Defining Changeability: Reconciling Flexibility, Adaptability, Scalability, Modifiability, and Robustness for Maintaining System Lifecycle Value

Adam M. Ross[1], Donna H. Rhodes[2], and Daniel E. Hastings[3]

Massachusetts Institute of Technology
Building NE20-388, 77 Massachusetts Avenue
Cambridge, MA 02139
adamross@mit.edu

Abstract

Designing and maintaining systems in a dynamic contemporary environment requires a rethinking of how systems provide value to stakeholders over time. Developing either changeable or classically robust systems are approaches to promoting value sustainment. But, ambiguity in definitions across system domains has resulted in an inability to specify, design, and verify to ilities that promote value sustainment. In order to develop domain-neutral constructs for improved system design, the definitions of flexibility, adaptability, scalability, modifiability, and robustness are shown to relate to the core concept of "changeability," described by three aspects: change agents, change effects, and change mechanisms. In terms of system form or function parameter changes, *flexibility* and *adaptability* reflect the location of the change agent—system boundary external or internal respectively. *Scalability*, *modifiability*, and *robustness* relate to change effects, which are quantified differences in system parameters before and after a change has occurred. The extent of changeability is determined using a tradespace network formulation, counting the number of possible and decision maker acceptable change mechanisms available to a system, quantified as the *filtered outdegree*. Designing changeable systems allows for the possibility of maintaining value delivery over a system lifecycle, in spite of changes in contexts, thereby achieving *value robustness*.

1   Introduction

Dynamics of the contemporary environment are driving the need to rethink how systems can provide value to stakeholders throughout the system lifecycle. Addressing this need motivated the June 2004 US Air Force/MIT Lean Aerospace Initiative (LAI) Workshop on Systems Engineering for "Robustness," challenging the aerospace community to develop approaches that

---

[1] Research Scientist, Massachusetts Institute of Technology, NE20-388, 77 Massachusetts Avenue, Cambridge, MA 02139, Tel: 617-253-7061, Fax: 617-253-3641, Email: adamross@mit.edu
[2] Principal Research Scientist and Senior Lecturer, Massachusetts Institute of Technology, NE20-388, 77 Massachusetts Avenue, Cambridge, MA 02139, Tel: 617-324-0473, Fax: 617-253-3641, Email: rhodes@mit.edu
[3] Professor of Engineering Systems and Aeronautics and Astronautics and Dean for Undergraduate Education, MIT, Email: hastings@mit.edu

enable systems engineering to develop valuable systems that leverage future changes in missions and technology [Rhodes, 2004]. "Robustness" according to Dr. Marvin Sambur, Assistant Secretary of the Air Force for Acquisition at the time of the workshop, means:

- Capable of adapting to changes in mission and requirements;
- Expandable/scalable, and designed to accommodate growth in capability;
- Able to reliably function given changes in threats and environment;
- Effectively/affordably sustainable over their lifecycle;
- Developed using products designed for use in various platforms/systems; and
- Easily modified to leverage new technologies.

The list-definition of "robustness" reflects aspects of separately defined ilities of flexibility, scalability, robustness, sustainability, reusability, and upgradeability. Experts at the workshop admitted no comprehensive approach existed for designing for "robustness" in this sense and that further research was required in order to adequately address the Air Force need, especially to further understanding of possible tradeoffs between and among Sambur's list of ilities.

One key problem raised at the workshop was that the customer often wants a system to have ilities, but is not willing (or does not know how) to pay for them. Additionally, even though customers often request these types of ilities when acquiring systems, it was unclear how to specify, evaluate, and validate these ilities requirements for systems. A framework that allows for consideration of ilities during conceptual design, including concrete specification of how the ilities are defined, can be quantified and relate to perceived value, would benefit both military and industry, adding clarity in understanding the value proposition for "robustness" as described by Sambur.

The desire for "robustness" stems from the fact that change is inevitable, both in reality and perception. Developing robust systems to deal with real-world changes has been advanced through such approaches as Axiomatic Design [Suh, 2001] and Taguchi Robust Design [Park, 1996] methods. The goal of system design is not robust systems per se, but rather the delivery of value to system stakeholders. Dealing with value mismatch, the difference between system offerings and stakeholder expectations, including dynamic changes, has not been dealt with in the same manner as classical notions of robustness. It is inevitable that decision makers change their minds [Fricke et al., 2000], and therefore their perception of value of a system. In order to maintain value delivery over a system lifecycle, either the system must change, or at least the perception of the system must change, in order to match new decision maker expectations.

The motivation for changeability over a system lifecycle is categorized into three major drivers according to [Fricke and Schulz, 2005]: 1) dynamic marketplace, 2) technological evolution, and 3) variety of environments. These drivers result in two key aspects for system architectures to address: 1) they must be able to be changed easily and rapidly, and 2) they must be insensitive or adaptable towards changing environments [Schulz and Fricke, 1999; Schulz et al., 2000]. Such notions are often referenced in discussions of "flexibility" and "adaptability" and are often found in the manufacturing literature [Giachetti et al., 2003]. In the product design literature, the pursuit of "flexibility" is cited as an important characteristic for companies who design products for rapidly changing technologies and constant pressure to frequently upgrade their products [King and Sivaloganathan, 1998; Rajan et al., 2005]. Additionally others have

tried to develop empirical measures of flexibility [Chen and Yuan, 1998], adaptability [Li et al., 2007], and robustness [Hwang and Park, 2005] in order to assist in this endeavor. Recent work has tried to synthesize these definitions into a prescriptive six-element framework in the space system domain [Nilchiani, 2005; Nilchiani and Hastings, 2007].

Efforts to develop definitions and design principles for flexibility [Rajan et al., 2005; Qureshi et al., 2006; Keese et al., 2007], flexibility, adaptability, agility, and robustness as aspects of changeability [Schulz et al., 2000], and agility as a meta-level ability to change [Dove, 2005] principally stem from empirical and experiential insights in product and system design and development. The strength of these definitions is their grounding in empirical data; however, a weakness in empirically derived definitions is their inherent contextual bias. When confronted with new contexts or applications, the applicability of the definitions and principles may fall short. Instead of focusing on individual "ilities," [Ross and Hastings, 2006] introduce a set of theoretically-based definitions derived from both the technical literature and from common figurative usage, demonstrating their application in several aerospace case studies. A truly useful set of definitions should be rigorous, empirically grounded, and free from contextual biases.

2   Changeability Defined

As changeability grows in importance as a consideration in the engineering of systems, there is a critical need to have a more rigorous and quantified definition. In this section of the paper an enhanced definition of system changeability is described, which characterizes the elements of changeability. Change can be defined as the transition over time of a system to an altered state. If a system remains the same at time i and time i+1, then it has not changed. The inevitability of the effects of time on systems and environments results in a constant stream of change, both of the system itself and of its environment.

A change event can be characterized with three elements: (1) the agent of change, (2) the mechanism of change, and (3) the effect of change. Figure 1 illustrates these three elements. The *agent of change* is the instigator, or force, for the change. The role of change agent can be intentional or implied, but always requires the ability to set a change in motion. The *mechanism of change* describes the path taken in order to reach state 2 from state 1, including any costs, both time and money, incurred. The *effect of change* is the actual difference between the origin and destination states. A system that is black in time period one and gray in time period two has had its color changed. The change agent could be Nature, which can impart physical erosion due to wind, water, or sun, or could be a person with paint can and brush. The change mechanism could be the erosion or painting process, costing no money, but taking a long time, or costing some amount of money, but taking a shorter amount of time. Figure 1 shows the three aspects of change that must be defined to specify a change.
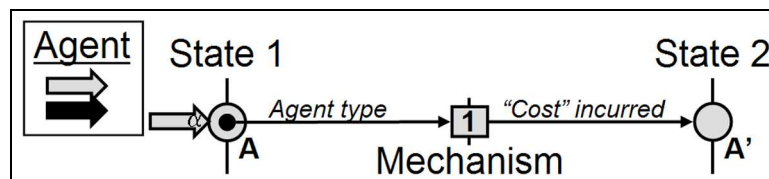

Figure 1. Change defined as state transition. Change specifications must include descriptions of change effect (difference between state 1 and state 2), change mechanism including "cost," and the change agent instigating the change.

The change described in Figure 1 is a simple case where there is only one particular change. In the *agent-mechanism-effect representation*, a particular change is represented by a *path*. The changeability of a system is determined by how easily it can undergo various changes. Figure 2 shows an example of an expanded view with multiple change paths enumerated.
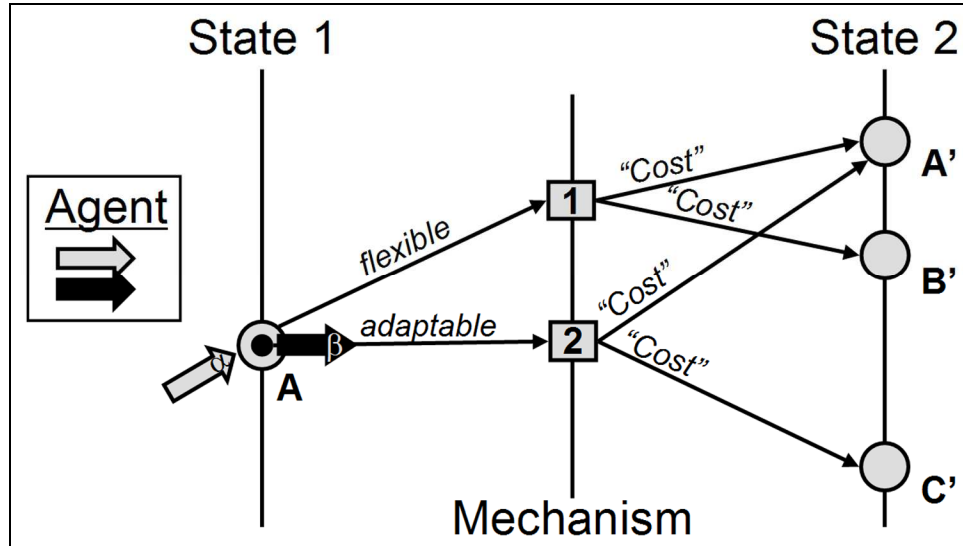

Figure 2. Multiple system changes depicted using the agents, mechanisms, and effects representation.

Table I enumerates the example agents, mechanisms, and effects shown in Figure 2. For a particular system, many agents, mechanisms, and end states may be possible.

Table I. Agents, mechanisms, effects, and paths shown in Figure 2.

Element	Description	As Illustrated in Figure 2
Change Agent	*The force instigator for the change to occur, for example humans, software, Mother Nature, etc.*	α, β
Change Mechanism	*The particular path the system must take in order to transition from its prior to its post state, including conditions, resources, and constraints*	1, 2
Change Effect	*The difference in states before and after a change has taken place.*	A'-A, B'-A, C'-A
Potential Paths	*The potential paths for the system to change from one state to another.*	α:A-1-A', α:A-1-B' β:A-2-A', β:A-2-C'

2.1 Change Agents.

One of the three elements of a change is the *change agent*. As defined above, the agent is the force instigator for the change to occur. Examples of change agents include humans, software, Mother Nature, etc. Intent is not required for a change agent, but the ability to set in motion a change mechanism is required. When characterizing an agent it is useful to think in terms of the steps required to put into motion a particular action.

Working backwards from most to least necessary, three capabilities can be used to differentiate the most to least sophisticated of change agents: 1) impact, 2) decision making, and

3) observation. Impact is the actual ability to set into motion a change mechanism. A physical impetus, such as a bat hitting a ball, or a social impetus, such as a manager tasking an employee, are examples of impact. Decision-making is the ability to process information in a structured manner in order to determine a course of action, presumably regarding whether to exert impact. Having decision-making ability allows for the incorporation of intent into actions. Observation is the ability to gather relevant information in order to conduct effective decision-making, and can increase the likelihood of making good decisions. It is important to realize that change can occur with only impact present, but intentional change usually occurs with decision-making, and the most effective intentional change usually occurs with all three abilities present. Adding "intelligence" to a change agent is equivalent to adding the decision-making and possibly observation abilities. The three abilities are loosely based on the analysis in [Hall, 1985], which differentiates between human and machine capabilities Psychomotor (Impact), Intellectual (Decision making), and Sensory/Perceptual (Observation).

The location of a change agent is a useful taxonomic distinction to be made when classifying change. If the change agent is external to the system, then the change under consideration is a *flexible*-type change. If the change agent is internal to the system, then the change under consideration is an *adaptable*-type change. Note that depending on the particular change being considered, a single system can be both flexible and adaptable. The definition of the system boundary must be explicitly defined in order to remove ambiguity when discussing whether a change should be considered as flexible or adaptable. Figure 3 summarizes these two change classification types.
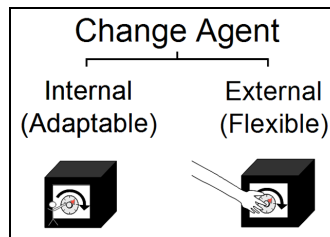

Figure 3. Change agent location (internal or external) for distinguishing between Adaptability and Flexibility.

2.2 Change Effects.

Another element of change to define is *change effect*. The change effect is the difference in states before and after a change has taken place, and often it is the effect that is first noticed to indicate a change has occurred. The changeability of a system can be classified into three categories of effects: *robustness*, *scalability*, and *modifiability*, which are proposed as a set from a parameter-based description of a system.

A system can be described in terms of sets of parameters, which capture physical, functional, and other performance aspects. *Robustness* is the ability to remain "constant" in parameters in spite of system internal and external changes. *Scalability* is the ability to change the level of a parameter. *Modifiability* is the ability to change the membership of the parameter set. The intent of the definitions is to balance generalizeability with precursor definitions. The definition of scalability thus presented would encompass concepts such as scaling up the number of satellites in a constellation (example parameter=number of satellites), the size of car chassis (example parameter=diameter or mass), and the number of users for the system (example

parameter=required data bandwidth).

As an example, consider the following parameter set for a car, which includes both function and form: {number of wheels, color of vehicle, quietness of cabin}. Suppose a design under consideration has the following particular parameter values: {4, "red," "moderately quiet"}. The possible ranges for these parameters, from lowest to highest, include: {[3, 4, 6, 8]; ["black", "red", "blue"]; ["very quiet", "moderately quiet", "little quiet", "not quiet"]}. If the current system can maintain its {4, "red," "moderately quiet"} in spite of its operating environment changing, such as due to driving on unpaved roads and past construction sites, then it is robust in these parameters to those particular environments. The more environments to which it is insensitive, the more robust the system.

If the system changes from having 4 wheels to having 6 wheels, then it has made a scalable change, and likewise if it changes from being "moderately quiet" to being "very quiet" or from being "red" to being "blue." If the system parameterization set is changed to be {number of wheels, color of vehicle, quietness of cabin, fraction cabin open} then it has undergone a modifiable change, enabling it to display a new function and/or form. In this example, the system is now designed to consider the amount of openness in the cabin. Figure 4 summarizes the three change effect types: robust, scaleable, and modifiable.
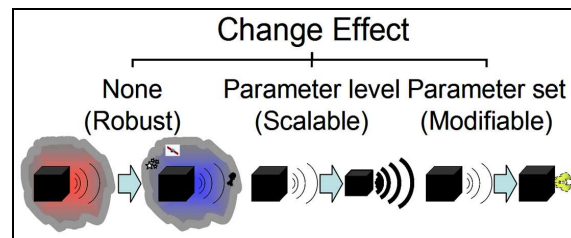

Figure 4. Change effect for distinguishing between Robustness, Scalability, and Modifiability.

Concepts such as modularity and integrality have been intentionally excluded from discussion on effects. Architectural concepts such as these are means to achieving ends (the effects) and should be considered separately. It is anticipated that modularity and other architectural approaches (networks, hierarchies, etc.) have costs and benefits, application dependent, and may or may not impact the change effects on a system. The next section, mechanisms, will discuss the "cost" for change as an important consideration for determining whether a system is actually changeable. Due to the context-dependent costs and benefits for particular architectural choices, these choices do not always increase or decrease the robustness, scalability, or modifiability of a system in straightforward manner. For example, modularity may increase one aspect of scalability of a system, due to lower costs for adding more components, but it may decrease other aspects of scalability of the system due to increased development costs for setting up the modularity in the first place.

2.3 Change Mechanisms.

Another element of change to define is *change mechanism*. The mechanism is the path the system must take in order to transition from its prior to its post state. A change path details the necessary components to bring about the change, including conditions, resources, and constraints for the change.

As an analogy, consider the problem of bringing a basketball down court from one basket

to the other by a player. (This simple example could represent the problem of relocating a satellite, for instance.) The change agent is the player, the system is the ball, and the change effect is on the location of the ball. Many possible paths can be taken by the player, each incurring a "cost" in terms of time and effort to change the location state of the ball. Depending on the circumstances (e.g., location of opposing and supporting teams, time of the game, penalty status of the player) the player may choose different paths in order to bring about the same change. More generally, systems likewise have many potential change mechanisms for bringing about different change effects. The more change paths available for a system to follow, the more changeable the system. Just as the "cost" for following a path played a role in determining the best course of action for the basketball player, so too does the potential path cost for a system determine its apparent changeability for a given decision maker. Over time the cost of a given path may change, especially as the context changes. The existence of paths may be objectively determined, but assessing whether a path at a point in time is reasonable to follow is subjectively decided by individuals based on their preferences and circumstances, including available resources. Figure 5 gives a notional example multiple paths with differing costs connecting the same prior and post states for change.
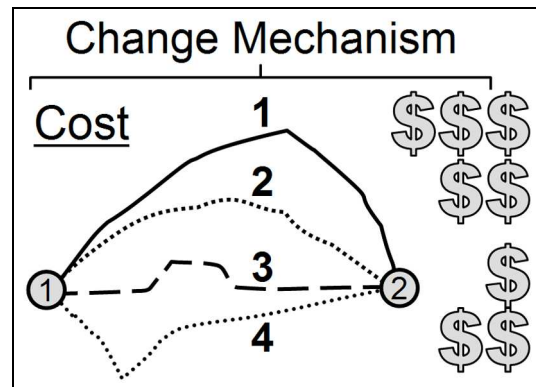

Figure 5. Change mechanism as paths for change for determining degree of changeability of a system.

2.4 Implications of definition.

As described, the changeability of a system is determined by the number of acceptable change paths that can be taken by a system. The number of acceptable change paths is determined both by the possible number of end states and the number of change mechanisms available (number of possible effects and number of possible mechanisms). The agent origin serves only taxonomic value in classifying whether a particular change is system-internally or externally motivated. In terms of affecting changeability, the change agent distinction can be used as descriptors on change mechanisms. For example, final assembly of a car could be done by a robot or a team of people. Even though both utilize the "put subassemblies mechanically together" mechanism, these two methods could be distinguished as particular ways to assemble the car, and thereby count as two mechanisms.

Refining the acceptable path counting further, two axes can be defined: number of change end states (specified vs. open-ended) and number of change mechanisms (countable vs. uncountable). These ranges correspond to finite vs. infinite numbers. Figure 6 below shows four possible cases in which a system can fall: (1) countable end states with specified mechanisms,

(2) countable end states with open-ended mechanisms, (3) uncountable end states with specified mechanisms, and (4) uncountable end states with open-ended mechanisms. The purpose of Figure 6 is to point out the differing degrees of changeability that can occur in a system. The remainder of the paper will discuss an approach to the case of quantifying possibly large, but finite numbers of mechanisms and end states in order to quantitatively assess the changeability of systems on a common basis (lower left quadrant of Figure 6).
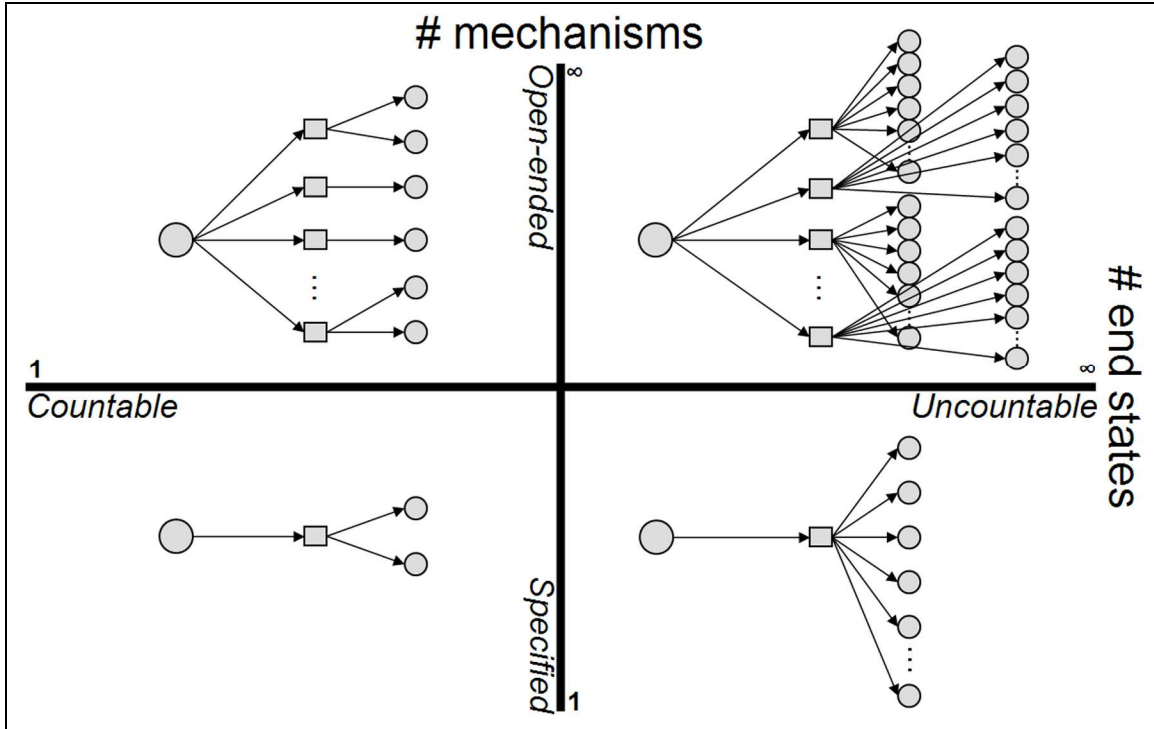

Figure 6. Two axes for counting change paths: number of mechanisms and number of end states, both on finite to infinite scales.

The meanings of flexibility, adaptability, scalability, modifiability, and robustness have been clarified so that a more rigorous consideration can be applied to system specification, evaluation, and verification. Subsequent sections of the paper will describe quantification approaches to these ilities so that a decision maker requesting "flexibility" in the system can move from an abstract notion to a quantifiable and verifiable system specification.

The ilities described by the change taxonomy is not exhaustively complete, for example it excludes agility, which is one of the four central changeability ilities in [Fricke and Schulz, 2005]. The reason is that agility is the ability of a system to make a change quickly. In this way, agility is a modifier describing the nature of the change, just as flexibility and adaptability describe the location of the change agent. An agile system is one that can make many types of changes "quickly," though the subjective assessment of "quickness" will vary from decision maker to decision maker, just as "cost" acceptability for following change paths. [Dove, 2005] defines agility in a similar manner to changeability, encompassing adaptability, flexibility, and robustness, along with notions of ease of change in terms of both dollars and time.

2.5 Changeability Example.

Consider the example where a designer wishes to determine the changeability of a computer system. If external change agents, such as users and technicians could make alterations to the system, then those alterations would be flexible-type changes. If internal change agents, such as automatic software updating, could make alterations to the system, then those alterations would be adaptable-type changes. In this way a computer is both flexible and adaptable, depending on the change under consideration. If the computer can have some of its parameters changed in level, then it is scalable, such as in memory. New capabilities can be added, to an extent, such as DVD-burners or TV-tuners, so the computer is modifiable in those new capabilities. Of course, if the cost of any of these changes is too much for a particular user, then those changes would not be considered to be acceptable and would not count toward the changeability of the computer. The more low cost the change mechanisms, both in terms of effort and dollars, the more changeable the computer will be perceived by the general user-base. Given Figure 6, each of the modifiable and scalable changes in the example would correspond to new end states for the computer.

3 Changeability Quantified

Designing systems for changeability can only really be achievable if there is an approach for quantification of changeability. Further, there must be a method for comparison of the possible options in order to understand which designs are more changeable than others. As such, tradespace exploration, which uses a parameterization of the system under consideration, can help to elucidate the broad comparison of the changeability of various system configurations.

3.1 Tradespace parameterization of systems.

Quantification is a necessary next step in order to develop a concrete specification for changeability. A reasonable approach to comparing a large number of systems simultaneously is through a tradespace [Ross and Hastings, 2005]. Figure 7 below depicts the elements that go into tradespace development. Typically during concept exploration, a number of system designs and concepts are considered and assessed in terms of cost and benefit (i.e., value) to decision makers. The design variable set, $\{DV^N\}$, represents the technical degrees of freedom for the system and can be assessed in terms of cost to develop, C, through the mapping f_C: $\{DV^N\} \rightarrow C$, which can be instanced, for example, through cost models. In this parameterization scheme, each design is represented by a set of N design variables. Fully enumerating all possible values for the set of design variables results in a space of designs.

The attribute set, $\{X^M\}$, is a parameterization of value perceived by particular decision makers. Each decision maker specifies his or her own set with acceptable ranges, but whose specific values are derived from system designs being considered. The attributes can be aggregated in terms of value delivered to a decision maker through the concept of utility, U, with a function mapping f_U:$\{X^M\} \rightarrow U$, which can be instanced, for example, through Keeney-Raiffa multi-attribute utility functions [Keeney and Raiffa, 1993]. In this parameterization scheme, each design is assessed in terms of a set of M attributes, which are then evaluated and aggregated in terms of utility. If formal utility functions cannot be defined, other measures of "goodness" can be substituted, though with less rigor, and therefore less confidence. In order to rank alternatives, a single scalar metric of "goodness" is required and used, either explicitly or implicitly.
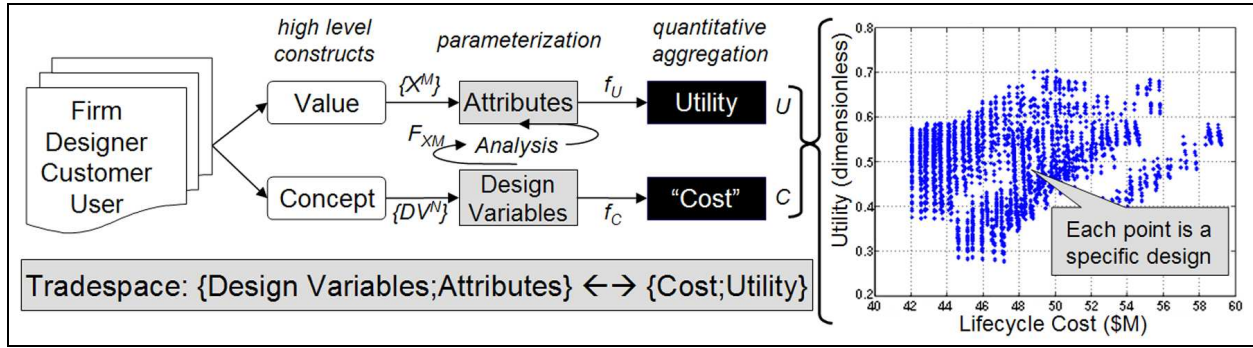
Figure 7. Tradespace defined as shorthand representations of designer controlled technical parameters and stakeholder perceived value parameters evaluated in terms of utility and cost (i.e., benefit and cost).

In order to determine the particular attribute set values for a particular design, the mapping $F_{XM}:\{DV^N\} \rightarrow \{X^M\}$ must be developed, which can be instanced through models and simulations in the analysis phase. Examples can be found in recent tradespace literature [Derleth, 2003; McManus and Schuman, 2003; Spaulding, 2003; Ross, 2004].

3.2 Tradespace networks: quantifying changeability through the Filtered Outdegree.

The typical tradespace plot displays the system designs on a Cost-Utility space, showing the resources required (cost) and value delivered (utility) for the systems in a concise format. A Pareto Set characterizes those "non-dominated" designs of highest utility at a given cost, across all costs, or those of lowest cost at a given utility, across all utilities. This set often shows the tradeoff of cost incurred for increased value. Considering each design as a potential starting or ending state for change, the tradespace frame suggests a mechanism for considering the changeability of system designs. As mentioned in the previous section, change specification requires a beginning state, an ending state, and a transition path. If in addition to specifying design parameters (static representations of a system) designers also specify transition paths (dynamic change opportunities), a traditional tradespace can become a *tradespace network* [Ross and Hastings, 2006].

A network is a model representation of nodes and arcs. Each node represents a location, or state, with each arc representing a path that connects particular nodes. In a tradespace network, system designs are nodes and the transition paths are arcs. Each arc represents a transition with a "cost" in terms of both dollars and time. The transition paths represent each of the potential change mechanisms, with change agent, available to a particular design. Figure 8 shows a traditional static utility-cost tradespace transformed into a tradespace network after the specification of transition rules, which are used to generate transition paths between design nodes. Transition rules, such as "burn on-board fuel to lower apogee" can be applied across a tradespace in an automated fashion to connect nodes efficiently. It is important to recognize that given the tradespace network representation, transition rules are necessarily defined in terms of changes to the design variable set. A design variable set, which is a parameterization of a system design concept, can include both physical and operational parameters. Therefore, valid transition rules can result in changes in either of these types of parameters. Designs that can follow more

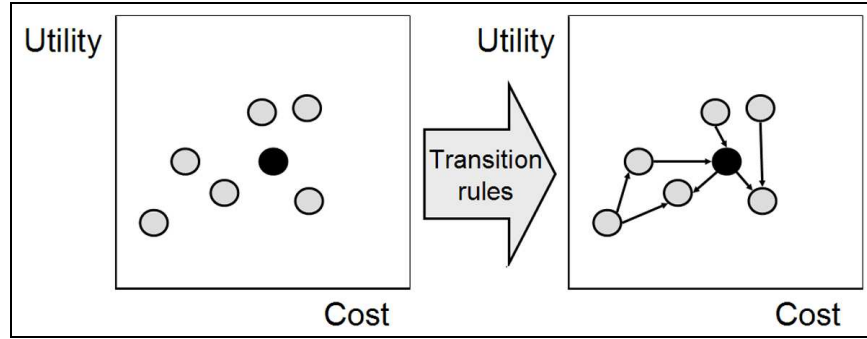transition paths will have more outgoing arcs connecting it to other designs.

Figure 8. Specifying transition rules transforms a tradespace into a tradespace network, with transitionable designs accessible through heterogeneous arcs for each transition rule (change mechanism with agent).

Each decision maker will have an acceptability threshold for time or money spent for enacting change. The number of outgoing arcs from a particular design is called the *outdegree* for that design. The number of outgoing arcs from a particular design whose cost is less than the acceptability threshold, $\hat{C}$, is here introduced as the *filtered outdegree* for that design [Ross, 2006]. The filtered outdegree is a quantification of the apparent changeability for a design for a decision maker, as summarized in Figure 9. The higher the filtered outdegree of a design, the more changeable it is to that decision maker.

The objective, coupled with subjective nature, of the filtered outdegree captures the apparent relativity in perceived changeability of various designs: *what may be changeable to one decision maker may not be perceived changeable to another*. The subjective acceptability threshold differentiates the results per decision maker. Acceptability threshold "cost" can be on dollars, time, or any other resource that must be "spent" in order to follow a path. The objective outdegree calculation provides a mechanism for system designers to explicitly improve the potential changeability of a system: increase the number of outgoing arcs (add new transition rules), or reduce the cost of following outgoing arcs (increase the likelihood for arcs to cost less than acceptability threshold). The subjectivity in the filtered outdegree means that the setting of the threshold is subjective to the particular decision maker and his preferences for spending resources for change. The full outdegree, without filter, is an objective quantity upon which all people will agree, given a set of enumerated design variables and a set of transition rules.

In addition to the sets $\{X^M\}$ and $\{DV^N\}$, and scalar functions f_U, f_C, and vector function $\boldsymbol{F_{XM}}$, another mathematical construct can be created to aid in the analysis of the tradespace network, which is the accessibility tensor $\boldsymbol{T_{ijk}}$. The (i,j,k) entry of the accessibility tensor contains the "cost" of the transition from design i (i.e., $\{DV\}_i$) to design j (i.e., $\{DV\}_j$), following transition rule k. For R rules and S designs, the size of the tensor is R x S x S. An algorithm can be applied to search the tensor to determine the filtered outdegree function, OD($<\hat{C}$), which provides the outdegree for a given design as a function of acceptable cost threshold.
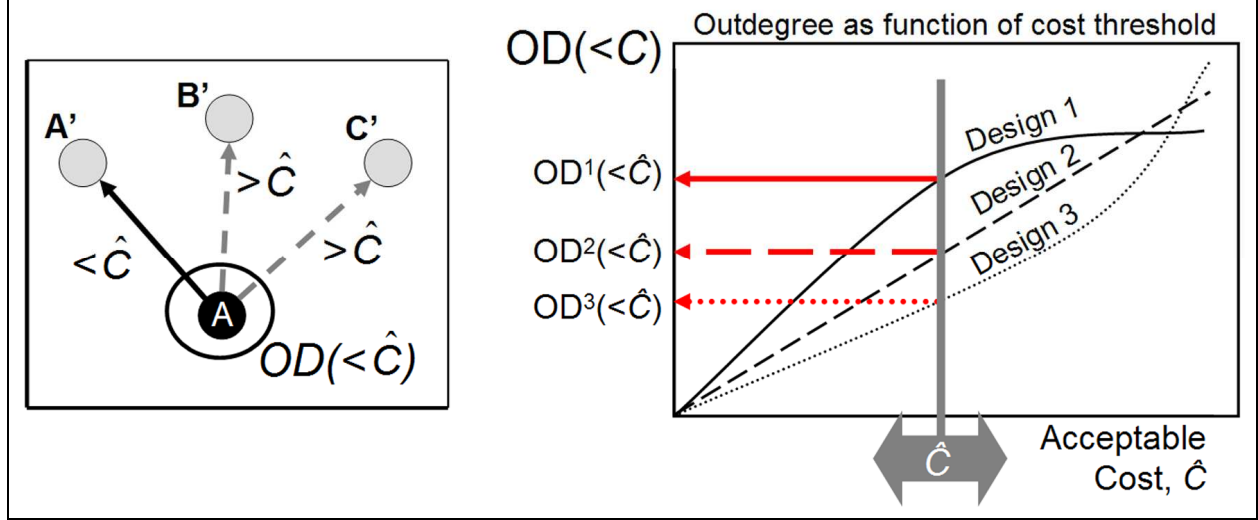
Figure 9. Filtered Outdegree, a measure of changeability, counts the number of potential transition paths available to a design (left), filtered by acceptable cost for change by a particular decision maker (right).

3.3 Other ilities quantified.

Now that the more generalized changeability has been quantified, the question of quantifying the specific changeability types can be pursued. How does one determine if system A is more *flexible* than system B? For comparing flexibility, determine the filtered outdegree for systems A and B, only counting change mechanisms caused by system external change agents. Similarly, for comparing adaptability, only count change mechanisms caused by system internal change agents. For quantifying in terms of the change effects, modifiability, scalability and robustness, formal equations can be defined.

Given a tradespace of size S, a rule set of size R, a vector mapping of design parameters to attributes F_{XM}, and a accessibility tensor T_{ijk} keeping track of change mechanism costs for paths from design i to design j using mechanism k, the change effect ilities can be quantified with equations, thereby enabling the application of algorithms to explicitly calculate the modifiability, scalability, and robustness of designs. (Note: $F_{XM}(\{DV\}_i) = \{X\}_i$).

Modifiability of design i to adding or subtracting attribute X^m.

$$\text{modifiability}_i^m: \qquad \sum_j \Big[F_{XM}(\{DV\}_i) \cap F_{XM}(\{DV\}_j) = X^m \text{ for } T_{ijk} < \hat{C} \Big], \forall j \in S, k \in R$$

For example, modifiability_i^m, or the modifiability of design i in attribute m, is the number of paths connecting design i to designs that are the same as design i except that have attribute m added to or subtracted from its attribute set, but whose transition mechanism costs less than the acceptability threshold.

Scalability of design i to raising or lowering the value of attribute X^m.

12

scalability$_i^m$: $\qquad \sum_j \left[\left[F_{XM} \left(\{DV\}_j \right) \right]^m - \left[F_{XM} \left(\{DV\}_i \right) \right]^m \neq 0 \text{ for } T_{ijk} < \hat{C} \right], \forall j \in S, k \in R$

Robustness is a more complex quantity, as it requires more information to calculate, and will be discussed later in the paper.

3.4 Examples.

The following few examples illustrate the five changeability concepts introduced in this paper: flexibility, adaptability, scalability, modifiability, and robustness (in value).

3.4.1 Flexible Change.

As an example illustrating ambiguity and conflict over perceptions of "flexibility," consider a debate in academic circles that entails the apparent conflict between two professors' definitions of flexibility. On the one hand, Professor A, of Electrical Engineering and Computer Science, believes that networks, such as a computer network, are highly flexible, while a space system is not. Professor B, of Aeronautics and Astronautics Engineering, believes that a space system that can be changed from halfway across the solar system to be able to perform a new mission is flexible, where the comparison to the flexibility of a computer network is irrelevant. The definition for flexibility given in this paper reconciles these two views and provides a good example for the definition.

The question of whether a system is flexible is the same as asking whether cost($DV_i \rightarrow DV_j$) < $\hat{C}$, *as accomplished by a system-external agent*. For the computer network question, the physical change could correspond to physical network changes (wires=arcs), computers (nodes), or informational, such as routing of packets. For these classes of changes of a computer network, the cost to change can vary from thousands of dollars down to the order of pennies, or $\$10^3$ to 10^{-2}. Additionally, the time for a change can vary from hours down to fractions of a second, or 10^3 to 10^{-2} minutes. For the space system in question, the physical change could correspond to the change of on-board payloads, or loaded operational software. For these classes of changes of a space system, the cost to change can vary from billions to hundreds of thousands of dollars, or $\$10^9$ to 10^5. Additionally, the time for a change can vary from years to days, or 10^6 to 10^3 minutes. The key point here is the subjectively set threshold for change cost acceptability, $\hat{C}$, including both money and time. For Professor A, these thresholds are probably on the order of $\$10^{-1}$ and 10^{-1} minutes, respectively, while for Professor B, these are probably on the order of $\$10^6$ and 10^5 minutes, respectively. Figure 10 illustrates the differences in system experiences and expectations for what is an acceptable "cost" for change for each professor. In a sense, both professors are correct and consistent in their definitions.

An important distinction to make, however, is that if the change agent for these two cases is located inside the system boundary, the change is considered an adaptable one. This example considers outside human intervention as the cause for change, though one could imagine a self-changing network or autonomous satellite, especially if the time scales for change are shorter than the capabilities of a human. For examples of time scale and spatial scale comparison of the abilities and limitations of humans versus machines on standard mission tasks see [Hall 1985].
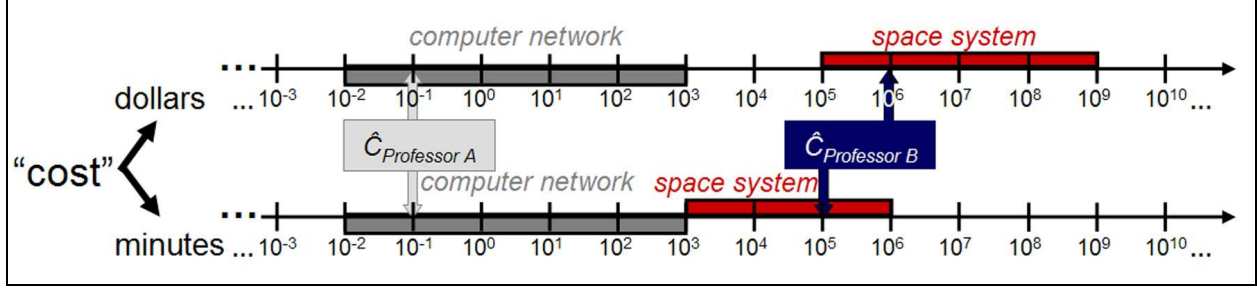
Figure 10. Professor A and Professor B "cost" thresholds compared in terms of acceptable dollars and minutes for change in two classes of systems: space systems and computer networks.

3.4.2 Adaptable Change.

As highly adaptable systems, humans constantly employ their Psychomotor, Intellectual, and Sensory/Perceptual abilities to create change of themselves. Consider weight-training and other exercise as a long term change from an "out-of-shape" system to a "fit-as-a-fiddle" system. Those people (systems) who perceive the cost of doing such an adaptable change as acceptable, execute on it, i.e, the question of whether to pursue an adaptable change through the exercise mechanism is whether $\text{cost}(DV_i \rightarrow DV_j) < \hat{C}$, *as accomplished by a system-internal agent* (the self). In other "words" the question one asks oneself is whether the: $\text{cost}(DV_{outofshape} \rightarrow DV_{fitasfiddle}) <$ (*max willingness to pay gym/trainer and max willingness to wait for results*).

The general consensus that humans are highly adaptable is because over a very large range of possible and actual changes, people decide that the cost is acceptable (i.e., the cost of self-change is reasonable, both in terms of dollars and time). Making a person more adaptable entails either decreasing the actual cost for the change, or increasing the cost threshold for the change. Personal training can have either effect, by increasing the number of mechanisms for personal growth on the one hand, and by increasing a person's patience on the other hand.

3.4.3 Scalable Change.

Suppose a consumer has a camera and cares about its megapixel rating, $X^{Megapixel}$. The current rating for the camera is $X^{Megapixel}_{current} = 4.0$. He now desires that the camera have $X^{Megapixel}_{future} = 7.0$. The change in a level of a current attribute, megapixel rating, is an example of a scalable change. The question facing the consumer is whether $\text{cost}(X^M_{current} \rightarrow X^M_{future}) < \hat{C}$. The mechanism for achieving the change must be determined through the inverse mapping F_{XM}^{-1}, $F_{XM}^{-1} : \{X^M\} \rightarrow \{DV^N\}$, for each attribute current and future. What design parameters need to change and what does it cost in order to effect a scalable change in megapixels? Possible answers include modification of the optics and charge-coupled device (CCD) photon receiver in the current camera, or throwing out the current camera and purchasing a new one (depending on the consumer, the costs for these mechanisms may vary, as will the subjective cost threshold—a do-it-yourself engineer may prefer the modification route, while the typical user may prefer the new purchase). In any case, if the cost for change is acceptable, the camera system has undergone a scalable change for the megapixel attribute.

3.4.4 Modifiable Change.

A simple example of a modifiable change is that which is often made to a personal computer. Suppose a decision maker has a standard computer with a monitor, keyboard, mouse, and hard drive, all of the standard features of a computer purchased circa 1999. Changes occur in the market place, with new technologies being offered to enhance computer capabilities. The decision maker decides at some point later in time that he really would like to be able to burn DVDs. That capability was absent from the original computer system, but with a simple change to the computer, that capability can be added. The decision problem posed to the decision maker is whether $\text{cost}(\{X^M\} \rightarrow \{X^M\} \cup \{X^{DVDburning}\}) < \hat{C}$. In order to determine the answer, the inverse mapping F_{XM}^{-1}, $F_{XM}^{-1} : \{X^M\} \rightarrow \{DV^N\}$, must be determined. Fortunately, the design problem has already been solved through modular design. All that is needed is the addition of an external (or internal) DVD-R drive. In this case, the mapping between design- and value-space is straightforward, $X^{DVDburning} \rightarrow DV^{DVDburner}$. Thus, the question becomes whether the addition of the $DV^{DVDburner} < \hat{C}$? (Of course, this is the question posed by most consumers when shopping for computer additions.) The beauty of modular design is that the cost of additional modules is often relatively low, thereby increasing the likelihood that the cost for change is less than $\hat{C}$.

3.4.5 Value Robustness.

Value robustness is the application of the concept of robustness to value, where the goal is to achieve insensitivity in perceived value of a system under changing preferences, environments, and system offerings [Ross, 2006]. Suppose a decision maker looking to procure a new box has two attributes: size and loudness. Four system offerings exist and are given in Figure 11 below. Size is more important than loudness. Within size, big is preferred to small. Within loudness, loud is preferred to quiet. Applying these preferences, the systems can be ordered according to their utility:

$$U(3) > U(2) > U(1) > U(4).$$

Thus the decision maker should choose system offering (3) in order to maximize his value. Suppose something happens and the decision maker now cares about color as well as size and loudness. Color is about as important as size, which is more important than loudness. Within color, red is preferred to gray is preferred to black. Applying these new preferences, the systems can be ordered according to their utility at time t=2:

$$U(4) > U(2) > U(3) > U(1).$$

Notice that at time t=2 the decision maker should choose system (4) in order to maximize his value. If, however, the decision maker must choose a system at time t=1, the best choice is to choose system (3) and then change it to system (4) at time t=2. If the switching costs are high,

however, the net value to the decision maker might be less than choosing system offering (2), which is second best in both time periods and entails no switching cost. Offering (2) is robust in value under this preference change. In this example, $Ut2 \neq Ut1$, and value robustness can be achieved by having $DV_i = DV_j$ when switching costs are high, i.e., $cost(DV_3 \rightarrow DV_4) > \hat{C}$. Or it can be achieved by having $DV_i \neq DV_j$, when switching costs are low enough, i.e., $cost(DV_3 \rightarrow DV_4) < \hat{C}$. The first case suggests the decision maker choose $DV_i = DV_j = DV_2$, and the second case suggests the decision maker choose $DV_i = DV_3$, and $DV_j = DV_4$.
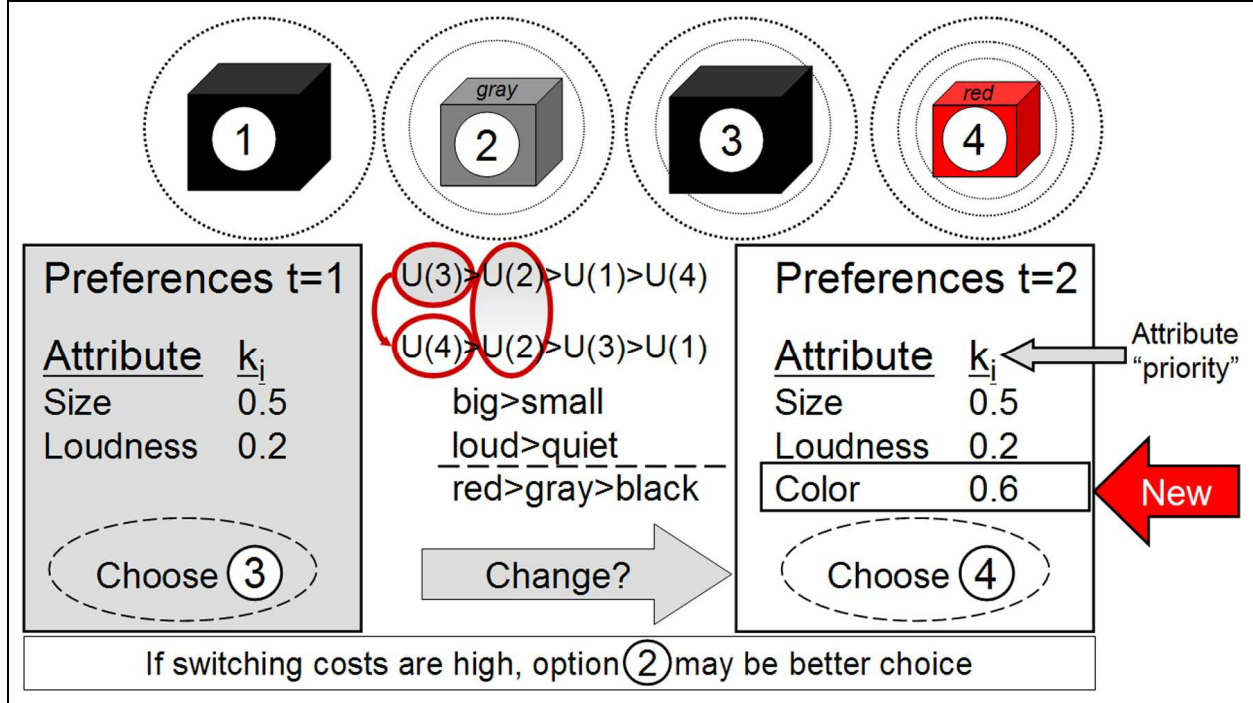

Figure 11. Example value robustness: Choosing boxes.

Choosing a passively value robust design requires finding designs that remain high value perceived across various future scenarios. A *Pareto Set* characterizes those "non-dominated" designs of highest utility at a given cost, across all costs, or those of lowest cost at a given utility, across all utilities. Passive value robustness can be quantified as the *Pareto Trace Number*, which is the number of scenarios, or "Epochs" [Ross and Rhodes, 2007], whose Pareto Set contains that particular design, reflecting the designs that have the most efficient utility for a given level of resource expenditure. Figure 12 shows the definition along with a tradespace over time plot showing the Pareto Trace designs that remain in the Pareto Sets across various Epochs. A relatively large Pareto Trace number implies a high passive value robustness factor for a particular design and can be a function of excess capability, insensitivity to the particular change scenario, or a locally stable region in a tradespace [Ross, 2006].
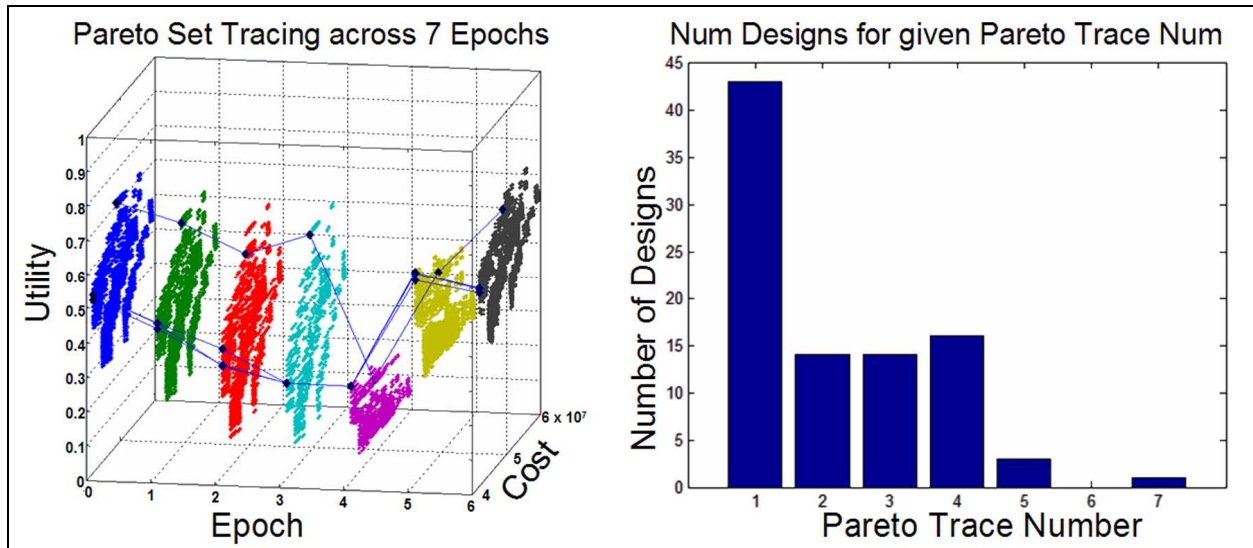
Figure 12. Passive value robustness as Pareto Trace Number.

4 Discussion

This section of the paper will discuss several topics. The first is the relation of the changeability taxonomy and quantification to valuation methods, including recent research directions. The second discussion area is the topic of dealing with uncertainty and its relationship to changeability. The third discussion involves the tension between changeability and robustness, and how this tension may be a positive factor in design. The last discussion involves general implications of this work for design and for policy.

4.1 Relation to Valuation Methods.

Valuation is the quantification into a single metric of the benefit in relation to cost of something. While the identification and quantification of several ilities was undertaken in this work, as well as motivation for pursuing these ilities, the explicit valuation of those ilities was intentionally excluded from consideration. Valuation of ilities as a single metric is an additional layer of analysis that can be put on top of the proposed framework. (Example valuation methods include Net Present Value, and Return on Investment calculations.) The reason valuation was excluded from this current work is that all valuation techniques rely upon specific assumptions regarding how to collapse time, utility, cost, and uncertainty into a single metric. Layering such assumptions into the framework would obfuscate the trades which exist among these quantities, as well as reduce the generalizeability of the framework.

As pointed out in [Roberts, 2003], the ability to change the onboard delta-v capability for the Space-based Radar system represents a real option for the system to alter its future state. Using the tradespace parameterization scheme described previously, this real option capability can be codified as a "path-enabling" variable, which is a parameter that reduces the cost or "turns on" particular change mechanisms for the system. Path enabling variables can be binary, discrete, or continuous and can represent physical or operational choices for the system. An example of a path enabling variable is the use of "commercial-off-the-shelf" (COTS) parts (binary: yes or no), which may reduce the cost of replacing or building components, though perhaps at higher risk.

Another example is the "amount of modularity used," which may reduce change cost, though perhaps at a higher initial cost. It has been suggested that one of the benefits of modularity is the ready ability to reuse old components, while freeing up possibilities for system variation that would not be available in an integral design [Lipson, Pollack, et al., 2001; Baldwin and Clark, 2000]. The pursuit of various architecture strategies [Moses, 2004], including, but not limited to modularity, platforming, layering, networking, pursuing part or process commonality, design reuse [Sivaloganathan and Shahin, 1999] and others, balance the upfront cost of fielding the architecture against the cost for changing the system at some point in the future. Trading these architectures against each other can be done on a common basis through the comparison of their outdegree functions (i.e. the function of number of available change mechanisms versus change cost for each of the architectures when applied to a system of interest). [Ross, 2006] gives examples of path enablers in case studies of the Terrestrial Planet Finder (TPF) astronomical system and the Joint Direct Attack Munition (JDAM) bomb modification kit systems.

Path enabling variables differ from design variables in that design variables are generated in order to create value, while path enablers are generated in order to create or enhance changeability. Real options and path-enabling variables are similar in that both factors allow for a system change, and may not contribute to system value if left unused. Valuation of path-enabling variables can be done utilizing real options analysis, and likewise, identification of real option opportunities can benefit from the definition of path-enabling variables. The present work is considered to be both preceding and complementary to traditional real options analysis. Ongoing work on defining real options "in" systems tends to focus more on valuation, rather than discovery of the key variables that should be investigated for real options [Wang and de Neufville, 2006; Wilds et al., 2007]. An exception is [Bartolomei, Hastings et al., 2006], whose research is trying to discover "hot spots" or key system aspects that should be addressed for leveraging changeability, that is identifying real option opportunities.

4.2 Dealing with Uncertainty.

One of the key motivations for system changeability is the inevitability of change in the future system context. Not knowing the specifics is akin to admitting uncertainty in future markets, technologies, and environments. Other types of uncertainty also exist, both regarding past and present factors affecting system value. Instead of motivating flexibility as a response to uncertainty [Shishko et al., 2004; Wilds et al., 2007], the present work is motivating changeability as a response to inevitable change [Fricke et al., 2000], which is a primary salient issue in concerns with uncertainty. Building changeability into systems admits the inability to accurately predict the future, while embracing the Bayesian technique of constantly improving estimates given new information. No longer does a system have to settle for putting off fixing its requirements for as long as possible in order to better match "actual" preferences. A changeable system can change to meet the "actual" *and* future preferences of stakeholders given improved information as it is revealed over time.

4.3 Tension between Changeability and Robustness.

One common perception is that system properties are in tension, such as flexibility and robustness, where one must give up one to have more of the other. Such is not necessarily the case; it depends on the parameters under consideration. For example, making a computer robust to noise and physical impacts does not reduce its modifiability for changing components. The

robustness of the system is embedded in its modular design; each module employs robust design principles and together the overall system maintains robustness. Robustness obtained through redundancy may suffer through modularization due to the reduction of interconnections inherent in modular design. [Fricke and Schultz, 2005] discusses the tensions between modularity and redundancy. But modularization is just one approach towards achieving changeability through its ability to reduce the costs for change. Likewise, redundancy is just one approach towards achieving robustness.

If the goal for system design and development is to deliver value to stakeholders over the system lifecycle, in the face of changing contexts, expectations, and technologies, then achieving "value robustness" is the ultimate goal for the designers. Value robustness can be achieved through either passive or active means, with the former more akin to traditional robust approaches, and the latter embracing changeability as a dynamic strategy for value sustainment. Passive value robustness delivers value through the development of "clever" designs that are perceived to maintain value over time. Successful development of passively value robust systems requires a bit of luck as well as anticipation of future system context, including potential value perceptions and the competitive environment with alternative systems. Active value robustness requires less omniscience, but does have the added complexity of needing a change agent changing the system over time to maintain high value perception.
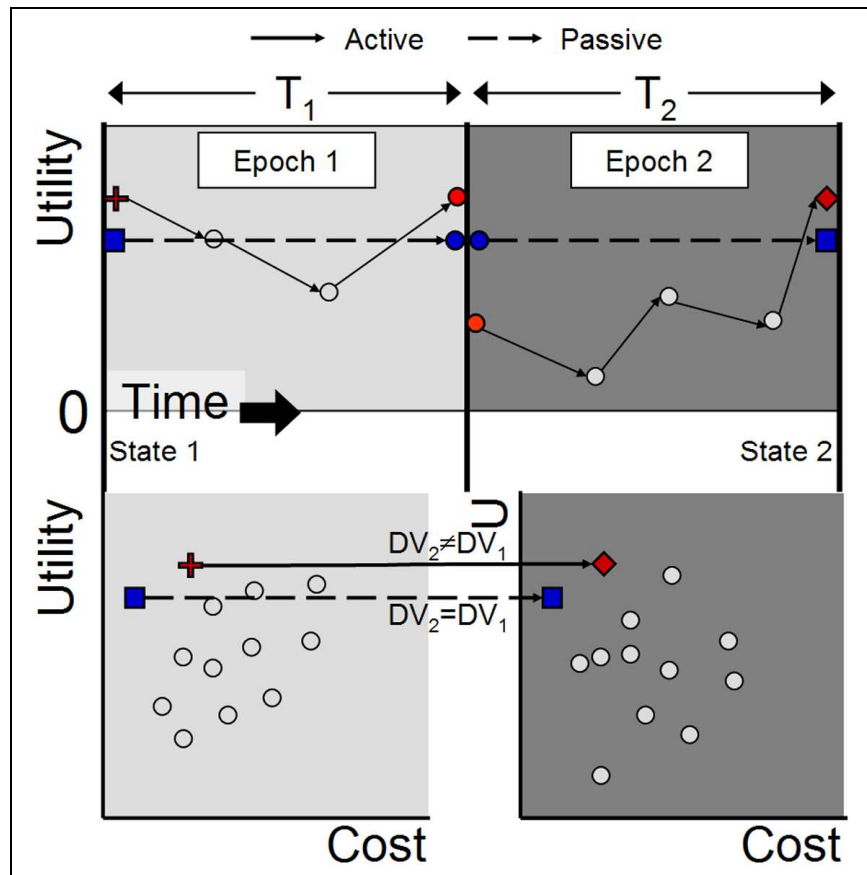

Figure 13. Active versus passive value robust strategies across two time periods.

Figure 13 depicts the two value robustness strategies. The dark blue square design follows the passive value robust strategy shown by the dashed line, while the red cross design

follows the active value robust strategy shown by the solid line. The top portion of the figure shows utility versus time across two time periods (epochs), while the bottom portion shows utility versus cost at the beginning of the first time period (epoch) and the end of the second time period (epoch) respectively. Comparing the utility-cost tradespaces show both value robust strategies result in "constant" utility across changing context, though the active strategy results in a different end design than the passive strategy.

4.4 Implications for Design.

Given the motivation of a value-centric definition of changeability and robustness, the implications for design become clear. Designers must no longer only consider the aspects of a design that meet today's needs and requirements, but rather keep an eye to the future by building in change mechanisms into a system to allow for change. Designers can parameterize the system both in terms of design parameters as well as path enablers, the former intended to create value and the latter intended to lower the cost or create the possibility for following a change path. In order to explicitly address the desire of a system customer to have "flexibility" it is necessary to push back and inquire for more information about the desired change effect, as desiring "flexibility" alone is an imprecise request.

In order to write requirements and evaluate systems on that basis, it is necessary to conduct four steps:

Step 1. Determine each decision maker's willingness to pay for change, meaning set the initial acceptability "cost" threshold.

Step 2. For a particular change ability desired, specify the origin of the change agent: internal (adaptable-type), or external (flexible-type) to the system boundary. Discussions around this issue will focus whether the system needs agents built in, or from without in order to cause the change to occur.

Step 3. For the same change ability specified in step 2, specify the desired change effect: change in level of a parameter (scalable-type), change in parameter set (modifiable-type) and in what parameter. For example, one could specify "scalable change in number of satellites," or "scalable change in total bandwidth." If the desire is for "no change" (robust-type), then robust in what parameter to what type of disturbance (e.g. environments, preferences, constraints, etc.). For example, one could specify "robust in output voltage to changes in temperature," or "robust in response time to changes in atmospheric conditions and fuel costs."

Step 4. Once the change capability is specified in terms of change agent location and change effect on which parameters, then dynamic tradespace analysis can be performed to determine the outdegree functions of various designs. The filtered outdegree of a design, which is a measure of its apparent changeability, can be assessed, with only mechanisms that match the specified change type counted toward the outdegree. For example, if a decision maker desires the system to be flexibly scaleable in image resolution, then only change mechanisms that result in a change in level of image resolution performance will be counted towards the calculation of the outdegree of designs in the tradespace network. In this way the specific flexibility (in terms of scaling image resolution) can be evaluated and compared against requirements. Desiring "flexibility" in a general sense is meaningless from a design perspective, since it is an inherently ambiguous term, and

must be further specified in order to be quantifiable and testable as a design goal.

Guiding designers, in addition to the concrete definitions in this paper, are the heuristics and guiding principles in [Fricke and Schulz, 2005], which lists three basic principles and six extending principles for designing for changeability. These principles are techniques that will allow designers to creatively develop change mechanisms that will more likely be acceptable to decision makers.

4.5 *Implications for Policy.*

One key drawback to adding changeability into systems is the potential for adding cost, with no obvious offsetting benefits, when change mechanisms remain unused. While not necessarily required, adding a "real option" often has a carrying cost associated with it and if unused, adds to the overall cost with no perceived benefit, making them unattractive to decision makers with limited budgets. Acquisition policy must allow for the positive accounting for unused change mechanisms through allowing for the carrying of unused change costs by appropriate organizations. As an example, consider the development of the JDAM system, which was developed by the Boeing Company. Boeing was allowed to have class 2 change authority on the system, which facilitated the changing of the design over time with less "cost," as well as allowed to maintain ownership over the design itself, which shifted the cost of calculation of design ownership from the customer (government) to the builder (Boeing). These two policies empowered Boeing to add path enablers to the system, which lowered the cost for future change, but may in the short term have added cost [Ross, 2006]. A similar situation is occurring in the services industry; as ownership costs shift from the customer to the supplier, life cycle cost considerations, including future change costs, will need to be considered more by the supplier than in the past.

5 Conclusion

As described earlier in the paper, when presented with the set of "ilities" by Dr. Sambur as aspects of "Robustness," system engineers from across industry, government, and academia had difficulty aligning classical conceptions of robustness with the set. Discussions pointed out an inability in the community to design and assess systems on the basis of these "ilities," especially in terms of concrete and unambiguous definitions. Through the development of the changeability taxonomy and quantification presented in this paper, the system "robustness" requested by Air Force leadership is revealed to be, in fact, *value robustness*, which is a strategy that may use various other ilities in its attempt to maintain system value delivery over time, in spite of context changes. Passive value robustness more closely approximates traditional notions of robustness through the pursuit of developing systems insensitive to external changes. Active value robustness embraces changeability, and through dynamic matching of system offerings to stakeholder expectations given varying contexts, can provide value over time even in highly uncertain or unexpected future scenarios.

The changeability definitions discussed in this paper provide a concrete basis for conversation about the ilities, specification for design, analysis techniques for comparison, and a new paradigm for considering systems as dynamic constructs for creating value over time. Designing for value robustness is a sound dynamic strategy for maintaining system lifecycle value, and the ilities of flexibility, adaptability, scalability, and modifiability no longer remain in the ambiguous realm of unquantifiable and unverifiable desires.

Acknowledgements

References

C.Y. Baldwin and K.B. Clark, *Design Rules, Vol. 1: The Power of Modularity*, MIT Press, Cambridge, MA, 2000.

J.E. Bartolomei, D. E. Hastings, R. de Neufville and D.H. Rhodes, Screening for Real Options "In" an Engineering System: A Step Towards Flexible System Development--PART I: The Use of Design Matrices to Create an End-to-End Representation of a Complex Socio-Technical System, 4th Conference on System Engineering Research, Los Angeles, CA, April 2006.

W. Chen and C. Yuan, *A Probabilistic Design Model for Achieving Flexibility in Design*, ASME Journal of Mechanical Design, 120 (1998), no. 9.

J.E. Derleth, "Multi-Attribute Tradespace Exploration and its Application to Evolutionary Acquisition," S.M. in Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA, 2003, p. 144.

R. Dove, *Fundamental Principles for Agile Systems Engineering*, 2005 Conference on Systems Engineering Research, Hoboken, NJ, March 2005.

E. Fricke, B. Gebhard, H. Negele and E. Igenbergs, *Coping with Changes: Causes, Findings, and Strategies*, Systems Engineering **3** (2000), no. 4, 169-179.

E. Fricke, and A.P. Schulz, Design for Changeability (DfC): Principles to Enable Changes in Systems Throughout Their Entire Lifecycle, Systems Engineering **8** (2005), no. 4, 342-359.

R.E. Giachetti, L.D. Martinez, O.A. Saenz and C.S. Chen, *Analysis of the structural measures of flexibility using a measurement theoretical framework*, International Journal of Production Economics **86** (2003), 47-62.

S.B. Hall (Editor), *The Human Role in Space: Technology, Economics and Optimization*, Noyes Publications, Park Ridge, NJ, 1985.

K.H. Hwang and G.J. Park, *Development of a Robust Design Process Using a New Robustness Index*, 2005 ASME International Design Engineering Technical Conference, Long Beach, CA, September 2005.

R.L. Keeney and H. Raiffa, *Decisions with Multiple Objectives—Preferences and Value Tradeoffs*, Cambridge University Press, Cambridge, UK, 1993.

D. Keese, A. Tilstra, C.C.C. Seepersad, and K.L. Wood, *Empirically-Derived Principles for Designing Products with Flexibility for Future Evolution*, 2007 ASME International Design Engineering Technical Conference, Las Vegas, NV, September 2007.

A.M. King and S. Sivaloganathan, *Flexible Design: a Strategy for Design Reuse*, Proceedings of Engineering Design Conference '98 on Design Reuse, London, UK, June 1998.

Y. Li, D. Xue and P. Gu, *Design for Product Adaptability*, 2007 ASME International Design Engineering Technical Conference, Las Vegas, NV, September 2007.

H. Lipson, J.B. Pollack and N.P. Suh, *Promoting Modularity in Evolutionary Design*, 2001 ASME Design Engineering Technical Conference, Pittsburgh, PA, September 2001.

H. McManus and T.E. Schuman, *Understanding the Orbital Transfer Vehicle Trade Space*, AIAA Space 2003 Conference and Exhibition, Long Beach, CA, September 2003.

J. Moses, Three Design Methodologies, their Associated Organizational Structures and their Relationship to Various Fields, MIT ESD External Symposium, Cambridge, MA, March 2004.

R. Nilchiani, "Measuring the Value of Space Systems Flexibility: A Comprehensive Six-element Framework," Ph.D. in Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA, 2005, p. 307.

R. Nilchiani and D. Hastings, Measuring the Value of Flexibility in Space Systems: A Six Element Framework, Systems Engineering **10** (2007), no. 1.

S.H. Park, Robust Design and Analysis for Quality Engineering, Chapman & Hall, New York, 1996.

A. Qureshi, J.T. Murphy, B. Kuchinsky, C.C.C. Seepersad, K.L. Wood and D.D. Jensen, *Principles of Product Flexibility*, 2006 ASME International Design Engineering Technical Conference, Philadelphia, PA, September 2006.

P.K.P. Rajan, M. van Wie, M.I. Campbell, K.L. Wood and K.N. Otto, *An Empirical Foundation for Product Flexibility*, Design Studies **26** (2005), no. 4, 405-438.

D.H. Rhodes, *Air Force/LAI Workshop on Engineering for Robustness*, Technical Report, Lean Aerospace Initiative, `http://lean.mit.edu`, July 2004.

C.J. Roberts, "Architecting Strategies Using Spiral Development for Space Based Radar," S.M. in Technology and Policy Program, Massachusetts Institute of Technology, Cambridge, MA, 2003.

A.M. Ross, "Managing Unarticulated Value: Changeability in Multi-Attribute Tradespace Exploration," Ph.D. in Engineering Systems, Massachusetts Institute of Technology, Cambridge, MA, 2006.

A.M. Ross, N.P. Diller, D.E. Hastings and J.M. Warmkessel, *Multi-Attribute Tradespace Exploration with Concurrent Design as a Front-End for Effective Space System Design*, AIAA Journal of Spacecraft and Rockets **41** (2004), no. 1, 20-28.

A.M. Ross and D.E. Hastings, *The Tradespace Exploration Paradigm*, 2005 INCOSE International Symposium, Rochester, NY, July 2005.

A.M. Ross and D.E. Hastings, Assessing Changeability in Aerospace Systems Architecting and Design Using Dynamic Multi-Attribute Tradespace Exploration, AIAA Space 2006, San Jose, CA, September 2006.

A.M. Ross and D.H. Rhodes, *Using Natural Value-Centric Time Scales for Conceptualizing System Timelines through Epoch-Era Analysis*, SEAri Working Paper WP-2007-1-3, `http://seari.mit.edu`, November 2007 (submitted to INCOSE International Symposium 2008).

A.P. Schulz and E. Fricke, Incorporating Flexibility, Agility, Robustness, and Adaptability within the Design of Integrated Systems--Key to Success?, IEEE/AIAA 18th Digital Avionics Systems Conference, St. Louis, MO, 1999.

A.P. Schulz, E. Fricke and E. Igenbergs, *Enabling Changes in Systems throughout the Entire Life-Cycle—Key to Success?*, 2000 INCOSE International Symposium, Minneapolis, MN, July 2000.

R. Shishko, D.H. Ebbeler and G. Fox, *NASA Technology Assessment Using Real Options*

Valuation, System Engineering **7** (2004), no. 1, 1-13.

S. Sivaloganathan and T. Shahin, *Design Reuse: An Overview*, Proceedings of the Institute of Mechanical Engineers, Vol. 213 Part B, 1999.

T.J. Spaulding, "Tools for Evolutionary Acquisition: A Study of Multi-Attribute Tradespace Exploration (MATE) Applied to the Space Based Radar (SBR)," S.M. in Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA, 2003.

N.P. Suh, *Axiomatic Design--Advances and Applications*, Oxford University Press, New York, 2001.

T. Wang and R. de Neufville, *Identification of Real Options 'In' Projects*, 4th Conference on Systems Engineering Research, Los Angeles, CA, April 2006.

J. Wilds, J. Bartolomei and R. de Neufville, *Real Options 'In' a Mini-UAV System*, 5th Conference on Systems Engineering Research, Hoboken, NJ, March 2007.

Technical Biographies

Adam M. Ross. Dr. Adam M. Ross is a Research Scientist in the MIT Engineering Systems Division with the Systems Engineering Advancement Research Initiative (SEAri). His research focuses on managing unarticulated value, designing for changeability, and dynamic tradespace exploration for complex systems. Dr. Ross received his Ph.D. degree from the MIT Engineering Systems Division in June 2006, and was previously a research assistant with the Lean Aerospace Initiative at MIT. He is one of the co-founders of the MIT Systems Engineering Advancement Research Initiative (SEAri) and actively conducts research and advises graduate students. Dr. Ross has published papers in the area of space systems design. He has work experience with government, industry, and academia including NASA Goddard, JPL, Smithsonian Astrophysical Observatory, Boeing Satellite Systems, MIT, Harvard, and Florida State University.

Donna H. Rhodes. Dr. Donna H. Rhodes is a Senior Lecturer and Principal Research Scientist in the MIT Engineering Systems Division, and co-director of the Systems Engineering Advancement Research Initiative (SEAri). Her areas of specialization include technical and management theory and practices for architecting and design of complex systems, systems-of-systems, and enterprises. Prior to joining MIT, Dr. Rhodes had 20 years of experience in the aerospace, defense systems, systems integration, and commercial product industries. Dr. Rhodes is a Past President and Fellow of INCOSE, and is a recipient of the INCOSE Founders Award and several INCOSE Distinguished Service Awards. She has published numerous papers and research reports in the field of systems, and has co-authored industry and corporate engineering policies, standards, and technical reports.

Daniel E. Hastings. Dr. Daniel E. Hastings is a Professor of Engineering Systems and Aeronautics and Astronautics, and Dean for Undergraduate Education at MIT. Dr. Hastings' research concentrates on issues of space systems and space policy. He has published papers and a book on spacecraft-environment interactions and papers on space propulsion and space systems. Dr. Hastings is an AIAA Fellow and a member of the International Academy of Astronautics. He is serving as a member of the National Science Board, the Applied Physics Lab Science and Technology Advisory Panel, and as past chair of Air Force Scientific Advisory Board. He served as chief scientist to the U.S. Air Force, and has led several national studies on government investment in space technology.