Modeling & Simulation of CubeSat Mission

Model-Based Systems Engineering (MBSE) Behavioral Modeling and Execution
Integration of MagicDraw, Cameo Simulation Toolkit, STK, and Matlab using ModelCenter

Sara Spangelo¹
Jet Propulsion Laboratory (JPL), California Institute of Technology

Hongman Kim²
Grant Soremekun³
Phoenix Integration, Inc.

May 29/30, 2013

¹spangelo.sara@gmail.com, ²hkim@phoenix-int.com, ³grant@phoenix-int.com
System Engineering Challenges

Conventional approaches:

- Focus on subset of subsystems
  - Over-simplified, low fidelity
  - Neglect subsystem interactions
- Requirements verification using average/best/worst-cases
  - Fail to capture realistic “dynamic” nature of missions
- Models and simulations are not integrated!
  - “Hacked” together for one-off cases
  - Not modular, extensible, reusable

Why? Lack of integrated modeling/simulation tools to enable system-level engineering design/analysis.
System Engineering Challenges

Particularly an issue for CubeSats\(^1\) because:

- Physical components physically integrated
- Extremely constrained:
  - Limited ability to collect and store energy (e.g. batteries)
- Operational constraints/ decisions coupled
  - When to collect data versus download data?
- Orbits are unknown/ dynamic
  - Little/ no control over launch orbit
  - Experience variation in eclipse duration, may de-orbit
- Operate in inefficient/ stochastic environments

*Integrated models and tools are critical to design and plan for these missions!* 

\(^1\)Type of miniature spacecraft (1U = 10cm\(^3\), <1 kg)
Model-Based Systems Engineering (MBSE\(^1\))

Why MBSE?

1) Enables system-level model capture
   - Formal, accurate, authoritative single source
   - Contains elements, relationships, interactions
   - Multiple compatible views, e.g. physical/functional
   - Requirements verification and traceability

\(^1\) "Formal" model to support requirements, design, analysis, verification
Model-Based Systems Engineering (MBSE)

Why MBSE?

2) Enables integration of models and simulations
   • Connect system-level model to analytical tools (STK, Matlab)
   • Execute dynamic simulation of end-to-end mission
   • Identify failure to satisfy requirements, sub-optimal designs
   • Accommodates re-evaluation when design changes occur
   • Enables co-simulation: simultaneous vehicle/mission design
Motivating Mission Example

- Radio Aurora Explorer (RAX) CubeSat mission
- Science target: plasma irregularities in ionosphere
- Experimental zone in Poker Flat, Alaska
- Global ground station network
- Vehicle constraints: solar panels, battery, data buffer
Motivating Mission Example

**Systems engineering questions:**
- How do satellite states evolve throughout mission?
- Does the vehicle design/operations meet all mission requirements?
- How do changes in spacecraft mission parameters impact performance and requirements satisfaction?
Project: “Model” Operational CubeSat

**Mission goals....**

**Goal #1: Develop fundamental systems model of CubeSat mission**
Capture structure, function, relationships, requirements, traceability.

*Pretty clear-cut if you know what you’re modeling. Accomplished by SSWG*\(^1,2\).*

**Goal #2: Execute realistic behavioral CubeSat scenarios**
Capture operational opportunities, state evolution, mission performance.

*No clear way to do this in March 2013.*


---


Project: “Model” Operational CubeSat

*Mission accomplished*...

Project Deliverables:

- Systems-level SysML model (in MagicDraw)
  - Structure of mission architecture and vehicle
  - Requirements definition and traceability
  - Parametric diagrams to capture analytical relationships
    - Evaluated using MBSE Analyzer
  - Behavioral diagrams to capture dynamic operations
    - Executed using Cameo Simulation Toolkit and MBSE Analyzer

- Analytical models for describing behavior
  - STK, Matlab, Java
  - ModelCenter enabled integration with SysML and automated execution of dynamic scenarios

---

1 A prototype capability was developed for this work that allows CST to execute parametric diagrams via MBSE Analyzer
Modeling Philosophies

For usability/ extensibility:

- **Modularity**: re-usable libraries of parts
  
  \[ e.g. \text{constraint block modules are re-used in many parametric diagrams} \]

- **Patterning**: re-use of modeling patterns
  
  \[ e.g. \text{common pattern in Power and Data Management subsystems} \]

- **Nomenclature**: simple and sufficiently descriptive
  
  \[ e.g. \text{subsystem naming codes used for data rate and power values} \]
CubeSat System Model Architecture

The system model captures requirements, structure, behavior, and parametrics.
Structural Diagrams

Mission Level

Vehicle Level
Mission Requirements

*Drive systems design*

Defines constraint on lowest battery level throughout mission

Defines constraint on minimum download

Defines constraint on lowest data storage level throughout mission
Parametric Diagram

.Constraint blocks defines opportunities

Pointing to a ModelCenter model with STK and Matlab
ModelCenter Model
STK and Matlab Plug-Ins

- Analysis models (STK, Matlab) wrapped and integrated with ModelCenter
- ModelCenter models imported into SysML model constraint blocks with MBSE Analyzer
Systems Tool Kit (STK)

Analytic simulation tool used to propagate orbit & compute:

- Solar state: sun/eclipse, solar panel angles
- Access to experimental zone
- Access to ground stations
Parametric Diagrams

Constraint blocks computes total power

Pointing to a ModelCenter model with Matlab plug-in
Parametric Diagrams

Constraint blocks update satellite states

- Compute energy level at the next time step
- Similar parametric diagrams for experiment data and data download
MBSE Analyzer: Parametric Diagram Solver

- Solves linked parametric diagrams (all 3) simultaneously
- Automated requirements verification (green: pass, red: fail)
Bringing the Model to Life

Main State Machine Diagram

- Entry point of Cameo Simulation Toolkit (CST) behavioral simulation
- Starts “RunOperation” activity diagram that steps through mission simulation
- Updates solar, experiment, and download states according to signals
Main Simulation Loop

Motivation
Overview
Modeling
Simulating
Design Trades
Reflections
Future Work
A prototype capability was developed for this work that allows CST to execute parametric diagrams via MBSE Analyzer.
How are Mission Simulations Performed?

- MagicDraw CST (Behavioral diagrams)
- MBSE Analyzer/ModelCenter (Parametric diagrams)
- STK, Matlab, etc. (Analytical models)
Each column contains updated state at time step

- During CST simulation, MBSE Analyzer is called at each time step
- Data Explorer automatically stores time history of the simulation data
Mission Simulation Results

- Combined simulation SysML behavioral diagrams to STK, Matlab using MBSE Analyzer
- MBSE Analyzer is called at each time step during CST simulation
- Time history of energy level, experiments, and data download is stored

Energy drop in eclipse

Energy drop during download

Motivation
Overview
Modeling
Simulating
Design Trades
Reflections
Future Work
Final Step: Requirements Verification

*Full end-to-end (dynamic) scenario*

- Post-CST simulation: final state stored in an instance specification
- Use MBSE Analyzer to verify requirements with visual tool!
Mission and Design Trade-Offs

Battery Capacity

Nominal Battery Capacity

1/8 Battery Capacity

Requirements defined in SysML model

Infeasibility: Failure to satisfy requirements
Mission and Design Trade-Offs

*Orbit Altitudes*

<table>
<thead>
<tr>
<th>Orbit Type</th>
<th>Semi-Major Axis</th>
<th>Apogee Altitude</th>
<th>Perigee Altitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nominal</td>
<td>7012 km</td>
<td>811.69 km</td>
<td>457.57 km</td>
</tr>
<tr>
<td>High</td>
<td>7500 km</td>
<td>1311.22 km</td>
<td>932.50 km</td>
</tr>
<tr>
<td>Low</td>
<td>6800 km</td>
<td>593.55 km</td>
<td>250.18 km</td>
</tr>
</tbody>
</table>

**Motivation**

**Overview**

**Modeling**

**Simulating**

**Design Trades**

**Reflections**

**Future Work**
Mission and Design Trade-Offs

Ground Station Locations

Downloaded Data, MBytes vs. Time, hours

- Ann Arbor and Menlo Park (Nominal)
- Ann Arbor and Fairbanks
- Fairbanks and Menlo Park
- Minimum Requirement

Location and Description Of Ground Stations In Network

<table>
<thead>
<tr>
<th>Name</th>
<th>State</th>
<th>Latitude (degrees)</th>
<th>Longitude (degrees)</th>
<th>Altitude (km)</th>
<th>Minimum Elevation (degrees)</th>
<th>Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>AnnArbor</td>
<td>MI</td>
<td>42.271</td>
<td>-83.73</td>
<td>0.256</td>
<td>5</td>
<td>0.8</td>
</tr>
<tr>
<td>Fairbanks</td>
<td>AK</td>
<td>64.88</td>
<td>-147.5</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>MenloPark</td>
<td>CA</td>
<td>37.457</td>
<td>-122.2</td>
<td>0.022</td>
<td>0</td>
<td>0.95</td>
</tr>
</tbody>
</table>
Reflecting on Project Experience

How did MBSE enable us to overcome challenges?
• Coupled analytic models with simulation capabilities
• Demonstrated dynamic behavioral modeling
• Achieved requirements verification for full end-to-end missions
• Extensible by use of standards, libraries, patterns, etc.

Lessons Learned
• Working with many tools is challenging (license, versions, etc.)
• STK has a lot of flexibility: exploit use vectors/angles
• Best to automate repeated tasks
• Working with vendors is necessary/advantageous
• Always ask: “Am I using the right modeling/simulation tool?”
Future Work

- Extend the system-level model
  - Higher fidelity models of the spacecraft subsystems
  - Include communication and experimental link budgets
- Extend and refine the behavioral and analysis models
  - Add spacecraft scheduling for optimal use of resources
  - Improve approach for data extraction at specific time (e.g. from STK)
- Automate system and mission parameters trade-offs
  - Extend MBSE Analyzer to drive simulations by CST
  - Enable sensitivity analysis and design optimization
- Generalize the model for applicability to a variety of mission concepts
Acknowledgements

• Mike Bruchanski, Greg Haun, Dave Kaslow from Analytical Graphics, Inc. (AGI)

• Chris Delp, Louise Anderson, Bjorn Cole, James Smith from JPL

• Radio Aurora eXplorer (RAX) Team, Prof. James Cutler

• CubeSat and Amateur Radio Communities

• Dr. Derek Dalle (graphics)