Model-Based Systems Engineering Tradespace Analysis with SysML and COSYSMO

Sponsor: DASD(SE)

By
David Jacques, Air Force Institute of Technology
Ray Madachy, Naval Postgraduate School
Kristin Giammarco, Naval Postgraduate School
CPT Dennis Edwards, United States Military Academy, West Point

9th Annual SERC Sponsor Research Review
November 8, 2017
FHI 360 CONFERENCE CENTER
1825 Connecticut Avenue NW, 8th Floor
Washington, DC 20009

www.sercuarc.org
Agenda

- Introduction
- SysML and COSYSMO Integration
- Case Study: Remote Targeting System UAS
- Conclusions and Future Work
Overview

• Total Ownership Cost (TOC) modeling to enable affordability tradeoffs with otherilities
  — Integrated costing of systems, software, hardware and human factors across full lifecycle operations
  — Combine with other MBSE architecture-based behavior and performance analysis

• Current shortfalls forilities tradespace analysis
  — Models/tools are incomplete wrt/ TOC phases, activities, disciplines, SoS aspects
  — No integration with physical design space analysis tools, system modeling, or each other

• Cost estimation can be improved by using the same architectural definitions for cost model inputs, without the need for independent cost modeling expertise and effort expenditure.

• Developing translation rules and constructs between MBSE methods, performance analysis and cost model inputs.

• Demonstrating tool interoperability and tailorability
Case Study Method

- Use various MBSE methods and tools to evaluate behavior and performance analysis in the face of requirements changes and System of System (SoS) architectural variations.

- Develop operational and system architectures to capture sets of UAS military scenarios for cooperative swarms with 3 UAS group sizes.

- Transition the architectures to MBSE environments.
  - SysML diagrams and executable activity models using Innoslate, CORE

- Develop cost model interfaces for components of the architectures in order to evaluate cost effectiveness in an uncertain future environment.
  - XML model files parsed automatically to extract cost model inputs

- Design and demonstrate UAS ISR tradespace including cost in integrated MBSE environment with executable models of architectures.
Example UAS Missions

- Single UAS Search and Target Tracking (Simple Mission)
- UAS Pair Search and Target Tracking
- Find, Fix and Finish Terrorist Leadership (1)
- Find, Fix and Finish Terrorist Leadership (2)
- Mobile Missile Launcher Monitoring (1)
- Mobile Missile Launcher Monitoring (2)
Single UAS Simple Mission Threads

• Launch
• Navigation and flight
• Search and target ID including evaluation
• Target tracking
• Return/recovery

• Enumeration of these in MBSE models constitutes primary size input for Constructive Systems Engineering Cost Model (COSYSMO)
Example Activity Model (OV-5b) for Two UAS Mission
UAV Mission
Nominal Cost Comparisons

Relative System Size/Cost

# Operational Scenarios

Mission Baselines

Single UAV (Simple)  UAV Pair  Find, Fix and Finish Terrorist Leadership (1)  Find, Fix and Finish Terrorist Leadership (2)  Mobile Missile Launcher Monitoring (1)  Mobile Missile Launcher Monitoring (2)
Agenda

• Introduction
• SysML and COSYSMO Integration
• Case Study: Remote Targeting System UAS
• Conclusions and Future Work
## COSYSMO Size Inputs

<table>
<thead>
<tr>
<th>Size Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements</td>
<td>The number of requirements for the system-of-interest at a specific level of design. Requirements may be functional, performance, feature, or service-oriented.</td>
</tr>
<tr>
<td>Interfaces</td>
<td>The number of shared physical and logical boundaries between system components or functions (internal interfaces) and those external to the system (external interfaces).</td>
</tr>
<tr>
<td>Algorithms/Blocks</td>
<td>The number of newly defined or significantly altered functions that require unique mathematical algorithms to be derived in order to achieve the system performance requirements.</td>
</tr>
<tr>
<td>Operational</td>
<td>Operational scenarios that a system must satisfy, including nominal and off-nominal threads.</td>
</tr>
<tr>
<td>Scenarios (Threads)</td>
<td></td>
</tr>
</tbody>
</table>

SySML to COSYSMO Mapping

SysML

Package Diagram
Requirements Diagram
<<requirement>>

Internal Block Diagram
Block Definition Diagram
port

Parametric Diagram
Block Definition Diagram
+

Use Case Diagram
use case

COSYSMO

System Size

# Requirements

# Interfaces

# Algorithms

# Operational Scenarios (Threads)
...<entity id="e4RWJH">
    <name>Flight Activation</name>
    <description/>
    <hidden>false</hidden>
    <locked>false</locked>
    <schemaClassId>C1</schemaClassId>
    <number/>
    <doubleAttribute schemaPropertyId="P4">
        <doubleValue>0.0</doubleValue>
    </doubleAttribute>
    <durationAttribute schemaPropertyId="P2">
        <doubleValue>1.0</doubleValue>
        <units>HOURS</units>
    </durationAttribute>
    <labelId>L60</labelId>
    <simulationData>
        <type>SERIAL</type>
        <controlStructure id="57bcf94d-cb13-42e5-9e5b-72dfd2a9fc17">
            <type>START</type>
            <successorStructure id="fe494a8a-d37b-44d1-9ffd-614416e6111e">
                <type>END</type>
            </successorStructure>
        </controlStructure>
    </simulationData>
...
## Example COSYSMO Estimate

### SysML COSYSMO

**System Size** | Input Method | File Input | Select Input File | `distiller.xml`
---|---|---|---|---
# of System Requirements | Easy | Nominal | Difficult |
| 28 | 2 | 1 |
| 29 | 2 | 1 |
| 3 | | |
| 1 | | |

### System Cost Drivers

- **Requirements Understanding**
- **Architecture Understanding**
- **Level of Service Requirements**
- **Migration Complexity**
- **Technology Risk**

**Maintenance** | Off

### System Labor Rates

Cost per Person-Month (Dollars) | $10000

**Calculate**

### Results

**Systems Engineering**

- Effort = 25.6 Person-months
- Schedule = 4.4 Months
- Cost = $255525

---

SSRR 2017

November 8, 2017
Agenda

- Introduction
- SysML and COSYSMO Integration
- Case Study: Remote Targeting System UAS
- Conclusions and Future Work
Remote Targeting System (RTS)

Background

• Comparing two architecture variants of Remote Targeting System (RTS) performed by semi-autonomous vehicles

• Baseline variant can have multiple vehicles, but uses human-in-the-loop to declare targets
  — Requires data link back to operator for each vehicle

• Autonomous Target Recognition (ATR) variant has heterogeneous sensors, and can use multiple vehicles to auto confirm target declarations without requiring a human.
  — Vehicle needs an autonomous target recognition algorithm (ATR)
  — Vehicle requires a data link between vehicles, in addition to the data link back to the human operator (which must be modified to accommodate the ATR declarations);
  — The Plan Mission Use Case must be modified to include loading of target templates
  — Search Use Case must be modified to accommodate the ATR activities
  — Additional <<include>> Use Cases, “Perform ATR” and “Confirm Target” must be added
  — New and modified requirements must be accommodated

• Both variants are modeled in SysML from which cost estimates are directly derived
Remote Targeting System (RTS) Scenarios
(Use Cases)

- Egress and Recovery
- Launch and Ingress
- Perform Setup
- Perform Surveillance
- Self Destruct
- Plan Mission
- Use Case
- Perform Search

**Actors:**
- Ground Operator
- Off Board C2 Operator
- Target
- GPS

Date: December 3, 2016
RTS Requirements

Command Destruct
- The System shall be capable of executing a Command Destruct during any portion of mission flight. The Co...

Inventory
- The RTS shall include all necessary ground and flight equipment to successfully execute all required functi...

MP Plan/Re-plan
- The Operator shall be able to initiate a new or modified Mission Plan during pre-launch, ingress, search,

Return-to-Launch (RTL)
- The System shall be capable of executing a Return-to-Launch (RTL) during any portion of mission flight.

Remote Targeting System
- The System shall be capable of executing a Command Destruct during any portion of mission flight. The Co...

Remote Targeting System
- The System shall be capable of executing a Command Destruct during any portion of mission flight. The Co...

GS Send Cnd (RTL)
- The System shall be capable of executing a Return-to-Launch (RTL) during any portion of mission flight.

GS Send Cnd (RTL)
- The System shall be capable of executing a Return-to-Launch (RTL) during any portion of mission flight.

A/V Rcv Cnd (RTL)
- The System shall be capable of executing a Return-to-Launch (RTL) during any portion of mission flight.

Remote Targeting System
- The System shall be capable of executing a Command Destruct during any portion of mission flight. The Co...

MP Plan/Re-plan
- The Operator shall be able to initiate a new or modified Mission Plan during pre-launch, ingress, search,

Inventory
- The RTS shall include all necessary ground and flight equipment to successfully execute all required functi...

Command Destruct
- The System shall be capable of executing a Command Destruct during any portion of mission flight. The Co...

Remote Targeting System
- The System shall be capable of executing a Command Destruct during any portion of mission flight. The Co...

A/V Rcv Cnd (RTL)
- The System shall be capable of executing a Return-to-Launch (RTL) during any portion of mission flight.

GS Send Cnd (RTL)
- The System shall be capable of executing a Return-to-Launch (RTL) during any portion of mission flight.
Requirements Decomposition
Level for Costing

Air Vehicle Requirements

Waypoint
The Air Vehicle shall be capable of GPS navigation.

Flight Mode
The Air Vehicle shall be capable of receiving commands to change flight modes.

Mission Plan Reception
The Air Vehicle shall accept Mission Plan from the Ground Station at all times once communication

Function Check Result
The Air Vehicle shall transmit results of function check to the Ground Station.

Mission Plan Success
The Air Vehicle shall send receipt of Mission Plan to the Ground Station.

Telemetry
The Air Vehicle shall be capable of telemetry from the Ground Station.
Perform Surveillance

**Description:** This Use Case covers surveillance activities

**Preconditions:** Target has been identified and Air Vehicle has entered Surveillance mode

**Primary Flow:**

1. Air Vehicle transmits telemetry to Ground Station(s)
2. Ground Station(s) receives and displays flight data
3. Ground Station(s) stores telemetry data
4. Air Vehicle loiters over target
5. Air Vehicle continues video transmission to Ground Station and Off-Board C2
6. Ground Station(s) receives and displays video transmission
7. Operator and Off-Board C2 monitor video and flight data
8. Ground station(s) calculate target coordinates based on video and telemetry
9. Ground station(s) displays target coordinates
10. Operator initiates RTL
11. Ground Station sends RTL command to Air Vehicle
12. Air Vehicle enters RTL mode

**Alternate Flow:** At any time:

a. If bad vehicle health, Operator enters RTL command on Ground Station
b. Ground Station sends RTL command to Air Vehicle
c. Air Vehicle enters RTL mode

**At any time:**

a. Operator initiates <<include>> Plan Mission Use Case
b. Vehicle ingresses to new Search Insertion point

**At any time:**

a. If vehicle compromise is evident, execute <<include>> Self Destruct Use Case

**Postconditions:** Air Vehicle is loitering over the target for > 10 minutes and target coordinates are calculated and displayed on Ground Station(s); Air Vehicle enters RTL mode
RTS Interfaces (Ports) (1/2)

ibd Remote Targeting System

GPS Link: GPS Signal
Video Link: Video data
Telemetry Link: Telemetry
Command Link: Cmd (BIT), Cmd (Launch), Cmd (RTL), Cmd (Self Destruct), Cmd (Surveillance), MP

ibd Ground Station

Data Modem: Data Modem
USB

Laptop: Laptop
MP S/W API
Video S/W API
Video Receiver

Video S/W API

Mission Control System

Video Processing System

University Edition - For Academic Use Only
Date: December 3, 2016
Constructive Systems Engineering Cost Model (COSYSMO)

System Size

<table>
<thead>
<tr>
<th></th>
<th>Easy</th>
<th>Nominal</th>
<th>Difficult</th>
</tr>
</thead>
<tbody>
<tr>
<td># of System Requirements</td>
<td>31</td>
<td></td>
<td></td>
</tr>
<tr>
<td># of System Interfaces</td>
<td>25</td>
<td></td>
<td></td>
</tr>
<tr>
<td># of Algorithms</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td># of Operational Scenarios</td>
<td>7</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

System Cost Drivers

- Requirements Understanding
- Architecture Understanding
- Level of Service Requirements
- Migration Complexity
- Technology Risk

Maintenance: Off

System Labor Rates
Cost per Person-Month (Dollars): 10000

Calculate

Results
Systems Engineering
Effort = 70.4 Person-months
Schedule = 6.1 Months
Cost = $703715

Total Size = 201 Equivalent Nominal Requirements

Acquisition Effort Distribution (Person-Months)

<table>
<thead>
<tr>
<th>Phase / Activity</th>
<th>Conceptualize</th>
<th>Develop</th>
<th>Operational Test and Evaluation</th>
<th>Transition to Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acquisition and Supply</td>
<td>1.4</td>
<td>2.5</td>
<td>0.6</td>
<td>0.4</td>
</tr>
<tr>
<td>Technical Management</td>
<td>2.6</td>
<td>4.5</td>
<td>3.0</td>
<td>1.8</td>
</tr>
<tr>
<td>System Design</td>
<td>7.2</td>
<td>8.4</td>
<td>3.6</td>
<td>1.9</td>
</tr>
<tr>
<td>Product Realization</td>
<td>1.4</td>
<td>3.2</td>
<td>3.4</td>
<td>2.6</td>
</tr>
<tr>
<td>Product Evaluation</td>
<td>3.9</td>
<td>5.9</td>
<td>8.7</td>
<td>3.3</td>
</tr>
</tbody>
</table>
### Constructive Systems Engineering Cost Model (COSYSMO)

#### System Size

<table>
<thead>
<tr>
<th>Equivalent Nominal Requirements</th>
<th>Distribution</th>
<th>Min</th>
<th>Most Likely</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements Understanding</td>
<td>Nominal</td>
<td>161</td>
<td>201</td>
<td>302</td>
</tr>
<tr>
<td>Architecture Understanding</td>
<td>Nominal</td>
<td>161</td>
<td>201</td>
<td>302</td>
</tr>
<tr>
<td>Level of Service Requirements</td>
<td>Nominal</td>
<td>161</td>
<td>201</td>
<td>302</td>
</tr>
<tr>
<td>Migration Complexity</td>
<td>Nominal</td>
<td>161</td>
<td>201</td>
<td>302</td>
</tr>
<tr>
<td>Technology Risk</td>
<td>Nominal</td>
<td>161</td>
<td>201</td>
<td>302</td>
</tr>
<tr>
<td>Maintenance</td>
<td>Off</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### System Labor Rates

Cost per Person-Month (Dollars): 10,000

#### Results

**Systems Engineering**
- Effort: 70.1 Person-months
- Schedule: 7.1 Months
- Cost: $700,000

Total Size = 201 Equivalent Nominal Requirements

#### Acquisition Effort Distribution (Person-Months)

<table>
<thead>
<tr>
<th>Phase / Activity</th>
<th>Conceptualize</th>
<th>Develop</th>
<th>Transition to Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acquisition and Supply</td>
<td>1.4</td>
<td>2.5</td>
<td>0.6</td>
</tr>
<tr>
<td>Technical Management</td>
<td>2.6</td>
<td>4.5</td>
<td>3.0</td>
</tr>
<tr>
<td>System Design</td>
<td>7.1</td>
<td>8.4</td>
<td>3.6</td>
</tr>
<tr>
<td>Product Realization</td>
<td>1.4</td>
<td>3.2</td>
<td>3.4</td>
</tr>
<tr>
<td>Product Evaluation</td>
<td>3.9</td>
<td>5.9</td>
<td>8.7</td>
</tr>
</tbody>
</table>

#### Acquisition Monte Carlo Results

**Systems Engineering Effort Distribution Function**

<table>
<thead>
<tr>
<th>Effort (PLI)</th>
<th>55-64</th>
<th>64-72</th>
<th>72-81</th>
<th>81-90</th>
<th>90-99</th>
<th>99-107</th>
</tr>
</thead>
<tbody>
<tr>
<td># Iterations</td>
<td>92</td>
<td>280</td>
<td>194</td>
<td>110</td>
<td>36</td>
<td></td>
</tr>
</tbody>
</table>

**Systems Engineering Effort Confidence Levels**

<table>
<thead>
<tr>
<th>Confidence Level</th>
<th>10%</th>
<th>20%</th>
<th>30%</th>
<th>40%</th>
<th>50%</th>
<th>60%</th>
<th>70%</th>
<th>80%</th>
<th>90%</th>
<th>100%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effort (PLI)</td>
<td>64</td>
<td>68</td>
<td>70</td>
<td>74</td>
<td>77</td>
<td>80</td>
<td>83</td>
<td>87</td>
<td>94</td>
<td>107</td>
</tr>
</tbody>
</table>

[http://csse.usc.edu/tools/59dffb2d69d30](http://csse.usc.edu/tools/59dffb2d69d30)
## Extrapolation for RTS

### Full Program Cost Distribution

#### Input Table

<table>
<thead>
<tr>
<th>Variable</th>
<th>Units</th>
<th>Optimistic</th>
<th>Expected</th>
<th>Pessimistic</th>
</tr>
</thead>
<tbody>
<tr>
<td>SE Hours</td>
<td>Hours</td>
<td>9476</td>
<td>10652</td>
<td>16284</td>
</tr>
<tr>
<td>SE Factor</td>
<td>Hours / Hour</td>
<td>0.150</td>
<td>0.125</td>
<td>0.100</td>
</tr>
<tr>
<td>Labor</td>
<td>$ / Hour</td>
<td>100</td>
<td>110</td>
<td>125</td>
</tr>
<tr>
<td>Materials</td>
<td>$M</td>
<td>2</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>Travel</td>
<td>Percentage</td>
<td>2.5%</td>
<td>3.5%</td>
<td>6.0%</td>
</tr>
<tr>
<td>G&amp;A</td>
<td>Percentage</td>
<td></td>
<td>10%</td>
<td></td>
</tr>
<tr>
<td>ODC</td>
<td>Percentage</td>
<td></td>
<td>10.0%</td>
<td></td>
</tr>
<tr>
<td>Fee</td>
<td>Percentage</td>
<td></td>
<td>10.0%</td>
<td></td>
</tr>
<tr>
<td>MR</td>
<td>Percentage</td>
<td></td>
<td>10.0%</td>
<td></td>
</tr>
<tr>
<td>Calibration Factor</td>
<td>No Units</td>
<td></td>
<td>1.3</td>
<td></td>
</tr>
</tbody>
</table>

### Total System Cost Estimate ($M)

![Graph showing the total system cost estimate with confidence levels and corresponding cost values.](#)
RTS Architecture Cost Comparisons

- RTS w/ ATR
- RTS Baseline

Systems Engineering Cost ($M)

- RTS w/ ATR
- RTS Baseline

Full Program Cost ($M)
Agenda

• Introduction
• SysML and COSYSMO Integration
• Case Study: Remote Targeting System UAS
• Conclusions and Future Work
**Requirements**
The number of requirements for the system-of-interest at a specific level of design.

**Interfaces**
The number of shared physical and logical boundaries between system components or functions (internal interfaces) and those external to the system (external interfaces).

**Algorithms**
The number of newly defined or significantly altered functions that require unique mathematical algorithm or component to be derived/designe in order to achieve the system performance requirements.

**Operational Scenarios**
Operational scenarios that a system must satisfy, including nominal and off-nominal threads.

These size drivers are further weighted for complexity levels.
Conclusions and Future Work

- Have demonstrated architectural tradespaces with simpler UAS swarm models for further elaboration on more complex mission scenarios

- We have found a strong correspondence between SysML constructs and system size measures of requirements, interfaces, algorithms, and operational scenarios.
  - Still comparing approaches for complex algorithm representations in SysMLML
  - Require additional attributes for modeling complexity levels of size drivers

- Continue transcribing all UAS architectural variations into SysML for cost tradeoffs to evaluate with other Measures of Effectiveness
  - Expanded mission sets to include heterogeneous UAS teams and more complex scenarios

- Apply method and case study with other MBSE tools, evaluate and compare
  - More detailed modeling to support thread, requirements, functions, algorithms and interface definitions

- Develop guidelines with examples for practitioners on modeling decomposition levels of detail
Completed Student Theses


