Motivation

Most DoD programs operate in a context of a large complex SoS
Want to take advantage of cutting edge MBE techniques, but it is not clear how well MBE scales to address SoS problems

▪ Approach:
  – Assess and improve return on investment (ROI) from MBE for SoS problems
    ▪ Software techniques
    ▪ Analysis methods

▪ Key Findings:
  – Base model - Kill-chain architecture that can be used as a template for modeling SoS in MBE environments
  – Robustness analysis - Results validated using multiple notional alternative architectures by EIMS
  – Software Solutions - Qualitative and quantitative data collection ongoing

Increase benefit by developing analysis techniques to answer crucial SoS-level questions
Software Methods
Technical Approach: Inheritable Architectures

Abstract

SOA

Avionics Service Bus

Enables Model Re-use corresponding to different architecture patterns
Base Model Architecture

- **Base/Derivative Model Framework**
  - Base Model captures key functional SoS architecture
  - Derivative model represent domain-specific behavior

- **This approach helps:**
  - Accelerate domain model development via Base Model reuse
  - Rapidly evaluate different options utilizing predefined stereotypes and analysis engines
  - Iterative design to continuously refine common SoS functions
Base Model: High Level Structure
Base Model: Inheritance Structure

Inheritable and reusable Statecharts

Statechart for Device
BASE Model: Inheritable Types

BASE
- Operations (i.e. functions)
  - processSignals()
- Attributes (i.e. metrics)
  - MaxRange

DERIVATIVE
(e.g. CDMaST)
- Towed Array Sonar Sensor
- Bow Sonar Sensor
- ISAR
Base Model CSV Importer

Base Model

10 Node Scenario
100 Node Scenario

MBE Utility to reduce development effort associated with modeling large SoS complex networks
CSV Importer Utility

Run CSV Importer Utility to automatically generate model/JMS Pub/Sub Architecture

Conceptualize SoS Architecture

Add Connectivity Framework

<table>
<thead>
<tr>
<th>PUBLISHERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUBSCRIBERS</td>
</tr>
<tr>
<td>S</td>
</tr>
<tr>
<td>AOC</td>
</tr>
<tr>
<td>C2</td>
</tr>
<tr>
<td>Tanker</td>
</tr>
<tr>
<td>Fighter</td>
</tr>
<tr>
<td>UAS</td>
</tr>
</tbody>
</table>

© 2018 The MITRE Corporation. All rights reserved.
Base Model GUI

- A MATLAB GUI has been built to simplify the process of populating a connectivity matrix
- The tool outputs a CSV file that can then be imported into the architecture model
Demonstration
Metrics – Experiments

▪ **Qualitative**
  – **Experiment 1**: Give the base model to MITRE employees to use on their projects as they see fit. Collect feedback.
    ▪ Likes, dislikes, pain points, time savings estimates, description of use case, experience level
    ▪ Time Cost: 30 min interview

▪ **Quantitative**
  – **Experiment 2**: Give MITRE employees a sample coms network and have them create it by hand and by using the CSV importer
    ▪ Networks of different sizes
    ▪ Measure time to complete exercise
    ▪ Time Cost: Approx. 45 min per data point
  – **Experiment 3**: Randomized control trial with ~20 new interns
    ▪ Group A: Create reference model from scratch
    ▪ Group B: Create reference model using base model
Metrics – Experiment 1 Results

- **Project 1:**
  - 3 reviewers
  - Not adopted

- **Feedback:**
  - “…This base model would be a great reference, e.g., utilizing the package structure framework used, with the inheritable architectures and the focus on reuse.”
  - “…We expect to draw ideas from it as we build our own model.”
  - “We intend to focus more on activity diagrams than state charts.”
  - “Our project is not in the context of the Air Force, so we would have to change the block and activity names.”
  - “Overall it is not a good fit for [our project].”

- **Project 2:**
  - 1 reviewer
  - Adopted

- **Feedback:**
  - **Qualitative**
    Base Model state charts look too “in-depth”, “specific”, need to take a closer look to see if they will work for my use case. But if they work, “that would be awesome”, it will save tons of time.
  
  - **Pseudo - Quantitative**
    Estimated time savings of 40 hours on work completed so far.
  
  - **Update**
    Base Model has proven a good fit for project and has been used extensively.
The Scenario
This is a hypothetical Air Force kill-chain scenario consisting of 1 ground control station (AOC), 1 air command and control (C2), 4 Fighter Jets, 4 Unmanned Aircraft Systems (UASs), and 1 Tanker.

- AOC needs to be able to communicate with C2, since C2 alerts AOC when there is a threat and then gets its orders from the ground.
- C2 also needs to be able to communicate with all fighters and the Tanker during the mission.
- Also, all fighters and UASs need to be able to communicate with the Tanker, since they'll occasionally need to refuel during flight.
- Every fighter needs to be able to communicate with every other fighter, and every UAS needs to be able to communicate with every other UAS.
- Moreover, every fighter should be able to communicate with every UAS, and vice versa.

You may assume all communication channels are bi-directional (any communication matrix you set up should be symmetric with respect to rows and columns).

Time to model 11-node scenario with and without CSV tool

<table>
<thead>
<tr>
<th>Participant Number</th>
<th>Minutes to complete task</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>With tool: 10, Without tool: 15</td>
</tr>
<tr>
<td>2</td>
<td>With tool: 11, Without tool: 16</td>
</tr>
<tr>
<td>3</td>
<td>With tool: 12, Without tool: 17</td>
</tr>
<tr>
<td>4</td>
<td>With tool: 13, Without tool: 18</td>
</tr>
<tr>
<td>5</td>
<td>With tool: 14, Without tool: 19</td>
</tr>
<tr>
<td>6</td>
<td>With tool: 15, Without tool: 20</td>
</tr>
</tbody>
</table>

**Time savings**

Mean: 39%
Standard Dev: 12%
Metrics – Experiment 2 Results

Time to model 29-node scenario with and without CSV+GUI tool

- **Time savings**
  - Mean: 63%
  - Standard Dev: 14%

- **Average mistakes**
  - Without tool: 9.2
  - With tool: 0.8
Analytic Methods
Motivation

1. Programs need help making decisions about changes to their existing architectures

2. What is the baseline architecture? What is the baseline performance?

3. How can new solutions be integrated?

4. How will the changes affect performance?

Performance vs. Cost

© 2018 The MITRE Corporation. All rights reserved.

As-is for Evaluating Architectures
SoS Analysis of Alternatives

- Establish baseline SoS architecture
- SoS graph abstraction and network analysis
- Generate SoS architecture alternatives
- MIP contribution: inform prioritization of alternatives using lightweight analytics
- Detailed evaluation with M&S environment
- Informed architecture selection

M&S Environment

- Physics
- Operations
Robustness Metric (Algebraic Connectivity Value)

- Represents average difficulty of isolating a node
  - Second smallest eigenvalue of a Laplacian Matrix

- Inputs:
  - Degree Matrix
    - Diagonal matrix that contains the number of nodes adjacent to a given node
      \[
      D_{ij} = \begin{cases} 
      d_i & \text{degree of component } i \text{ when } i = j \\
      0 & \text{otherwise}
      \end{cases}
      \]
  - Adjacency Matrix
    - Symmetric matrix that contains a 1 if two given nodes are adjacent and 0 otherwise
      \[
      A_{ij} = \begin{cases} 
      1 & \forall [(i, j) | (i \neq j) \text{ and } (i, j) \in \Delta] \\
      0 & \text{otherwise}
      \end{cases}
      \]

Identifying Robust SoS Architectures

Architecture 1

Robustness Metric Value: 0.5858

Architecture 2

Robustness Metric Value: 0.8299
Multi-layer Architecture Analysis
Example Architecture

Available Communication Methods

- Link 16
- SATCOM
- HF Radio
- VHF Radio

Weapon (1)

C2 (2)

CO (3)

Sensor (4)
Mapping Architecture to Multilayer Graph – Intralayer Graph Representation

- **Available Communication Methods**
  - **Weapon (1)**: Link 16, SATCOM, HF Radio, VHF Radio
  - **C2 (2)**: Link 16, SATCOM, HF Radio, VHF Radio, Link 11
  - **CO (3)**: Link 16, SATCOM, Link 11
  - **Sensor (4)**: Link 16, SATCOM, HF Radio, VHF Radio, Link 11

**Nodes:**
- Link 16
- VHF Radio
- HF Radio
- SATCOM
- Link 11

**Edges:**
- Communication methods between the nodes.
Mapping Architecture to Multilayer Graph – Intralayer Adjacency Representation

[Diagram showing intralayer adjacency for VHF Radio, HF Radio, SATCOM, and Link 11, Link 16 with adjacency matrices for each layer.]

© 2018 The MITRE Corporation. All rights reserved.

### Mapping Architecture to Multilayer Graph – Intralayer Adjacency Representation

<table>
<thead>
<tr>
<th></th>
<th>VHF Radio</th>
<th>HF Radio</th>
<th>Link 11</th>
<th>Link 16</th>
<th>SATCOM</th>
</tr>
</thead>
<tbody>
<tr>
<td>VHF Radio</td>
<td><img src="image1" alt="Image" /></td>
<td><img src="image2" alt="Image" /></td>
<td><img src="image3" alt="Image" /></td>
<td><img src="image4" alt="Image" /></td>
<td><img src="image5" alt="Image" /></td>
</tr>
<tr>
<td>HF Radio</td>
<td><img src="image6" alt="Image" /></td>
<td><img src="image7" alt="Image" /></td>
<td><img src="image8" alt="Image" /></td>
<td><img src="image9" alt="Image" /></td>
<td><img src="image10" alt="Image" /></td>
</tr>
<tr>
<td>Link 11</td>
<td><img src="image11" alt="Image" /></td>
<td><img src="image12" alt="Image" /></td>
<td><img src="image13" alt="Image" /></td>
<td><img src="image14" alt="Image" /></td>
<td><img src="image15" alt="Image" /></td>
</tr>
<tr>
<td>Link 16</td>
<td><img src="image16" alt="Image" /></td>
<td><img src="image17" alt="Image" /></td>
<td><img src="image18" alt="Image" /></td>
<td><img src="image19" alt="Image" /></td>
<td><img src="image20" alt="Image" /></td>
</tr>
<tr>
<td>SATCOM</td>
<td><img src="image21" alt="Image" /></td>
<td><img src="image22" alt="Image" /></td>
<td><img src="image23" alt="Image" /></td>
<td><img src="image24" alt="Image" /></td>
<td><img src="image25" alt="Image" /></td>
</tr>
</tbody>
</table>
Mapping Architecture to Multilayer Graph – Interlayer Matrix Representation

HF Radio

SATCOM

SATCOM

HF Radio

Weapon (1)

C2 (2)

CO (3)

Sensor (4)
Mapping Architecture to Multilayer Graph – Adding Interlayer to Intralayer in Matrix

<table>
<thead>
<tr>
<th>VHF Radio</th>
<th>HF Radio</th>
<th>Link 11</th>
<th>Link 16</th>
<th>SATCOM</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 1 1 1 0 0 0 0 1 0 0 0 1 0 0</td>
<td>1 0 0 1 0 0 0 1</td>
<td>0 1 0 0 0 1 0 0 0</td>
<td>0 1 0 0 0 0 1</td>
<td></td>
</tr>
<tr>
<td>1 0 0 1 0 0 0 1 1 0 0 0</td>
<td>0 1 1 0 0 0 0 0 1 0 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td></td>
</tr>
<tr>
<td>0 1 1 0 0 0 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td></td>
</tr>
<tr>
<td>0 0 0 0 1 0 1 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td></td>
</tr>
<tr>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td></td>
</tr>
<tr>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td></td>
</tr>
<tr>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td></td>
</tr>
<tr>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td></td>
</tr>
<tr>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td>0 0 0 0 0 1 0 0 0 1 0 0</td>
<td></td>
</tr>
</tbody>
</table>

Diagrams showing multilayer connections for VHF, HF, Link 11, Link 16, and SATCOM.
Summary

▪ Results
  – Developed a scalable rapid analysis capability for MBSE software tools
  – Identified a proxy for resilience that can be measured using lightweight analysis techniques
  – Tested the analysis method on notional architectures and compared the results with a low fidelity operational modeling and simulation tool

▪ Lessons Learned
  – Detailed analysis will have to accompany the graph theoretic analysis to account for operationally critical architectural components
  – Based on the domain the optimal graph theoretic value may vary