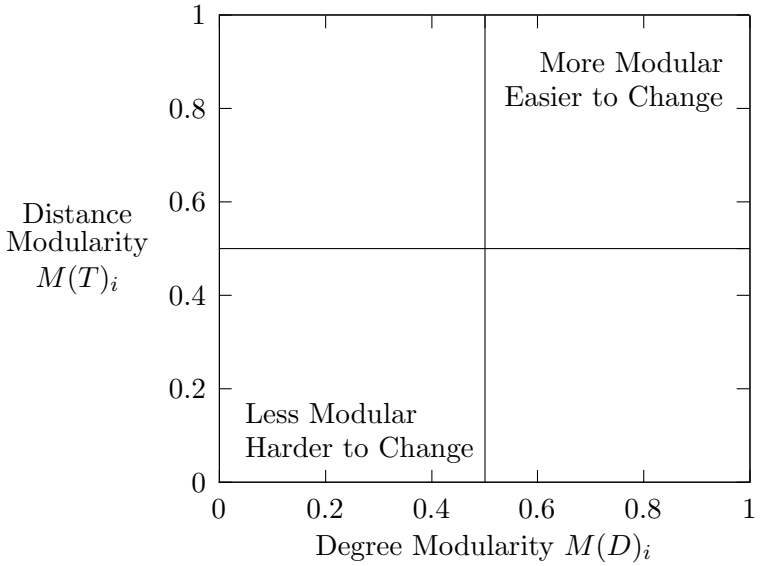


Figure 4-18: Key to interpreting modularity plots

An example of the analysis can be found by looking at the modularity metrics from the perspective of cost of transition for a component set from one design variable value to another value. Sosa et al. (2007) shows that components with high degree and distance modularity are the most likely to be redesigned in order to achieve different levels of per-

formance. The original hypothesis for this finding was based on the idea that components that have low levels of direct connectivity to other components, measured by higher degree modularity $M(D)$, and are far away from other components, measured by higher distance modularity $M(T)$, are the easiest to change and hence would the most likely candidates for planned changes. Another way to state this finding is that as degree and distance modularity increase for a component, it would cost less to transition it from one design point to another as its change would cause the least amount of design change propagation to other components; this is represented graphically in figure 4-18. We can look at each possible design change scenario and the related components. This will be done by plotting $M(T)$ by $M(D)$ for all components in the system and highlighting the key components, based on the mapping done in table 4.7, needed to realize a certain design change scenario. At this point qualitatively analysis will be completed for each of the four change scenarios.

The following plots were generated by calculating the the modularity metrics, based on the DSM introduced in section 4.3.1. The 'selected components' are those determined by the design variable for each plot. Because there were four design variables that had variability across the selected designs, there are four plots to consider.

Skid steer, bucket width See figure 4-19. The single critically affected component has the highest level of degree and distance modularity compared to all other components in the system. We would expect the cost of change from one level of this design variable to another would be relatively low cost. This conclusion is supported by existing skid steer market designs, as the quick attach assembly (QAA) component is designed such that attachments, like a bucket, can be easily swapped by the end-user.

Skid steer, engine power See figure 4-20. The components affected by this change, with one exception, have degree modularity in the top half of the range and distance modularity in the lower half of the range; one interpretation of these modularity metrics is that most of the components have a middle level of modularity and some

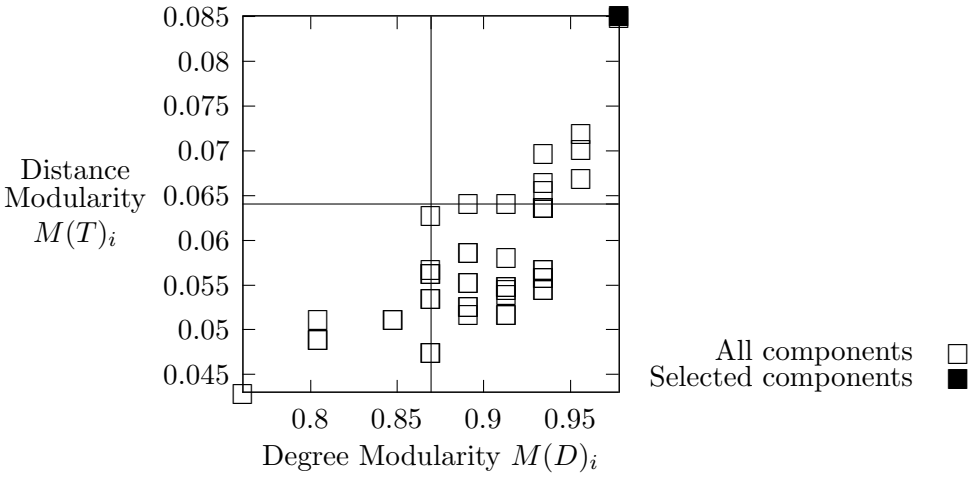


Figure 4-19: Modularity of skid steer bucket width modules

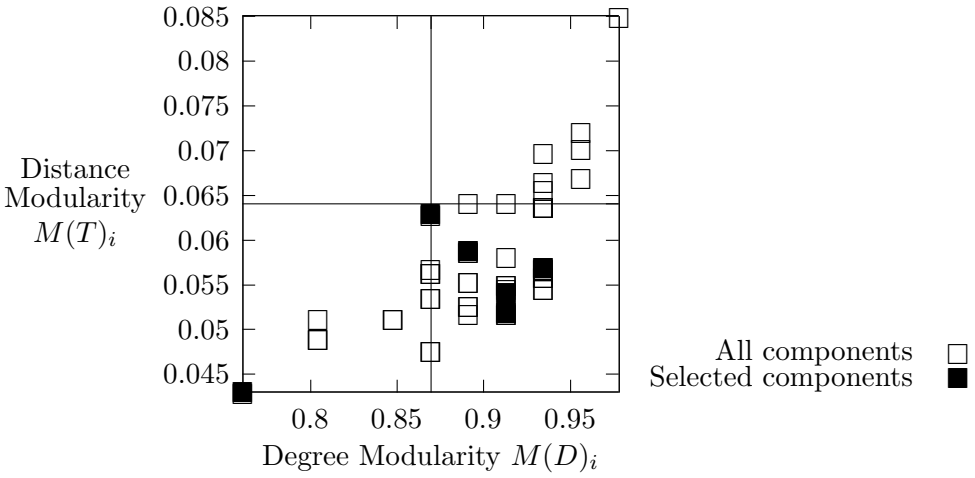


Figure 4-20: Modularity of skid steer engine power modules

ability to easily change and one component has a low level of modularity and a low ability to change, all relative to the other components in the system. The component with the lowest level of modularity is the hydraulics control component. The values of the modularity metrics for the target components indicate this design change would be more costly, due to propagation to other components, than the skid steer bucket width change. Also, because of the low level of degree and distance modularity of the hydraulics component, a designer might want to pay special attention to the design of that component possibly by building in extra capacity or developing extra interface components that will shield this component from outside change and propagating change. In the scenario of adding additional capacity to the components, this could help make the component a change propagation absorber. This type of design decision would not have an affect on component modularity in this case example as the binary DSM would not have changed as a result of changing the design of the component. If the DSM had represented strength of dependency between components on a scale, as was done in Sosa et al. (2007), then adding the extra capacity would have changed the modularity metric values for this component because the strength of design dependency between the component and other components would have changed[7]. In the scenario of adding interfaces, the binary DSM would have additional element(s) introduced and the modularity metrics would need to be recalculated to determine the impact on modularity of this component[8].

Tractor, width See figure 4-21. The components affected by this change have degree

[7]The decision as to construct a binary DSM or a DSM that includes the strength of the dependencies between the components is a trade-off between DSM construction effort and the fidelity of the resulting modularity metrics. While the binary DSM is lower effort to construct than a DSM that includes the strength of the dependencies, the modularity metrics resulting from the binary DSM can only resolve if there is a dependency between components whereas a DSM that indicates the strength of the dependencies between components would be able to resolve a more detailed view of a components modularity. In the light that trade-space exploration is done during early design phases of a system it may not be pratical to construct a DSM that includes design dependency constraints between components.

[8]It is anticipated that the modularity metrics for hydraulics control component would improve as a result of adding extra capacity to that component. It should also be noted that the extra capacity would likely come at the expense of additional cost and/or weight to the total system.

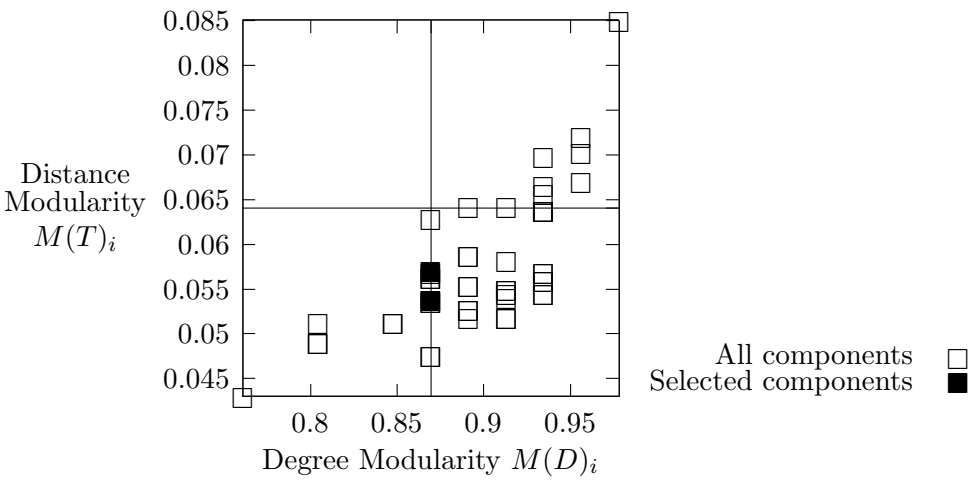


Figure 4-21: Modularity of tractor width modules

modularity in the middle of the range and distance modularity in the lower half of the range. These would indicate that this design change may be costly to make, as the change may propagate to other components. This is again an area where a designer might employ technology to affect the change propagation of these components. Alternatively, one might make a trade-off decision to only support one width or another and abandon the option to vary the system in this dimension of the design variable (that is, consider the width non-tradeable). The implication of this is if one would need to provide two distinct system options to cover the epoch-space, rather than one system that can change to accommodate the widths. This decision as to whether to offer two distinct system options or one system that can change would then be a cost-benefit calculation that considers both the supplier and the decision maker and the user. As a example, the user/decision maker, would need to determine if the switching cost plus the added complexity of the single modular solution would or would not exceed the dual acquistion, storage, maintenance, and other costs of

two distinct systems.

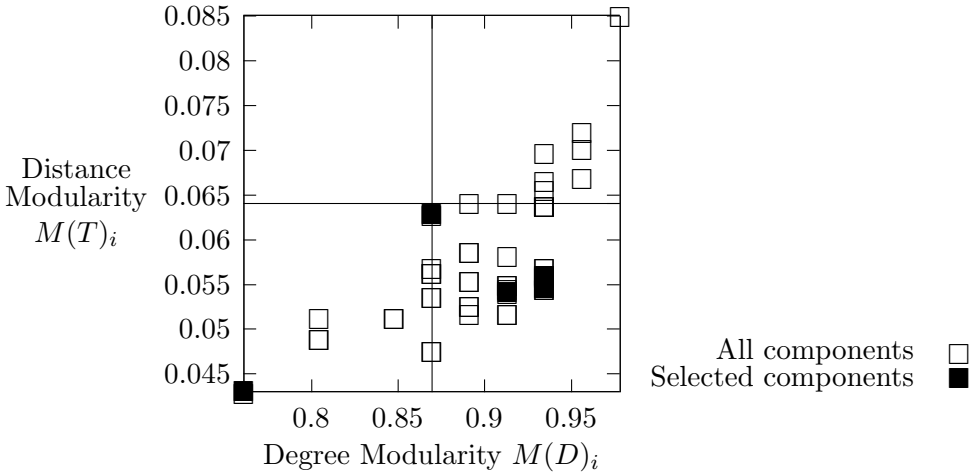


Figure 4-22: Modularity of tractor engine power modules

Tractor, engine power See figure 4-22. This component has much of the same profile as the skid steer engine power components in terms of modularity. The analysis there would apply here as well.

4.8.2 Connection to Tradespace Network Analysis

One other use of the modularity metrics might be using them as input into the rule-effects matrix and subsequent tradespace network analysis. At this point we will build up a rule-effects matrix that might result from this analysis. This is only one rule-effects matrix, but the justification for each entry will be made and tied back to the tradespace analysis and modularity analysis.

Baldwin and Clark (2000) speculate on the use of modularity in various phases of the lifecycle of a product, labeling them as Modularity-in-Design (MID), Modularity-in-Production (MIP), and Modularity-in-Use (MIU). The construct of modularity in a par-

ticular lifecycle phase will be adapted here to build the transition rules for the LifeTracs design vector. The suggested design rules are included in table 4.10. These rules were built from the modularity analysis completed previously based on the values of the modularity metrics, as indicators as to when certain transition might be able to occur. As an example of the results of that analysis, change in engine/hydraulic was deemed to only be possible during the design phase owing to the low modularity metric values of some components associated with changing the power design variable (see "Skid steer, engine power" item in section 4.8). The change agent origin column indicates if an external agent must act on the system to invoke the underlying rule or if the system can make the change internally without input from an outside agent. Externally invoked changes are called 'Flexible' changes and internally initiated changes are called 'Adaptable' changes Ross and Rhodes (2011).

Table 4.10: LifeTrac transition rules

Rule	Description	Change Agent Origin
R1: Frame-Design	Change in frame width during design	External (Flexible)
R2: Attach-Design	Change in attachment size during design	External (Flexible)
R3: Power-Design	Change in engine/hydraulic power during design	External (Flexible)
R4: Frame-Prod	Change in frame width during production	External (Flexible)
R5: Attach-Prod	Change in attachment size during production	External (Flexible)
R6: Attach-Use	Change in attachment size during use	External (Flexible)

From the list of transition rules, a rule-effects matrix can be constructed with the six transition rules and the effect each rule could have on each design variable in table 4.11 (see Ross and Rhodes (2011) for more information about rule-effects matrix construct). The columns listed under design variables are the design variables that would change if a particular transition rule were executed. An up arrow (↑) would indicate that that transition rule only changes the indicated design variable(s) to a higher level, a down arrow (↓) would indicate that that transition rule only change the indicated design variable(s) to a lower level, and the combination of the up and down arrows (↑↓) are used to indicate that that rule allows the given design variable to change in level either upward or downward.

The next set of columns are the modularity path enablers. The enablers are additional design variables that enable the transition from one design to another at a lower cost or shorter schedule than would be available if they were not present. In the path enabler columns, the presence of a slash ('/') indicates that that path enabler is optional for that particular rule to be available, but if that path enabler is present it may lower the cost and/or time of the transition. An 'X' in a path enabler column indicates that that path enabler is required in order for that transition to be executed. The absence of any entry in the path enabler columns, as is the case with Rule 3, indicates that that transition can be executed but at a cost and/or time impact that would be higher than if a path enabler were present. Two path enablers are then considered, modularity in attachment (allowing the bucket width to be changed at a lower cost and lower time as compared to if that path enabler were not present) and modularity in vehicle width (allowing the vehicle width to be changed at a lower cost and lower time as compared to if that path enabler were not present). These path enablers were considered because the prior modularity analysis indicated that these path enablers may already be in place in the underlying system.[9] R3 has no path enablers because the modularity metrics indicate that there is a relatively low level of modularity in the components required to make this transition. As stated above (see "Skid steer, engine power" item in section 4.8), a designer could take action to make the affected components more modular, hence creating a path enabler. The decision to do this would be determined by a tradeoff between cost and changeability.

It is important to note that this is a different approach to constructing this matrix than the Responsive Systems Comparison method because this case is inheriting an existing system for analysis instead of starting with a clean sheet design. Indeed, the entries here, depending on the design that is moved forward, could be specified as necessary to dictate downstream modularity requirements as discussed in the next section.

While not completed here, the subsequent tradespace network analysis, which uses the

[9] A review of the LifeTrac website indicates that the ability to change attachments at low cost and with little time was a design goal of the system.

Table 4.11: LifeTrac Rule-Effects Matrix

	Design Variables			Path Enablers		
Rule	Frame Width	Attachment Size	Engine Power	Modularity in Frame	Modularity in Attachment	Change Origin
R1: Frame-Design	↑↓			/		Flexible
R2: Attach-Design		↑↓			/	Flexible
R3: Power-Design			↑↓			Flexible
R4: Frame-Prod	↑↓			X		Flexible
R5: Attach-Prod		↑↓			X	Flexible
R6: Attach-Use		↑↓			X	Flexible

rule-effects matrix as input, could be used to further downselect the potentially valuable designs. For example, the tradespace network analysis results could be used as a further refinement in addition to the the previously used fuzzy Pareto Trace and the value of changeability for each design could be assessed using measures such as effective fuzzy Normalized Pareto Trace (Fitzgerald et al., 2012a,b). This additional analysis would then include the cost or performance implications of modularity (as argued in (Whitney, 2003)) in a more general way than is found in (Baldwin and Clark, 2000), where it is assumed that modularity does not change performance of the system.

4.9 Alternative Approach Flow

This case presents an approach that begins with an existing system architecture, represented in a design structure matrix as system components and their dependencies and analyzed by the modularity of the components, and then moves through a tradespace study to gain further insights into potentially valuable design vectors, and then compares the findings from the modularity of the components to the need to vary those components to achieve valuable designs. Many product development efforts start in this fashion, with

an existing design and then make modifications to that design to achieve a different bundle of attributes that are valuable to decision makers or markets. It is with this understanding that the prior case was selected and the approach flow proposed.

However, there are scenarios where no existing product exists and the development effort is more green field, more open to considering many more dimensions in the design space as well as values of the design vectors. This work did not attempt to make a connection to that scenario, but allow us to suppose one that could be validated with an additional research effort. In this scenario, it is imagined, that the Responsive Systems Comparison method would be followed, including the development of the rule-effects matrix. The rule-effects matrix, since we are focused on modularity, would include rules such as those from the section 4.8.2, describing design variables that need to be changeable in various lifecycle phases, such as 'in-design', 'in-production', or 'in-use', with corresponding path enablers that are modularity path enablers. The modularity path enablers might even describe a required level of modularity in terms of the modularity metrics, and these levels would be used to create the transition matrices (as is done in RSC). From there if a design were selected that included a modularity path enabler, during downstream development (as in something like the concurrent engineering part of the MATE-CON process (Ross et al., 2004)) these modularity path enablers would be flowed down as requirements on the components of the system that contributed to achieving a particular design variable. As design fidelity increases and the architecture is more fully described using a DSM or network representation, the achievement of those modularity requirements could be checked using the modularity metrics specified in the tradespace exploration phase.

4.10 Discussion

From these findings one can create a set of designs and path enablers which results in a product line (Meyer and Lehnerd, 1997) to meet the highlighted needs. This product line could be represented as three products, with catagories being good, better, and best. The

"good" offering would be a design with minimally acceptable utility at low cost, and the 'best" offering being at or near maximum utility at a high cost point, and "better" being somewhere in between. Further, one could suppose the product line needs to account for the transition from epoch 3 to epoch 4 (different row spacing). The presence of modular path enablers would then connect the various designs.

As an example, allow us to start with design 22 from table 4.9. With modularity path enablers in place for changing the attachment width and changing the width of the product, one could transition from design 22 to the remainder of the designs in table 4.12 through executing the modularity path enablers.

Table 4.12: Example designs reachable through modularity path enablers

Design ID	Hydraulic /Engine Power (HP)	Vehicle Width (in)	Bucket Width (in)	Reachable Via[a]	Reachable Phase[b]
22	25	72	56	B	-
170	25	72	72	AM	PU_{AM}
281	25	72	84	AM	PU_{AM}
318	25	90	56	FM	P_{FM}
466	25	90	72	AFM	PU_{AM}; P_{FM}
577	25	90	84	AFM	PU_{AM}; P_{FM}

[a] B = Baseline, AM=Attachment Modularity, FM=Frame Modularity, AFM=Attachment and Frame Modularity, meaning both types of modularity are required

[b] PU_{AM}=Production or Use for Attachment Modularity, P_{FM}=Production Only for Frame Modularity

From this shorter list of designs, we will inspect the utility information to determine what designs could be offered into the target markets. Beginning with the construction epochs, the utility information is given in table 4.13 for each of the designs from table 4.12. This new list of designs is sorted in descending order of utility in epoch 1, in order to make determination of the "good, better, best" offerings easier to descern. Cost was not included in the analysis, because for commercial products offered into a market the production cost is not correlated with value to the customer. Instead utility indicates value to the customer

and the product would be offered at a price to the customer based on utility; the difference in cost to produce and the price realized in the market is the resultant margin delivered to the manufacturer of the product. The "good, better, best" determination was done via intelligent inspection; this is an example determination and, based on the goals of the producing entity, other determinations may be made.

Table 4.13: Product offerings for construction epochs (epochs 1 and 2)

	Hydraulic /Engine	Vehicle	Bucket	Utility - Construction		Product
Design ID	Power (HP)	Width (in)	Width (in)	Epoch 1	Epoch 2	Offering
577	25	90	84	0.079	0.645	
170	25	72	72	0.060	0.662	Best
281	25	72	84	0.031	0.579	Better
22	25	72	56	0.016	0.341	Good
466	25	90	72	0.015	0.512	
318	25	90	56	0.004	0.191	

The same information can be used for determining the offerings in the agricultural sector. The list of considered designs is shown again in table 4.14, this time with the utility values appropriate for the agricultural epochs. Of note here is that in each of epoch 3 and epoch 4 the acceptable designs have the same utility values. Upon inspection the tradespaces for these epochs (see figures 4-13 and 4-15) one finds that utility in these tradespaces is a function of the overall engine power. Because the engine power is not varied in the considered designs and owing to the fact that a modularity path enabler for power was not in place for this example, there are is only one product available to users for each of epochs 3 and 4. The producing entity might now decide to make the cost/time trade-off decision to put an engine power modularity path enabler in place such that it could offer a wider selection of products to the producting entities target markets.

Now that four designs, $\{22, 170, 281, 318\}$, have been selected as offerings to two different markets, they can be tied together into an overall product line. Design ID 22 could be

Table 4.14: Product offerings for agriculture epochs

Design ID	Hydraulic /Engine Power (HP)	Vehicle Width (in)	Utility-Agriculture		Product Offering
			Epoch 3	Epoch 4	
22	25	72	< 0	0.809091	Yes
170	25	72	< 0	0.809091	
281	25	72	< 0	0.809091	
318	25	90	1	< 0	Yes
466	25	90	1	< 0	
577	25	90	1	< 0	

considered the base "platform" by the supplier; design 22 could then be transitioned to the other designs via modularity in production or use of either the frame or the attachment. The family offering of products, based on 22 as the base platform, is shown graphically in figure 4-23. Also shown in figure 4-23 is an indication of which variants would be offered to which use types: construction, agriculture, or both. In the case of designs 22, 170, and 281, if a construction customer acquired any of these three designs then they could transition to any of the other two designs via modularity in attachment during the use phase. In the case of designs 22 and 318, if an agricultural customer acquired either of these they could not transition to the other design because modularity in the frame is only available at the time of production; design 22 would be acquired for epoch 4 needs and design 318 would be acquired for epoch 3 needs.

4.11 Case Summary

This chapter has presented a case of a terrestrial vehicle being considered for two use scenarios (construction and agriculture). These two use scenarios were represented as four epochs (two epochs for each use scenario) and Multi-Attribute Tradespace Exploration, Dynamic MATE, and Multi-Epoch Analysis was completed to downselect from all possible designs to 46 potentially valuable designs. Analysis of the potentially valuable designs

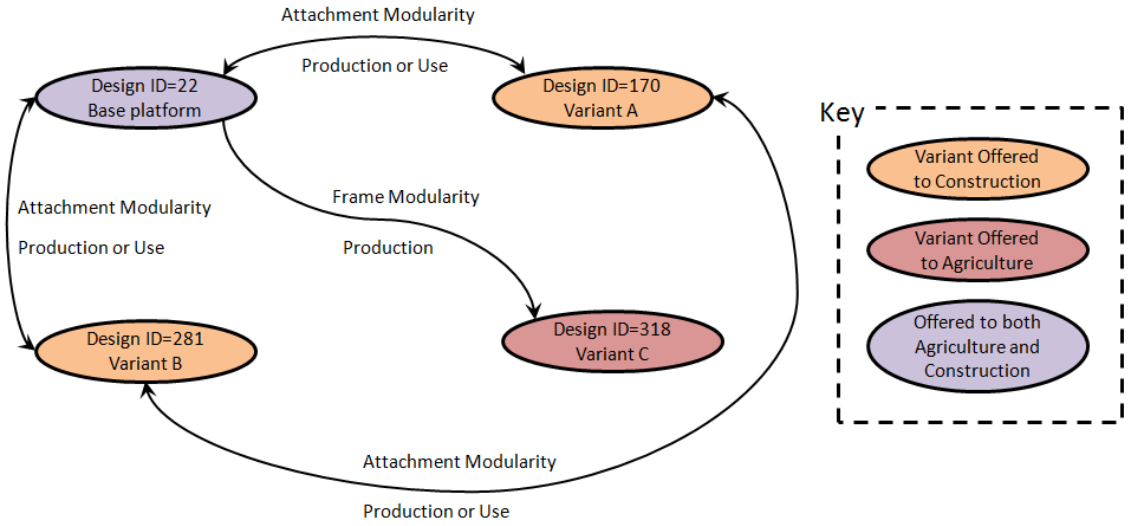


Figure 4-23: Plaform, variants, users, and available transitions for modular LifeTrac

was completed to determine the design variable variation across the selected designs; this variation was used as an indicator that there may need to be allowances to make changes in those design variables during the life of the system. A system architecture representation, using a design structure matrix, was generated of the target system. The design structure matrix was then used to generated modularity metrics for the components of the system that would need to change in response to a change in a design variable. The component modularity by design variable was analyzed to indicate to what extent those components supported the change via modularity. In addition to the case where an existing architecture existed that may need to be altered, there was a discussion of how this approach could be used for clean sheet design efforts. Also, the results of the tradespace study and modularity analysis were combined to demonstrated how a complete product line could be developed and offered to the market based on the insights gained from executing the value-centric approach to modularization.

Chapter 5

Discussion and Implications

This chapter will briefly review the benefits of the proposed design approach to focus modularization efforts, review how the research objectives were addressed by this thesis, cover the limitations of the proposed approach, suggest possible areas for further research, and close with final thoughts.

5.1 MATE output to target modularization

This thesis set out with a goal to provide designers value-centric, quantitative tools that could be used to focus precious design efforts with regard to embedding modularity. Decisions made early in the system design process can have long term impact on the value delivered by the system. This thesis proposed and demonstrated the use of MATE to focus design for changeability (DfC) (Fricke and Schulz, 2005) effort on the design variables that may vary due to different use scenarios and/or decision maker preference sets. These design variables were then linked to components in a system architecture that would need to change to accommodate a change in design variables. Using a DSM approach, the modularity of those components was measured using metrics based on network centrality measures of the connectivity of a component to other components. Comparison of the

quantitative modularity of the components to the desired changeability, based on potential variability design variables, was demonstrated. Overall, it is believed that this presents a new method to focus designer's modularization efforts on those portions of the system that are more likely to need to change based on what a decision maker values.

5.2 Research Objectives / Questions

In section 2.5 of this thesis, there were three questions that drove this research:

1. How could the MATE framework be used to inform modularization decisions in support of changing and/or differing stakeholder requirements?

2. What is the relationship between component modularity, system path enablers, and transition rules within the dynamic MATE framework?

3. How can the findings from above be used to inform existing approaches to modularization?

Through a case study, it was shown how the MATE framework could be used to indicate which design variables could change as a result of different use scenarios and preference sets and how to trace those changing design variables to the affected system components. This tracing was then combined with network centrality measures of component modularity to indicate to what degree those components supported modular changeability to achieve the different desired levels of design variables. The output from this can then be used to focus the modularity efforts of system designers.

While only qualitatively considered, a connection was made between component modularity and system path enablers and transition paths. If one is starting with an existing design, the modularity metrics could be used to indicate which modularity path enablers are present and to what degree, providing higher fidelity as compared to binary existence of modularity. Additionaly, these opportunities for targeted modularity can be used to

identify families of systems that are, together, value robust across multiple epochs. As an example of the benefits of the proposed approach, the results of the tradespace study and modularity analysis were combined to demonstrated how a complete product line could be developed and offered to the market based based on the results of the approach.

Finally, this thesis has utilized MATE to provide a value centric focus to modularity efforts. Instead of relying on experiential and tacit knowledge, this approach can focus designer's modularization efforts on the elements of the system that may need to change to accommodate changes in decision makers preferences or use contexts (i.e., epoch changes). Further, the proposed approach provides designers with tools to use to target and focus modularization efforts for new systems on which they may not be able to rely on experiential or tacit knowledge.

5.3 Limitations

This thesis was limited to a single case study; additional case studies need to be done to further validate the approach. The single case study considered changes to an existing architecture; while an approach was proposed for utilization on a clean sheet design, a clean sheet design example was not considered and would entail the challenge of constructing one or more component DSMs in order to calculate the modularity metrics. Full utility functions were not elicited from the decision maker and, in the agriculture case, a literature review was used in place of a human decision maker. This work only considered two modularity metrics to measure component design; sensitivity to these metrics and their use to indicate the presence of a modularity path enabler was not completed in this research. The case was limited to a single terrestrial vehicle and this makes no claim to other types of systems (e.g., information technology or aerospace systems). The system considered had a limited number of components and there was no testing as to the ability of the approach to scale to a system with much larger number of components.

5.4 Future Research

In addition to the research opportunities implied by the limitations, this research has raised other questions in the author's mind that could be the subject of future research:

- Application of the approach to additional cases would refine the approach with the goals of removing embedded assumptions and developing a repeatable, generic process.
- There is an opportunity to address the scenario where a form-form interaction representation of the system cannot be reasonably built.
- The opportunity to study the impact of this approach on a system or commercial design effort could provide additional insights as to how to help designers focus their modularization efforts.
- Research into the application of this approach to supplying massively customizable products to commercial consumers could provide insights into the ability of the process to deal with extreme levels of changeability.
- There is an opportunity to extend the use of MATE to commercial system products (e.g., agricultural equipment). This seems particularly interesting as this case would include market size number of decision makers, with many needs, use scenarios, preference sets, and changing context variables.
- The current approach requires a DSM representation of a system architecture in order to complete the analysis; a DSM representation will not always be available. Methods to evaluate modularity during early phases that do not require an DSM architectural representation are an area of possible research, with a Design for Modularity (DfM) approach, building upon the Design for Changeability (DfC) approach of (Fitzgerald et al., 2012a,b), being one potential way of dealing with the absence of an architectural representation.

- While doing this research, it was found that there is an opportunity to formalize the syntax and semantics of the Rule-Effects matrix representation.

5.5 Final Thoughts

Component modularity within a system can aid in the changeability of that system if necessary. Modularity seems to be a key enabling -ility for other system lifecycle properties, such as flexibility, evolvability, scalability, and others (de Weck et al., 2012). However, modularity often comes with drawbacks, such increased system development time, mass, and/or cost penalities (Whitney, 2003). Balancing these trade-offs is an important portion of a designer's job (Maier and Rechtin, 2009). Because of the importance modularity may play in overall systems lifecycle properties, research into when it is appropriate to leverage modularity is beneficial to the designers, funders, and users of engineering systems. It is believed by the author that the approach discussed in this thesis, which combines Multi-Attribute Tradespace Exploration, Dynamic MATE, Multi-Epoch Analysis and network centrality measures of component modularity, can aid the designer in focusing modularization efforts during early design phases so as to create a high amount of leverage for a system's users and producers.

Appendix A

LifeTrac Design Structure Matrix

This appendix contains a zoomed in view of the LifeTrac DSM from section 4.3.1. Physical dependencies are represented with a 'P', mass flow dependencies are represented with a 'M', energy flow dependencies are represented with an 'E', and information flow dependencies are represented with an 'I'.

Figure A-1: Component DSM of LifeTrac - Upper-left Quadrant

Figure A-2: Component DSM of LifeTrac - Upper-right Quadrant

Figure A-3: Component DSM of LifeTrac - Lower-left Quadrant

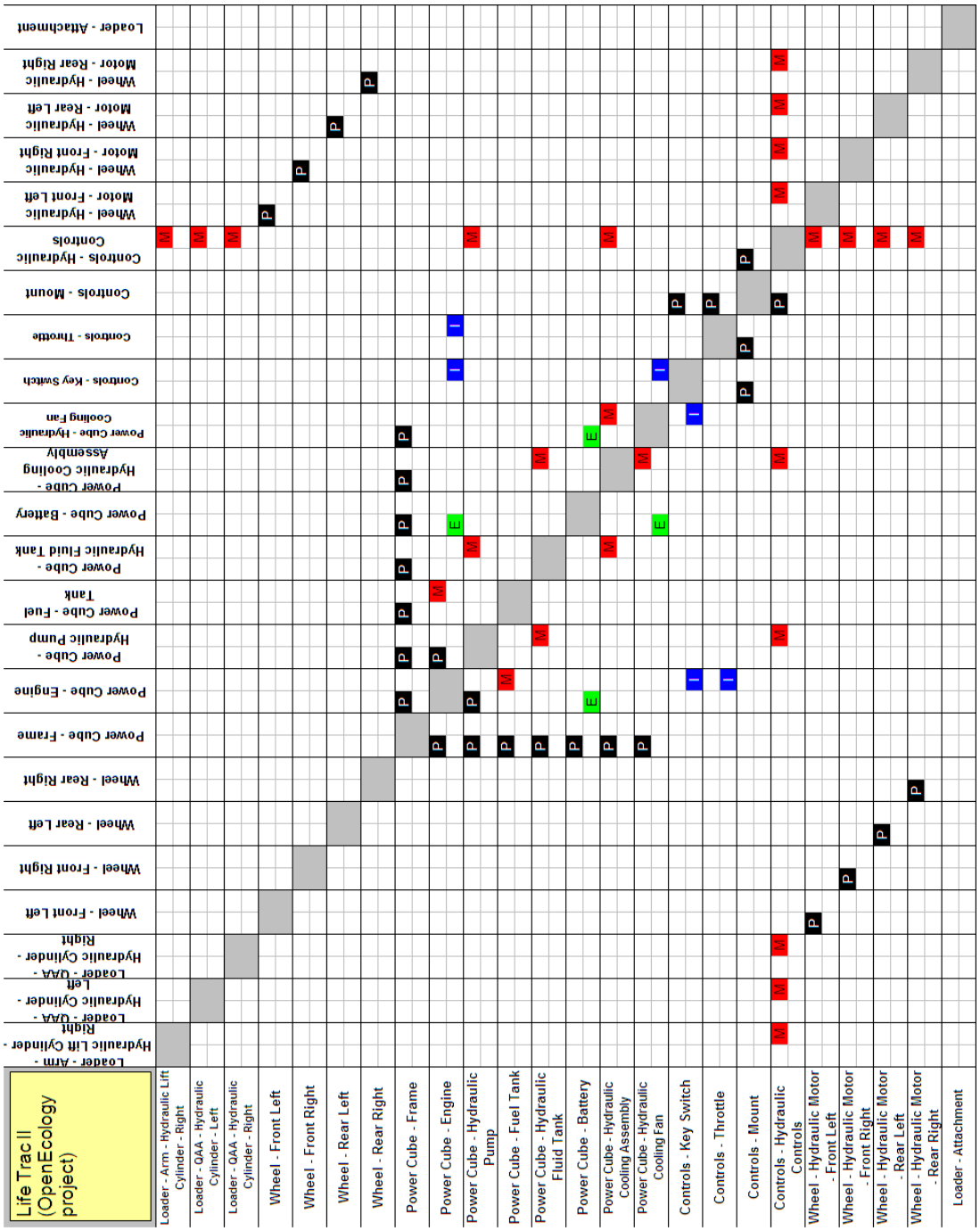


Figure A-4: Component DSM of LifeTrac - Lower-right Quadrant

Appendix B

LifeTrac Modularity Metrics

In table B.1 one will find a complete listing of the calculated network centrality and modularity metrics for the LifeTrac case. Initially Freeman degree, Freeman farness, and betweenness metrics were calculated using Analytic Technologies UCINet version 6 for Microsoft Windows (Borgatti et al., 2002), then these were converted to the component modularity metrics. Information on how the software calculates these metrics is available at `http://faculty.ucr.edu/~hanneman/nettext/C10_Centrality.html#Freeman`, `http://faculty.ucr.edu/~hanneman/nettext/C10_Centrality.html#Closeness`, and `http://faculty.ucr.edu/~hanneman/nettext/C10_Centrality.html#freebet`.

Table B.1: Centrality and modularity metrics for LifeTrac

i	DSM Entry	UCINET Degree	Degree Modularity $M(D)_i$	UCINET Farness	Distance Modularity $M(T)_i$	UCINET Betweenness	Bridge Bridge $M(B)_i$
1	Frame - Lower Section - Front piece	6	0.870	116	0.054	33.454	0.984
2	Frame - Lower Section - Mid piece	6	0.870	123	0.057	28.721	0.986
3	Frame - Lower Section - Back piece	6	0.870	116	0.054	33.454	0.984
4	Frame - Lower Section - Outer Right piece	6	0.870	122	0.056	25.713	0.988
5	Frame - Lower Section - Inner Right piece	9	0.804	106	0.049	75.771	0.963
6	Frame - Lower Section - Inner Left piece	9	0.804	106	0.049	75.771	0.963
7	Frame - Lower Section - Outer Left piece	6	0.870	122	0.056	25.713	0.988
8	Frame - Verticals - Front Right piece	5	0.891	120	0.056	22.762	0.989
9	Frame - Verticals - Mid Right piece	5	0.891	114	0.053	36.212	0.983
10	Frame - Verticals - Back Right piece	6	0.870	103	0.048	94.879	0.954
11	Frame - Verticals - Front Left piece	6	0.870	103	0.048	94.879	0.954
12	Frame - Verticals - Mid Left piece	5	0.891	114	0.053	36.212	0.983
13	Frame - Verticals - Back Left piece	5	0.891	120	0.056	22.762	0.989
14	Frame - Upper Crossbrace - Right piece	7	0.848	111	0.051	67.530	0.967
15	Frame - Upper Crossbrace - Left piece	7	0.848	111	0.051	67.530	0.967
16	Frame - Upper Section - Front piece	4	0.913	119	0.055	11.236	0.995
17	Frame - Upper Section - Mid piece	4	0.913	126	0.058	7.136	0.997
18	Frame - Upper Section - Back piece	4	0.913	119	0.055	11.236	0.995
19	Loader - Quick Attach Assembly	5	0.891	139	0.064	51.284	0.975
20	Loader - Arm - Left	5	0.891	127	0.059	58.586	0.972
21	Loader - Arm - Right	5	0.891	127	0.059	58.586	0.972
22	Loader - Arm - Support Beam	2	0.957	156	0.072	1.784	0.999
23	Loader - Arm - Hydraulic Lift Cylinder - Left	4	0.913	112	0.052	47.222	0.977
24	Loader - Arm - Hydraulic Lift Cylinder - Right	4	0.913	112	0.052	47.222	0.977
25	Loader - QAA - Hydraulic Cylinder - Left	3	0.935	123	0.057	26.675	0.987
26	Loader - QAA - Hydraulic Cylinder - Right	3	0.935	123	0.057	26.675	0.987
27	Wheel - Front Left	3	0.935	138	0.064	4.000	0.998
28	Wheel - Front Right	3	0.935	138	0.064	4.000	0.998
29	Wheel - Rear Left	3	0.935	138	0.064	4.000	0.998
30	Wheel - Rear Right	3	0.935	138	0.064	4.000	0.998
31	Power Cube - Frame	9	0.804	111	0.051	157.750	0.924
32	Power Cube - Engine	6	0.870	136	0.063	25.667	0.988
33	Power Cube - Hydraulic Pump	4	0.913	117	0.054	38.817	0.981
34	Power Cube - Fuel Tank	2	0.957	152	0.070	0.000	1.000
35	Power Cube - Hydraulic Fluid Tank	3	0.935	144	0.067	0.333	1.000
36	Power Cube - Battery	3	0.935	151	0.070	0.583	1.000
37	Power Cube - Hydraulic Cooling Assembly	4	0.913	118	0.055	35.650	0.983
38	Power Cube - Hydraulic Cooling Fan	4	0.913	139	0.064	7.983	0.996
39	Controls - Key Switch	3	0.935	142	0.066	9.867	0.995
40	Controls - Throttle	2	0.957	145	0.067	4.100	0.998
41	Controls - Mount	5	0.891	112	0.052	94.233	0.954
42	Controls - Hydraulic Controls	11	0.761	93	0.043	278.130	0.866
43	Wheel - Hydraulic Motor - Front Left	3	0.935	118	0.055	22.229	0.989
44	Wheel - Hydraulic Motor - Front Right	3	0.935	121	0.056	24.212	0.988
45	Wheel - Hydraulic Motor - Rear Left	3	0.935	121	0.056	24.212	0.988
46	Wheel - Hydraulic Motor - Rear Right	3	0.935	118	0.055	22.229	0.989
47	Loader - Attachment	1	0.978	184	0.085	0.000	1.000

Appendix C

LifeTrac Modularity Metrics for Selected Components

This appendix contains a convenient listing of the modularity metrics for the key components mapped in table 4.7. Each metric will be presented in turn in tables C.1, C.2, and C.3; these tables take the same form as the DMM developed earlier. Of note in these tables is along with the raw modularity metric, a rank of that component for that metric compared to all the other components ($n = 47$) is provided. This is because that while the absolute metrics provide some insight, the level of a components modularity relative to the other components in the system that can help provide insight at this early stage in the design process.

Table C.1: Degree modularity, $M(D)_i$ and $(Rank)$, of select LifeTrac components

		Design Variable			
		Skid Steer		Tractor	
i	Determined Component	Bucket Width	Engine Power	Vehicle Width	Engine Power
1	Frame - Lower Section - Front piece			0.870 (34)	
2	Frame - Lower Section - Mid piece			0.870 (34)	
3	Frame - Lower Section - Back piece			0.870 (34)	
20	Loader - Arm - Left		0.891 (26)		
21	Loader - Arm - Right		0.891 (26)		
23	Loader - Arm - Hydraulic Lift Cylinder - Left		0.913 (18)		
24	Loader - Arm - Hydraulic Lift Cylinder - Right		0.913 (18)		
25	Loader - QAA - Hydraulic Cylinder - Left		0.935 (5)		
26	Loader - QAA - Hydraulic Cylinder - Right		0.935 (5)		
32	Power Cube - Engine		0.870 (34)		0.870 (34)
33	Power Cube - Hydraulic Pump		0.913 (18)		0.913 (18)
42	Controls - Hydraulic Controls		0.761 (47)		0.761 (47)
43	Wheel - Hydraulic Motor - Front Left				0.935 (5)
44	Wheel - Hydraulic Motor - Front Right				0.935 (5)
45	Wheel - Hydraulic Motor - Rear Left				0.935 (5)
46	Wheel - Hydraulic Motor - Rear Right				0.935 (5)
47	Loader - Attachment	0.978 (1)			

Table C.2: Distance modularity, $M(T)_i$ and $(Rank)$, of select LifeTrac components

		Design Variable			
		Skid Steer		Tractor	
i	Determined Component	Bucket Width	Engine Power	Vehicle Width	Engine Power
1	Frame - Lower Section - Front piece			0.054 (33)	
2	Frame - Lower Section - Mid piece			0.057 (18)	
3	Frame - Lower Section - Back piece			0.054 (33)	
20	Loader - Arm - Left		0.059 (15)		
21	Loader - Arm - Right		0.059 (15)		
23	Loader - Arm - Hydraulic Lift Cylinder - Left		0.052 (37)		
24	Loader - Arm - Hydraulic Lift Cylinder - Right		0.052 (37)		
25	Loader - QAA - Hydraulic Cylinder - Left		0.057 (18)		
26	Loader - QAA - Hydraulic Cylinder - Right		0.057 (18)		
32	Power Cube - Engine		0.063 (14)		0.063 (14)
33	Power Cube - Hydraulic Pump		0.054 (32)		0.054 (32)
42	Controls - Hydraulic Controls		0.043 (47)		0.043 (47)
43	Wheel - Hydraulic Motor - Front Left				0.055 (29)
44	Wheel - Hydraulic Motor - Front Right				0.056 (23)
45	Wheel - Hydraulic Motor - Rear Left				0.056 (23)
46	Wheel - Hydraulic Motor - Rear Right				0.055 (29)
47	Loader - Attachment	0.085 (1)			

Table C.3: Bridge modularity, $M(B)_i$ and $(Rank)$, of select LifeTrac components

		Design Variable			
		Skid Steer		Tractor	
i	Determined Component	Bucket Width	Engine Power	Vehicle Width	Engine Power
1	Frame - Lower Section - Front piece			0.984 (28)	
2	Frame - Lower Section - Mid piece			0.986 (27)	
3	Frame - Lower Section - Back piece			0.984 (28)	
20	Loader - Arm - Left		0.972 (37)		
21	Loader - Arm - Right		0.972 (37)		
23	Loader - Arm - Hydraulic Lift Cylinder - Left		0.977 (34)		
24	Loader - Arm - Hydraulic Lift Cylinder - Right		0.977 (34)		
25	Loader - QAA - Hydraulic Cylinder - Left		0.987 (25)		
26	Loader - QAA - Hydraulic Cylinder - Right		0.987 (25)		
32	Power Cube - Engine		0.988 (22)		0.988 (22)
33	Power Cube - Hydraulic Pump		0.981 (33)		0.981 (33)
42	Controls - Hydraulic Controls		0.866 (47)		0.866 (47)
43	Wheel - Hydraulic Motor - Front Left				0.989 (16)
44	Wheel - Hydraulic Motor - Front Right				0.988 (20)
45	Wheel - Hydraulic Motor - Rear Left				0.988 (20)
46	Wheel - Hydraulic Motor - Rear Right				0.989 (16)
47	Loader - Attachment	1.000 (1)			

Appendix D

LifeTrac System Model

This appendix contains the LifeTrac system model; system models are used in MATE to map from the DV space to a system performance space. Because the focus of this case was on connecting MATE and component modularity, these models should not be considered of sufficient quality as to be used in decision making.

The lifecycle cost model was approximated as

$$L_d(H) = C_d + R(E, H)$$

where $L_d(H)$ is the lifecycle cost of a particular design as a function of the number of hours operated H, C_d is the acquisition cost of a particular design, and $R(E, H)$ is cost to operate as a function of engine power E and hours operated over the lifecycle H. For the LifeTrac $R(E, H)$ was approximated as 1 United States Dollar (USD) per horsepower-hour; and the lifecycle was set at 1500 hours for all designs.

The acquisition cost model is

$$C_d = N(E) + F(W)$$

where C_d is the cost of a single design, $N(E)$ is the cost of the engine in as a function of

E power, and $F(W)$ is the cost of the remainder of the machine as a function of W the width of the machine.

The engine cost was modeled as

$$N(E) = 59.69 \cdot E - 19.70$$

where 59.69 is USD per horsepower (USD/hp), E is the power of the engine in horsepower, and 19.70 in USD. The constants 59.69 and 19.70 were determine by linear regression, using Microsoft Excel 2010's charting trendline function, to the data in table D.1; the linear regression has $R^2 = 0.95$. This set of engines were used because they were of the same type (air cooled, gasoline, vertical shaft) as the LifeTrac design. These engine prices were gathered on 11-November-2011 from `http://smallenginesuppliers.com/shop/html/pages/briggs_vertical_shaft_engines.html`. According to LifeTrac website, `http://opensourceecology.org/wiki/LifeTrac/Research_Development`, on 20 November 2011 the cost of the power cube assembly is approximately $USD1750$; the model, using the stated power of the power cube of $27HP$ predicts the cost of the engine as $USD1589.23$. Given that there are additional components to the power cube assembly (e.g., frame, cooling, hydraulic pump) this was considered of sufficient fidelity and precision for this case.

The frame cost was modeled as

$$F(W) = 6000 + W \cdot 41.67$$

where 6000 is the base cost of the product with the exception of the engine component and frame elements that contribute to the width, W is the width in inches, and 41.67 is in units of $USD/inch$.

The skid steer function performance was determine empirically from existing available commercial designs; these performance findings are in table D.2, collected from `http://www.speccheck.com/` on 20-November-2011.

Table D.1: Retail Engine Prices

Engine	HP	Price (USD)
Briggs & Stratton 10A902	4	241
Briggs & Stratton 126L02	5.75	311
Briggs & Stratton 217902	10.5	751
Briggs & Stratton 215802	10.5	612
Briggs & Stratton 215907	10.5	716
Briggs & Stratton 217807	11.5	706
Briggs & Stratton 219807	12.5	703
Briggs & Stratton IC219972	12.5	581
Briggs & Stratton 219907	12.5	792
Briggs & Stratton 219907	12.5	812
Briggs & Stratton 219807	12.5	792
Briggs & Stratton 21B977	13.5	581
Briggs & Stratton 21B807	13.5	792
Briggs & Stratton 31C707	17	988
Briggs & Stratton 31C707	17.5	792
Briggs & Stratton 31C707	17.5	973
Briggs & Stratton 31P777	18.5	1047
Briggs & Stratton 31P677	19.5	1047
Briggs & Stratton 40G777	20	1412
Briggs & Stratton 386777	23	1421
Briggs & Stratton 49M777	28	1645
Briggs & Stratton 49M977	30	1610
Kohler CV1000	40	2500

Table D.2: Skid Steer Specifications

Model	Engine Power (kW)	Bucket Breakout Force (kg)	Tipping Load (kg)
Deere 313	33.5	1498	1146
Deere 315	33.5	1498	1362
Deere 318D	43.3	2327	1634
Deere 320D	47.1	2327	1816
Deere 326D	51.9	3519	2361
Deere 328D	59.6	4676	2542
Deere 332D	66.7	6299	2906
Bobcat S100	26.5	N/A	908
Bobcat S130	34.3	N/A	1195
New Holland L150	26.8	1591	1225
Thomas 105	22.5	795	907

From these, the performance was modeled as

$$B(E) = 113.39 \cdot E - 2123.8$$

and

$$T(E) = 48.03 \cdot E - 323.13$$

where $B(E)$ is the bucket breakout force in kilograms (kg) as a function of engine power E in kilowatts (kW), and $T(E)$ is the tipping load in kg. These functions were determined by using Microsoft Excel 2010's curve fitting function. The function $B(E)$ has $R^2 = 0.90$ and $T(E)$ has $R^2 = 0.96$.

For the tractor function performance was determined using the models for drawbar horsepower for tillage tools from `http://www.soil.ncsu.edu/open_furrow/Documents/DHC/Soil,Draft,andTraction.pdf`, accessed on 11-November-2011. This was modeled as an optimization of the acres covered per hour as a function of the speed through the field and the number of moldboards for a given engine power. These were calculated using Microsoft Excel 2010.

Bibliography

Allen, Tom, Mcgowan, Don, Moses, Joel, Magee, Christopher, Hastings, Daniel E., Moavenzadeh, Fred, Nightingale, Debbie, Little, John, and Roos, Dan (2002). "ESD Terms and Definitions". URL `http://esd.mit.edu/WPS/esd-wp-2002-01.pdf`.

Baldwin, Carliss Y. and Clark, Kim B. (2000). *Design Rules, Vol 1: The Power of Modularity*. MIT Press, Cambridge, MA.

Bartolomei, Jason E (2007). *Qualitative Knowledge Construction for Engineering Systems : Extending the Design Structure Matrix Methodology in Scope and Procedure by Doctor of Philosophy in Engineering Systems*. Ph.D. thesis.

Borgatti, S. P., Everett, M. G., and Freeman, L. C. (2002). "Ucinet for Windows : Software for Social Network Analysis".

Browning, Tyson R (2001). "Applying the design structure matrix to system decomposition and integration problems: a review and new directions". *Engineering Management, IEEE Transactions*, 48(3):292–306. URL `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=946528`.

Browning, Tyson R (2011). "The Design Structure Matrix : Introduction and Applications (Tutorial Presentation)".

Browning, Tyson R and Eppinger, Steven D. (2002). "Modeling impacts of process architecture on cost and schedule risk in product development". *Engineering Management, IEEE Transactions*, 49(4):428–442. URL `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1176870`.

Crawley, Edward (2007). "System Architecture". URL `http://ocw.mit.edu/courses/engineering-systems-division/esd-34-system-architecture-january-iap-2007/`.

Danilovic, Mike and Browning, Tyson R (2007). "Managing complex product development projects with design structure matrices and domain mapping matrices". *International*

Journal of Project Management, 25:300–314. doi:10.1016/j.ijproman.2006.11.003. URL http://www.sciencedirect.com/science/article/pii/S0263786306001645.

de Weck, Olivier, Roos, Dan, and Magee, Christopher (2011). *Engineering Systems : Meeting Human Needs in a Complex Technological World*. MIT Press.

de Weck, Olivier, Ross, Adam M, and Rhodes, Donna H (2012). "Investigating Relationships and Semantic Sets amongst System Lifecycle Properties (Ilities)". URL http://esd.mit.edu/WPS/2012/esd-wp-2012-12.pdf.

Derleth, Jason Edward (2003). "Multi-Attribute Tradespace Exploration and its Application to Evolutionary Acquisition". URL http://seari.mit.edu/documents/theses/SM_DERLETH.pdf.

Engler, Candice (2011). "Personal communication, 13-October-2011".

Fitzgerald, Matthew E, Ross, Adam M, and Rhodes, Donna H (2012a). "Mitigating Contextual Uncertainties with Valuable Changeability Analysis in the Multi-Epoch Domain". In "6th Annual IEEE Systems Conference", .

Fitzgerald, Matthew E, Ross, Adam M, and Rhodes, Donna H (2012b). "Sustaining Lifecycle Value: Valuable Changeability Analysis with Era Simulation". In "6th Annual IEEE Systems Conference", .

Fricke, Ernst and Schulz, Armin P. (2005). "Design for changeability (DfC): Principles to enable changes in systems throughout their entire lifecycle". *Systems Engineering*, 8(4):342–359. ISSN 1098-1241. doi:10.1002/sys.20039. URL http://doi.wiley.com/10.1002/sys.20039.

Goering, Carroll E. and Hansen, Allen C. (2004). *Engine And Tractor Power*. Amer Society of Agricultural, 4th edition.

Hauser, John R and Clausing, Don (1988). "The House of Quality". *Harvard business review*, 66(3):63–73.

Hölttä-Otto, Katja (2005). *Modular product platform design*. Ph.D. thesis. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.98.739&rep=rep1&type=pdf.

Hölttä-Otto, Katja and de Weck, Olivier (2007). "Degree of Modularity in Engineering Systems and Products with Technical and Business Constraints". *Concurrent Engineering*, 15(2):113–126. ISSN 1063-293X. doi:10.1177/1063293X07078931. URL http://cer.sagepub.com/cgi/doi/10.1177/1063293X07078931.

Johnson, H. Thomas and Broms, Anders (2000). *Profit Beyond Measure: Extraordinary Results through Attention to Work and People*.

Keeney, R.L. and Raiffa, Howard (1993). *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. Cambridge University Press, Cambridge, England.

Lindemann, Udo, Maurer, Mike, and Braun, Thomas (2009). *Structural Complexity Management*. ISBN 9783540878889.

Maier, Mark W and Rechtin, Eberhardt (2009). *The Art of Systems Architecting*. CRC Press, 3rd edition.

McManus, H.L., Haggerty, Al, and Murman, Earll (2005). "Lean Engineering : Doing the Right Thing Right". In "1st International Conference on Innovation and Integration in Aerospace Sciences", .

McManus, H.L. and Hastings, Daniel E. (2006). "A framework for understanding uncertainty and its mitigation and exploitation in complex systems". *IEEE Engineering Management Review*, 34(3):81–94. ISSN 0360-8581. URL `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.132.912&rep=rep1&type=pdf`.

Meyer, Marc H. and Lehnerd, Alvin P. (1997). *The Power of Product Platforms*. The Free Press, New York.

Open-Ecology-Project (2011a). "LifeTrac". URL `http://opensourceecology.org/wiki/LifeTrac`.

Open-Ecology-Project (2011b). "LifeTrac Bill of Materials". URL `http://opensourceecology.org/wiki/LifeTrac/Bill_of_Materials`.

Open-Ecology-Project (2011c). "LifeTrac III (SolidWorks parts & assembly)". URL `http://openpario.mime.oregonstate.edu/attachments/download/4304/LifeTrac_III__SolidWorks_parts___assembly_.zip`.

Open-Ecology-Project (2011d). "LifeTrac Image". URL `http://opensourceecology.org/wiki/File:LifeTrac.png`.

Open-Ecology-Project (2011e). "Open Source Ecology License". URL `http://opensourceecology.org/wiki/License`.

Ross, Adam M (2003). "Multi-attribute tradespace exploration with concurrent design as a value-centric framework for space system architecture and design". URL `http://seari.mit.edu/documents/theses/SM_ROSS.pdf`.

Ross, Adam M (2006). *Managing unarticulated value: changeability in multi-attribute tradespace exploration*. Ph.D. thesis. URL `http://dspace.mit.edu/handle/1721.1/35089`.

Ross, Adam M (2008). "Defining and Using the New "ilities"". URL http://seari.mit.edu/documents/working_papers/SEAri_WP-2008-4-1.pdf.

Ross, Adam M, Hastings, Daniel E., Warmkessel, Joyce M, and Diller, Nathan P (2004). "Multi-Attribute Tradespace Exploration as Front End for Effective Space System Design". *Journal of Spacecraft and Rockets*, 41(1):20–28.

Ross, Adam M, McManus, H.L., Rhodes, Donna H, Hastings, Daniel E., and Long, A. (2009a). "Responsive Systems Comparison Method: Dynamic Insights into Designing a Satellite Radar System". In "AIAA Space", American Institute of Aeronautics and Astronautics, 1801 Alexander Bell Dr., Suite 500 Reston VA 20191-4344 USA,.

Ross, Adam M and Rhodes, Donna H (2011). "Anatomy of a Change Option : Mechanisms and Enablers". URL http://seari.mit.edu/documents/working_papers/SEAri_WP-2011-1-2.pdf.

Ross, Adam M, Rhodes, Donna H, and Hastings, Daniel E. (2009b). "Using Pareto Trace to determine system passive value robustness". In "2009 3rd Annual IEEE Systems Conference", pages 285–290. Ieee. ISBN 978-1-4244-3462-6. doi:10.1109/SYSTEMS.2009.4815813. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4815813.

Shah, Nirav Bharat (2004). "Modularity as an Enabler for Evolutionary Acquisition". URL http://seari.mit.edu/documents/theses/SM_SHAH.pdf.

SmallEngineSuppliers.com (2011). "Briggs and Stratton Vertical Engines Retail Prices". URL http://smallenginesuppliers.com/shop/html/pages/briggs_vertical_shaft_engines.html.

Sosa, Manuel E., Eppinger, Steven D., and Rowles, Craig M. (2007). "A Network Approach to Define Modularity of Components in Complex Products". *Journal of Mechanical Design*, 129(11). ISSN 10500472. doi:10.1115/1.2771182. URL http://link.aip.org/link/JMDEDB/v129/i11/p1118/s1&Agg=doi.

Stone, Robert B., Wood, Kristin L., and Crawford, Richard H. (2000). "A heuristic method for identifying modules for product architectures". *Design Studies*, 21(1):5–31. ISSN 0142694X. doi:10.1016/S0142-694X(99)00003-4. URL http://linkinghub.elsevier.com/retrieve/pii/S0142694X99000034.

Ulrich, Karl (1995). "The role of product architecture in the manufacturing firm". *Research Policy*, 24:419–440.

Whitney, Daniel (2003). "Physical limits to modularity". URL http://esd.mit.edu/WPS/internal-symposium/esd-wp-2003-01.03.pdf.

Yu, Tian-Li, Yassine, Ali A., and Goldberg, David E. (2007). "An information theoretic method for developing modular architectures using genetic algorithms". *Research in Engineering Design*, 18:91–109. doi:10.1007/s00163-007-0030-1. URL `http://www.springerlink.com/index/E167818426401301.pdf`.