Research Highlights:
Architecting Systems of Systems with Illities

Adam M. Ross                Donna H. Rhodes

April 9, 2014
About SEArI
Research to advance theories, methods, principles, and practices for engineering complex socio-technical systems through collaborative research

Faculty collaborators within MIT and several other universities

Ongoing research within ...

Research Leadership: Dr. Donna Rhodes, Director/Principal Investigator & Dr. Adam Ross, Lead Research Scientist
Primary Faculty Advisor: Prof. Daniel Hastings
2014 Researcher Assistants: 4 PhD students, 7 masters students, UROPs

Dynamic Tradespace Exploration
- Multi-Attribute Tradespace Exploration method
- Epoch-Era Analysis for exploring alternative futures
- Case applications for systems and systems of systems

Architecting Under Uncertainty
- Responsive Systems Comparison method
- Principles and metrics for designing for illities
- Valuation Approach for Strategic Changeability

Multi-Stakeholder Negotiations for Architecture Decisions
- Multisensory Tradespace Exploration Laboratory
- Methods for reaching design decision compromises

Methodologies include:
- Engineering Analysis, Decision Theory, Economics, Operations Research, Social Sc. Methods, Probability Theory

Domain Areas:
- Space, Defense, Aerospace, Cyber-Physical, Energy, Transportation, Ship Design
Systems of Systems
System of systems is a collection of task-oriented or dedicated systems that pool their resources and capabilities together to obtain a new, more complex, 'meta-system' which offers more functionality and performance than simply the sum of the constituent systems.

Entanglement of Systems and Enterprises
The understanding of the organizational and technical interactions in our systems, emphatically including the human beings who are a part of them, is the present-day frontier of both engineering education and practice.

Dr. Michael D. Griffin, former NASA Administrator

Dynamic Global Environment
The engineering environment of this century involves collaboration across regions and nations, and coping with changes in policies, resources, markets, technologies, economies, and stakeholder demographics.
What observed mechanisms best predict collaborative systems thinking?

Best Predicting Traits (high-low):
1. Consensus Decision Making
2. Concurrent Program Experience
3. Realistic Schedule (actual and perceived)
4. Overall Creative Environment
5. Real-Time Interactions

Collaborative systems thinking is an emergent behavior of teams resulting from the interactions of team members and utilizing a variety of thinking styles, design processes, tools and communication media to consider the system, its components, interrelationships, context, and dynamics toward executing systems design. (Lamb 2009)

Research - motivated by urgent workforce shortages and changing nature of engineering systems requiring systems skills and collaboration - including: (1) predictive factors for collaborative system; (2) key factors in development of systems engineers, and (3) success factors for collaborative distributed SE


Lamb 2009
SEArI Tradespace Exploration and Evaluation Methods

Multi-Attribute Tradespace Exploration (MATE)
Exploring distribution of attributes, costs, and utilities across many designs

Dynamic MATE
Using tradespace networks to design for and quantify changeability

Epoch-Era Analysis (EEA)
Considering the impact of short run and long run context and needs changes on the success of systems

Tradespace Exploration Lab (TSELab) with VisLab (software)
Interactive tradespace exploration environment

Responsive Systems Comparison Method (RSC)
Using MATE, EEA, and other approaches, RSC is a set of seven processes for gaining insights into developing value robust systems

Valuation Approach for Strategic Changeability (VASC)
Framework and metrics for changeability value in both multi-epoch and era domains

Epoch Syncopation Framework (ESF)
Investigating how epoch ordering and change strategies affect timing of design change decisions

seari.mit.edu © 2014 Massachusetts Institute of Technology
Motivations and Research Approach
Complex DoD systems tend to be designed to deliver optimal performance within a narrow set of initial requirements and operating conditions at the time of design. This usually results in the delivery of point-solution systems that fail to meet emergent requirements throughout their lifecycles, that cannot easily adapt to new threats, that too rapidly become technologically obsolete, or that cannot provide quick responses to changes in mission and operating conditions.

- Office of the Secretary of Defense (SERC RT-18 Task Description, Sept 2010)

Engineering design must move beyond optimization of “first use” considerations in order to create complex systems that are able to sustain value delivery in the face of uncertainty.
Motivations for Dynamic Strategies

The engineering of systems has always considered a multitude of dimensions…. and increasingly requires formal methods and enabling technologies to respond to uncertain and changing futures.

- **STAKEHOLDER NEEDS CHANGE AS PERCEPTION OF SYSTEM AND VALUE DELIVERED EVOLVES**

- **SYSTEMS EXIST IN DYNAMIC CULTURAL, POLITICAL, FINANCIAL, MARKET ENVIRONMENTS**

- **HIGHLY COMPLEX AND INTERCONNECTED SYSTEMS WITH CHANGING TECHNOLOGY OVER LONG LIFESPANS**

---

Engineering complex socio-technical systems in a dynamic world requires multi-faceted methods that evolve over time and through synergies of individual research contributions.

Deere & Company
Designing for a Dynamic World

- Built on decade of foundational research for designing “value sustainable” systems
- Specifically target the high leverage early concept phase
- Metrics inform selection of promising concept designs for further analysis
- Uses exogenous uncertainties to frame the need for the ability of a system to respond to perturbations

Systems developed in dynamic world and must accommodate shifts in context and needs (epoch) across their lifespan (era)

Success for modern systems is strongly determined by being able to respond to perturbations on appropriate timescales
Perverse Emergent Dynamic: Mismatch of Design with Context

1960’s Paradigm

- CORONA: 30-45 day missions
- 144 spacecraft launched between 1959-1972
  (Wheelon 1997)

Evolution to Current State

- Inability to adapt to uncertain future environments, including disturbances, leads to “gold-plated” designs

“Our spacecraft, which take 5 to 10 years to build, and then last up to 20 in a static hardware condition, will be configured to solve tomorrow’s problems using yesterday’s technologies.” (Dr. Owen Brown, DARPA Program Manager, 2007)
More than Missed Opportunities:
Failures from Context Changes

New competitor/technology changes needs before system completed

Adversary timescale shorter than “system” lifecycle

Changing contexts can lead a technically sound system to fail

Changing contexts can have high consequences if systems fail...

Source: Wired Magazine, August 2010
The time in a lifecycle when a (system or context/needs) change occurs is an important consideration for “ilities” and tradeoffs.

The farther back a system change goes into the lifecycle, the longer (usually) it takes before utility is experienced again.

Choices can be made to give an option to change later in lifecycle, or to reduce the time and cost for getting back to operations.

“Success” determined by matching response time and cost to the perturbation.
Underlying Research Approach

- **Prescriptive methods seek to advance state of the practice based on sound principles and theories, grounded in real limitations and constraints**
- **Normative research**: propose principles and theories -- “should be”
- **Descriptive research**: observe practice and identify limits and constraints

**METHODS**

- Model-based approaches, advanced analyses, simulations, metrics, MATLAB, Agent-based and STK Models
- Empirical studies of historical systems, programs, and current practices
- Experiment-based studies: observed decision-making, visualizing complex data sets
Four Research Threads

Engineering “involves a relation among three terms: the purpose or goal, the character of the artifact, and the environment in which the artifact performs”


How can one balance System, Context, and Expectations over time, during engineering design, evaluation and selection, given human cognitive and perceptual limitations?

SEArri research aims to develop a meta-cognitive set of theories and methods to holistically address these challenges.

<table>
<thead>
<tr>
<th>Game-based Research Thread</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSE and Visual Analytics for Negotiation Research Thread</td>
</tr>
<tr>
<td>Theory of “ilities” Research Thread</td>
</tr>
<tr>
<td>Methods and Metrics Research Thread</td>
</tr>
</tbody>
</table>
Research Project Highlights

SoS Architecting with llities
Collaborative Research Project
Sponsor Motivations

• Classical systems engineering recognized as **insufficient** for Systems of Systems (SoS)*
• Architecting **throughout lifespan** as SoS evolves
• Architects need **new ways** of performing work and making system decisions
• **Ilities** are **ill-defined and not adequately considered** during architecting process

A “capability” perspective:

– System of systems engineering is the interdisciplinary, CROSS-SYSTEMS process that ensures the development and evolution of CAPABILITIES to meet multiple stakeholders’ evolving needs across periods of time that exceed the lifetimes of its individual systems

Ref: Dr. Jeremy Kaplan; 1st Annual conference on SoSE

A “connecting the parts” perspective:

– System of systems engineering is the process of discovering, developing, and implementing standards among systems that promote interoperability among systems developed via disjointed sponsorship, management, and primary mission acquisition processes

Maritime Security System of Systems Research Case Study

- Stakeholders want to monitor and protect a particular Area of Interest (AOI)
  - Narrow strait, where a large volume of ships pass through

- Boats are entering and leaving the AOI
  - Commercial /civilian boats
  - Pirates, smugglers, terrorists

- Stakeholders want a system that:
  - Detects, identifies boats
  - Intercepts “bad guys”

- Common SoS problems:
  - Diverse, interconnected assets
  - System is subject to various “disturbances”
  - Components geographically separated and operate under different contexts.
  - Dynamic configuration, technology upgrades, removal of assets

Problem: What will be the “best” system of systems when considering performance, survivability, evolvability, flexibility and cost?
What is a Maritime Security SoS?

System of Systems

- Geographical Separation
  - Mix of land, sea, air contexts
  - Leads to contextual diversity
- Operational & managerial independence
  - Most of the constituent systems act independently within system
- Evolutionary development
  - Components are constantly added, upgraded
  - Assets are removed

Constituent Systems
## Multi-faceted complexity of SoS

### Architecting involves many interrelated form and operation choices

<table>
<thead>
<tr>
<th>Form</th>
<th>CONOPs</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Composition:</td>
<td>• Roles:</td>
</tr>
<tr>
<td>– Swarms of UAVs or mixed</td>
<td>– Distinct roles</td>
</tr>
<tr>
<td>Manned / Unmanned, UCAVs</td>
<td>– Overlapping roles</td>
</tr>
<tr>
<td>or patrol boats</td>
<td>• Geographical segmentation</td>
</tr>
<tr>
<td>– Number of ground</td>
<td>• Take off and landing</td>
</tr>
<tr>
<td>control stations</td>
<td>– From patrol boats (organic)</td>
</tr>
<tr>
<td>– Number of operators</td>
<td>– From mainland (inorganic)</td>
</tr>
<tr>
<td>• Technology</td>
<td>• Interception: UCAV or</td>
</tr>
<tr>
<td>– RF or EO sensors</td>
<td>patrol boats</td>
</tr>
<tr>
<td>• One control station vs.</td>
<td>• One control station vs.</td>
</tr>
<tr>
<td>several stations</td>
<td>several stations</td>
</tr>
</tbody>
</table>

### SOS operates in a dynamic context

#### Temporary context change
- Sudden increase in the number of boats entering AOI
- Intelligent adversaries
- Weather
- Component failure due to corrosion
- Intentional/unintentional removal of assets

#### Permanent context change
- Resource changes
  - Loss of certain frequencies
  - Fuel price increase
- Technology upgrades
  - Better sensors
  - One operator controls multiple UVs
- Intentional/unintentional removal of assets
Project Key Research Challenges

• What is meant by the “ilities”?  
  – Definitions for survivability, evolvability, flexibility (changeability)

• How do you design for them?  
  – Design principles, empirical case examples

• How do you know you have them?  
  – Ilities metrics

• How do you decide how much ilities to add to system?  
  – Valuating ilities

• How do we do add ilities to the system during our architecting process?  
  – SAI Method
Defining the ilities for the Project

- **Survivability**: ability of a system to minimize the impact of a finite duration disturbance on value delivery.
  - Ex) Architecture 1 is minimally impacted to the Jamming perturbation and is therefore survivable to it.

- **Evolvability**: the ability of an architecture to be inherited and changed (e.g. using variation and selection) across generations [over time].
  - Ex) In the context of MarSec SoS, changing authority type or number of zones are evolvability-type changes. This means that they imply a change across two generations for the same SoS.

- **Flexibility**: ability of a system to be appropriately changed by a system-external change agent with intent.
  - Ex) Flexibility is a changeability sub-ility, which leads to consideration of design principles such as reconfigurability and margin (among others).
SoS Architecting with Ililities (SAI) Method

**Initial State of Practice**
Architecting with ililities as ad-hoc process with ililities considered later in lifecycle

**Post-research Practice**
Prescriptive method grounded in theory and empirical findings with ililities purposely designed into system architecture through design principles

Step 7

Novel analytic techniques and metrics evolved and developed

7 Analyze Architecture Alternatives

Step 7: Analyze Architecture Alternatives

2. Propose change execution strategies.
   a. Evaluate ility screening metrics.
   b. Select alternatives of interest.
   c. Complete Multi-Epoch Analysis
5. Collect set of alternatives of interest with ility metrics.
Design Principles for Utilities
Dual Approach to Developing Design Principles for Ilities

**Normative**
- Extract relevant ideas from literature, look for trends, forming basis of a theoretical model
- Explore candidate ility metrics and build on these to form a more comprehensive metric(s)
- Develop theory-based design principles based on analysis of applications of metric to cases

**Descriptive**
- Derive initial design principles using purported principles in literature and knowledge gathered from interviews
- Test validity of design principles by inductively mapping characteristics of existing systems to preliminary set
- Revise initial principles with insight gained from empirical test cases

This collection provides a means for determining which ilities are represented in different historical system changes, and to map those ilities to various implemented design principles.
### Model-based Testing of Design Principles

Using an Agent-based Discrete Event Simulation to Test Survivability DPs

<table>
<thead>
<tr>
<th>Design Principle</th>
<th>System</th>
<th>SoS</th>
<th>Enterprise</th>
<th>Testable*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preemption</td>
<td></td>
<td></td>
<td>Attacking pirates on land</td>
<td>No</td>
</tr>
<tr>
<td>Mobility</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Concealment</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Deterrence</td>
<td></td>
<td></td>
<td>Piracy Laws</td>
<td>Yes (Decoys)</td>
</tr>
<tr>
<td>Avoidance</td>
<td>Communications Relay Moving Locations</td>
<td>Smaller Vehicles</td>
<td>Decoys</td>
<td>Yes</td>
</tr>
<tr>
<td>Hardness</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Redundancy</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Margin</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Heterogeneity</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Distribution</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Fail-safe</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Containment</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Replacement</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Repair</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Deflection</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Defensive Posture</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Adaptation</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Stable Intermediate Forms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reversion</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
</tbody>
</table>

Evolvability Empirical Cases
Evolvability Empirical Case Studies

Research Goals:
1) further validate the usefulness of design principles for architecting systems that possess desirable lifecycle properties such as evolvability.
2) contribute real-world examples that architects may use to inspire specific design options for their system of interest.

**SOSE Case Selection Method**

**GOAL:** Identification of the collection of SoSs most appropriate* for purposes of evolvability study

*Appropriate means enabling the identification of at least 3 SoS examples for each evolvability design principle

<table>
<thead>
<tr>
<th>Candidate SoS</th>
<th>METRICS DEFINITION</th>
<th>CANDIDATE SoS EVALUATION</th>
<th>CANDIDATE COLLECTION COMPOSITION</th>
<th>CANDIDATE COLLECTION EVALUATION</th>
<th>CANDIDATE COLLECTION SELECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GCS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TBMCS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ABCS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BMDS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C2 Convergence</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FCS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MILSATCOM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NIFC-CA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SIAP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AOC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CAC2S</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DCGS-AF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DoDIIIS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TJTN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Define the metrics (and their levels) that will help assessing if candidate is good for the study

Assess all candidate SoSs in terms of defined metrics

Based on candidate SoSs’ assessments, generate different collections of 3 SoSs

Assess candidate SoS collections in terms of collection-level metrics

Based on candidate SoS collections’ assessments, select final collection
Design Principles to Change Options
Key Concepts

• The **change mechanism** is the **method** through which a system goes from state A to state B (e.g., *swapping payload on UAV*).

• The **path enabler** (i.e., a physical object, an action or a decision) is what gives the “option” of executing the change mechanism (e.g., *modular payload bay in original design*).

---

Research confirmed importance of considering ilities at three levels

---

SoS Program Architecture

Constituent System Architecture

SoS Architecture

“Guiding thoughts [for design] based on empirical deduction of observed behaviour or practices that prove to be true under most conditions over time” [Wasson, 2006]
Ballistic Missile Defense System (BMDS)

- Counters ballistic missiles of all ranges: short, medium, intermediate and long
- Aims at having interception capabilities at all phases of hostile missile flight (layered architecture): boost, midcourse, terminal

Three main categories of systems: sensors, interceptors, comms

BMDS composed of variety of sensors and systems:

Interceptors:
- Ground-based Midcourse Defense (GMD)
- PATRIOT Advanced Capability-3 (PAC-3)
- Terminal High Altitude Area Defense (THAAD)
- Aegis BMD Standard Missile 3

Detection and tracking sensors
- Space Tracking and Surveillance System (SSTS)
- Sea-Based X-band (SBX) Radar
- Aegis BMD SPY-1 Radar
- Early warning radar and forward-based radar

Command, Control, Battle Management, and Communications (C2BMC)

http://www.mda.mil
Before 2005, the BMDS had tracking capabilities only in the U.S. territory and in space.

In 2005, the Sea-Based X-Band (SBX) Radar was added to the BMDS to enhance detecting and tracking capabilities. The new BMDS featured a SBX Radar in the Pacific Ocean, which assisted operations in the Alaskan and Californian bases.

INTEGRABILITY

Ballistic Missile Defense Network (part of C2BMC)

integrating the SBX Radar in the BMDS

DECENTRALIZATION

The inclusion of SBX Radar

relocating tracking capabilities at various, appropriate locations

Decentralization

Distributing assets, capabilities and/or operations to appropriate multiple locations, rather than having them located in a single location. This entails locating components beyond other components’ physical spheres of influence.
The Sea-Based X-band (SBX) Radar was acquired by the US in 2003 from Moss Maritime, a Norwegian company that specializes in the construction of special purpose off-shore ships.

Between December 2007 and April 2008, the SBX Radar traveled more than 4,000 nautical miles across the Pacific Ocean.
# Studying Historical Cases Through Lens of Architect’s Intent

<table>
<thead>
<tr>
<th>Architect’s Intent</th>
<th>OPTIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Desire to add new constituent system</td>
<td>Modular space-based interceptor component allowed for swapping of single space-based constituent systems without repercussions in other constituent systems</td>
</tr>
<tr>
<td>Desire to add more of existing constituent system</td>
<td></td>
</tr>
<tr>
<td>Desire to replace or upgrade capabilities of existing constituent system</td>
<td></td>
</tr>
<tr>
<td>Desire to physically relocate resources/capabilities</td>
<td></td>
</tr>
<tr>
<td>Desire to change the way in which a function is performed</td>
<td></td>
</tr>
</tbody>
</table>

**Desire to Add New Constituent System**

**Desire to Add More of Existing Constituent System**

**Desire to Replace or Upgrade Capabilities of Existing Constituent System**

**Desire to Add More of Existing Function**

**Desire to Change the Way in Which a Function is Performed**

**Desire to Physically Relocate Resources/Capabilities**

---

**TBMCS-2013 Spiral-1**

Isolated applications allowed for implementing staged deployment of the upgraded IT to bundles of applications.
Quantifying Changeability for Analysis and Design Decisions

This is a “hard” problem and was one of the key technical thrusts in the recent DARPA META-II Program: Metrics for Adaptability

The DARPA META-II program goal is to substantially improve the design, manufacturing, and verification of complex cyber-physical systems, and particularly defense and aerospace systems.
Changes as Enabled Paths

• Conceptually, it can be helpful to think of change events as “paths” between design points

• Agent-Mechanism-Effect framework
  – Agent: instigator
  – Mechanism: mode of change
  – Effect: Δstate

• Option as Enabler-Mechanism
  – Path enabler: design feature
  – Mechanism: mode of change

• Intention for framework: think holistically about implementing changeability

Can we use this concept to create an approach for valuing changeability (the ability to access these paths), and to assist in the design process?

Valuating Changeability

- Challenges: Cost of changeability easier to quantify than benefit leading to insufficient investment; most existing approaches require probabilistic data and monetization of costs and benefits
- Key goal: reduce upfront assumptions to promote applicability to a wide range of problems and cases in technical design

Valuation Approach for Strategic Changeability (VASC)

System Model

EEA Model preferences / uncertainty

Strategy
Determines system response to possibility of each uncertainty

Multi-Epoch Analysis
across uncertainty space

Lifecycle (Era) Simulation

Era Analysis
with uncertainty ordering and system evolution

Five Steps

1. Set up Epoch-Era Analysis* (EEA)
2. Select Designs of Interest
3. Define Changeability Usage Strategies
4. Multi-Epoch Analysis
5. Era Simulation and Analysis

Multi-dimensional Value from Changeability

Value of changeability

- Avoid risk arising from uncontrollable uncertainty
- Seize opportunities to maximize performance value under different contexts / user preferences

Magnitude vs. Counting value

- Two sources – targeted alternatively by many previous metrics/methods
  - Magnitude: **amount** of value increase ➔ ROA, “Space” techniques
  - Counting: **number** of options ➔ Outdegree, some DSM metrics

Red: largest value increase (as measured by utility)
Blue: twice as many paths → redundancy in event of breakages, potentially useful in more contexts

Both of these types of value must be addressed in analysis.
Allows EEA to appropriately combine magnitude and counting value
- Each design/epoch pair is associated with at most one active change path
- Selected path is scored for its magnitude
- Counting value manifests in increased magnitude across epochs due to better options

Strategy encapsulates “value achieved only by executed changes” truism
Ex: Identified Adaptability-Enhancing Design Features in a Satellite System

For given system, defined possible change mechanisms that allow system to adapt to new missions

From "X-TOS" satellite system analysis

<table>
<thead>
<tr>
<th>Rule</th>
<th>Description</th>
<th>Change Enablers</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1: Plane Change</td>
<td>Increase/decrease inclination, decrease DV</td>
<td>Extra fuel</td>
</tr>
<tr>
<td>R2: Apogee Burn</td>
<td>Increase/decrease apogee, decrease DV</td>
<td>Extra fuel</td>
</tr>
<tr>
<td>R3: Perigee Burn</td>
<td>Increase/decrease perigee, decrease DV</td>
<td>Extra fuel</td>
</tr>
<tr>
<td>R4: Plane Tug</td>
<td>Increase/decrease inclination, requires &quot;tugable&quot;</td>
<td>Grappling point</td>
</tr>
<tr>
<td>R5: Apogee Tug</td>
<td>Increase/decrease apogee, requires &quot;tugable&quot;</td>
<td>Grappling point</td>
</tr>
<tr>
<td>R6: Perigee Tug</td>
<td>Increase/decrease perigee, requires &quot;tugable&quot;</td>
<td>Grappling point</td>
</tr>
<tr>
<td>R7: Space Refuel</td>
<td>Increase DV, requires &quot;refuelable&quot;</td>
<td>Refuelable tank</td>
</tr>
<tr>
<td>R8: Add Sat</td>
<td>Change all orbit, DV</td>
<td>Satellite spares</td>
</tr>
</tbody>
</table>

For a selected strategy "maintain greater than 40% mission utility over 15 years", identified features that enhanced adaptability

Adaptability-Enhancing Design Features

- Add grappling points to allow for space tug
- Carry extra fuel to allow for re-maneuvering
- None needed


© 2014 Massachusetts Institute of Technology

seari.mit.edu
More complete accounting of benefits of changeability for investment and design decisions

Design: 128 ▲ 256 ★ 384 ▼

<table>
<thead>
<tr>
<th></th>
<th>128</th>
<th>256</th>
<th>384</th>
</tr>
</thead>
<tbody>
<tr>
<td># of changeability features</td>
<td>0</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Initial cost for changeability</td>
<td>$0M</td>
<td>$272M</td>
<td>$544M</td>
</tr>
<tr>
<td>Average # missions completed</td>
<td>14.6</td>
<td>15.3</td>
<td>17.9</td>
</tr>
<tr>
<td>Avg. % deviation from cost efficiency</td>
<td>38.8</td>
<td>33.6</td>
<td>20.1</td>
</tr>
<tr>
<td>Avg. anticipated lifetime net benefit</td>
<td>$70B</td>
<td>$66B</td>
<td>$104B</td>
</tr>
</tbody>
</table>

N= 5000 alternative futures (per design)

Static tradespace shows apparent excessive cost for red or cyan designs.

VASC era analysis quantifies the cost tradeoff for additional lifetime value.

Changeability shown to be more available and utilized by the higher cost designs.

Quantification of Changeability Decision Options
Additional initial investment can result in increases in missions completed, efficiency, and net benefit.

Orbit transfer vehicle scenario simulates future missions sought for on-orbit realignment.

Max Profit Rule Usage

Cost

MAU

© 2014 Massachusetts Institute of Technology

seari.mit.edu
Further Quantifying Illities: Tradespace-derived Metrics

3 types of illities metrics:
1. Existence
2. Degree
3. Value

Changeability metrics can be applied to flexibility, adaptability, and evolvability

<table>
<thead>
<tr>
<th>Concept</th>
<th>Type</th>
<th>Acronym</th>
<th>Stands For</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epoch difficulty</td>
<td>Epoch</td>
<td>YN</td>
<td>Yield</td>
<td>Fraction of design space considered valid within an epoch</td>
</tr>
<tr>
<td>Degree of changeability</td>
<td>Epoch</td>
<td>OD</td>
<td>Outdegree</td>
<td># outgoing transition arcs from a design</td>
</tr>
<tr>
<td>Degree of changeability</td>
<td>Epoch</td>
<td>FOD</td>
<td>Filtered Outdegree</td>
<td>Above, considering only arcs below a chosen cost threshold</td>
</tr>
<tr>
<td>“Value” gap</td>
<td>Epoch</td>
<td>FPN</td>
<td>Fuzzy Pareto Number</td>
<td>% margin needed to include design in the fuzzy Pareto front</td>
</tr>
<tr>
<td>“Value” of a change</td>
<td>Epoch</td>
<td>FPS</td>
<td>Fuzzy Pareto Shift</td>
<td>Difference in FPN before and after transition</td>
</tr>
<tr>
<td>“Value” of a change</td>
<td>Epoch</td>
<td>ARI</td>
<td>Available Rank Increase</td>
<td># of designs able to be passed in utility via best possible change</td>
</tr>
<tr>
<td>Robustness via “no change”</td>
<td>Multi-Epoch</td>
<td>NPT</td>
<td>Normalized Pareto Trace</td>
<td>% epochs for which design is Pareto efficient in utility/cost</td>
</tr>
<tr>
<td>Robustness via “no change”</td>
<td>Multi-Epoch</td>
<td>tNPT</td>
<td>Fuzzy Normalized Pareto Trace</td>
<td>Above, with margin from Pareto front allow ed</td>
</tr>
<tr>
<td>Robustness via “change”</td>
<td>Multi-Epoch</td>
<td>eNPT, efNPT</td>
<td>Effective (Fuzzy) Normalized Pareto Trace</td>
<td>Above, considering the design’s end state after transitioning</td>
</tr>
<tr>
<td>“Value” of a change across epochs</td>
<td>Multi-Epoch</td>
<td>FPS Dist</td>
<td>Fuzzy Pareto Shift Distribution</td>
<td>Epoch frequency of FPS scores for a design across epochs</td>
</tr>
<tr>
<td>Survivability value</td>
<td>Era</td>
<td>TAUL</td>
<td>Time-weighted Average Utility Loss</td>
<td>Difference between baseline utility and time-weighted average utility</td>
</tr>
<tr>
<td>Survivability timing</td>
<td>Era</td>
<td>AT</td>
<td>Threshold Availability</td>
<td>Ratio of time above critical value thresholds to design life</td>
</tr>
</tbody>
</table>
Applying the Ilities Metrics

No one solution was dominant, so must trade off ilities through alternatives

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Stands For</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>NPT</td>
<td>Normalized Pareto Trace</td>
<td>% epochs for which design is Pareto efficient in utility/cost</td>
</tr>
<tr>
<td>sNPT</td>
<td>Fuzzy Normalized Pareto Trace</td>
<td>Above, with margin from Pareto front allowed</td>
</tr>
<tr>
<td>eNPT</td>
<td>Effective (fuzzy) Normalized Pareto Trace</td>
<td>Above, considering the design’s end state after transitioning</td>
</tr>
<tr>
<td>FPS</td>
<td>Fuzzy Pareto Shift</td>
<td>Difference in FPS before and after transition</td>
</tr>
<tr>
<td>FOD</td>
<td>Filtered Outdegree</td>
<td>Above, considering only arcs below a chosen cost threshold</td>
</tr>
<tr>
<td>TAUL</td>
<td>Time Weighted Avg Utility Loss</td>
<td>Integral of utility loss over time</td>
</tr>
<tr>
<td>-</td>
<td>Accumulated Utility vs. Discounted Cost</td>
<td>Lifecycle cost benefit</td>
</tr>
</tbody>
</table>

Design 49 of Architecture 1 scores well in changeability, affordability, survivability, and robustness metrics; change mechanisms 1, 2, and 4 must be available

<table>
<thead>
<tr>
<th>Design ID</th>
<th>Change, Afford, Survive, Robust</th>
<th>Hermes</th>
<th>Shadow</th>
<th>Prop</th>
<th>Helo</th>
<th>Boats</th>
<th>Task</th>
<th>Operators</th>
<th>Spare Helos</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>2,0,1,1, Robust</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>Multi</td>
<td>2:1</td>
<td>2</td>
</tr>
<tr>
<td>49</td>
<td>2,2,1,1</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>4</td>
<td>Dedi</td>
<td>2:1</td>
<td>1</td>
</tr>
<tr>
<td>1729</td>
<td>1,2,1,1</td>
<td>2</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>Dedi</td>
<td>2:1</td>
<td>2</td>
</tr>
<tr>
<td>3505</td>
<td>1,0,1,1</td>
<td>2</td>
<td>6</td>
<td>0</td>
<td>1</td>
<td>4</td>
<td>Dedi</td>
<td>2:1</td>
<td>1</td>
</tr>
</tbody>
</table>

These metrics allowed for the trade off of particular ilities with performance and cost
Synthesizing the Research
SoS Architecting for Ilities (SAI) Method

Operational Needs

1. Determine value props and constraints
   - 6.1 System Taxonomy
     - MAUA/AHP
2. Identify potential perturbations
   - 6.2 Perturbation Taxonomy
3. Identify initial desired ilities
   - 6.3 Ilities Hierarchy
   - 6.4 Ilities Statement Generator
4. Generate initial architecture alternatives
   - 6.1 System Taxonomy
   - 6.5 Generate CONOPS Trades
5. Generate ility-driving options
   - 6.7 Design Principles
   - 6.8 Change Options Framework
   - 6.9 System Examples Database
   - 6.10 Cause-Effect Mapping
   - 6.11 Multi-Epoch Analysis
   - 6.12 VASC/ESF
   - 6.13 Ilities Metrics
6. Evaluate potential alternatives
   - Modeling/Simulation
   - Value Metrics/AHP
7. Analyze architecture alternatives
8. Trade-off and select “best” architecture(s) with ilities

Supporting method, process or tool

Technical appendix number

© 2014 Massachusetts Institute of Technology
Impact on Practice

*With SEAr’s research, [our agency] now has a systematic approach to design robust and evolvable systems of systems*

Project Sponsor

Academic Impact

- 24 publications
- 2 best paper awards
- 3 master theses
- 1 doctoral thesis
- Dataset and discrete event simulation supporting research
- New analysis techniques
Special Thanks to Our Students

**Current SEAri Graduate Students**

- Michael Curry
- Matthew Fitzgerald
- Paul Grogan*
- Paul La Tour
- Alexander Pina
- Benjamin Putbrese
- Nicola Ricci
- Michael Schaffner
- Marcus Wu
- Li Qian Yeong
- Hunter Zhao

*SEAri affiliated student