SVMTool

User's Guide

Wesley Voshell & David Burns

Northrop Grumman Corporation

January 2009

Table of Contents

Introduction 3
Installation 4
SVMTool Capabilities 6
SVMTool Inputs 7
Analysis Example: LEO Monolith (Test Case #1) 9
- Step 1: Architecture Building Sheet 9
- Step 2: Replacements Sheet 10
- Step 3: Design Variables Sheet 10
- Step 4: Launch Vehicles Sheet 12
- Step 5: Environment Variables Sheet 13
- Step 6: Attribute Selection Sheet 14
- Final Step: MATLAB Dashboard 15

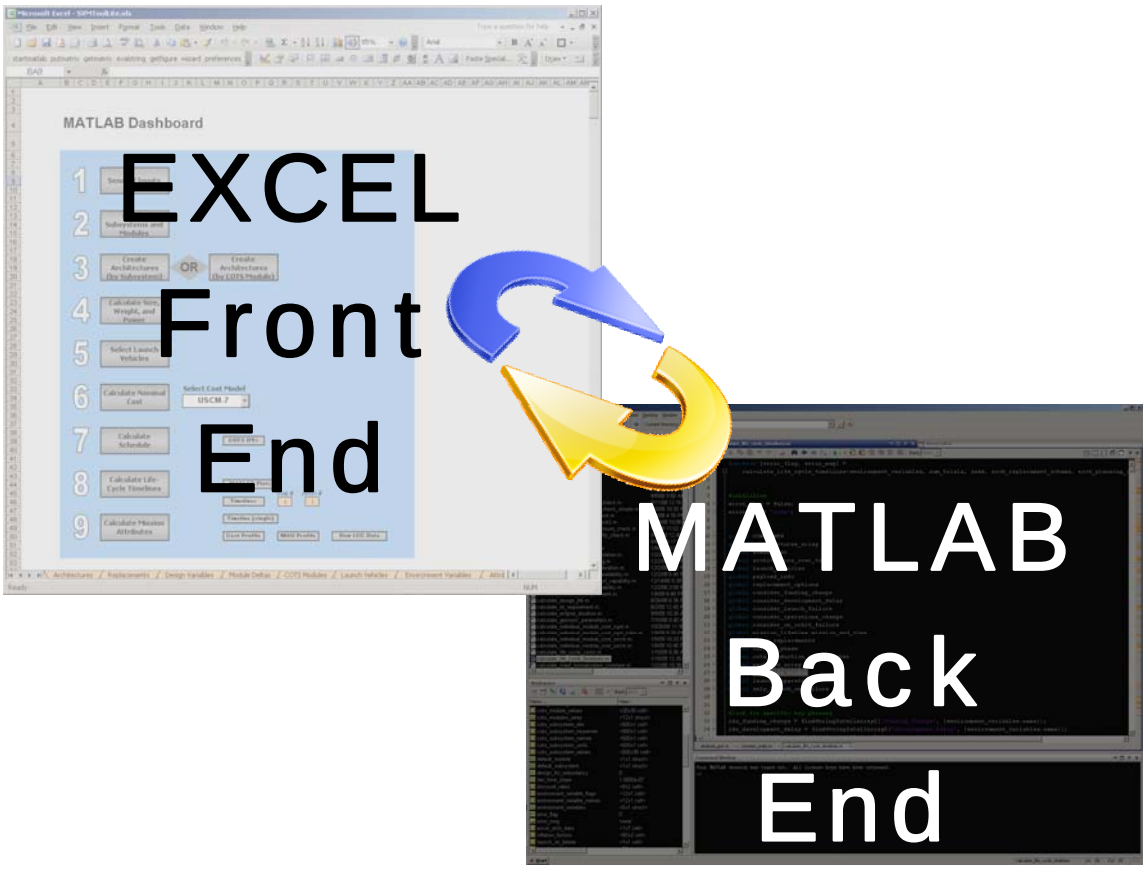
Understanding the Outputs 16
- SWAP Estimate Sheet 16
- LV Selection Sheet 17
- Cost Estimate Sheet 17
- Life-Cycle Analysis Sheet 18
- Life-Cycle Timelines Sheet 19
- Life-Cycle Cost Profile Sheet 20
- Life-Cycle Utility Profile Sheet 20

Analysis Example: Fractionated Architecture (Test Case #2) 22
- Step 1: Architecture Building Sheet 22
- Step 2: Replacements Sheet 22
- Step 3: Design Variables Sheet 23
- Step 4: COTS Modules Sheet 23
- Step 5: Environment Variables Sheet 23
- Final Step: MATLAB Dashboard 24
- Examining the Output Sheets: 24

Design Variables Sheet 26
COTS Modules Sheet 27
Launch Vehicles Sheet 28
Environment Variables Sheet 28
Cost Model Sheet 29
Schedule Data Sheet 29
MATLAB Dashboard 30
Known Issues 32
SVMTool-Recognized Units, with Abbreviations 33

Introduction

The SVMTool is an Excel- and MATLAB-based trade tool that lets users construct and analyze space system concepts on their desktop computers. SVMTool users primarily interact with the Excel-based front end, which includes input sheets, an analysis "dashboard," and various output data sheets and charts.



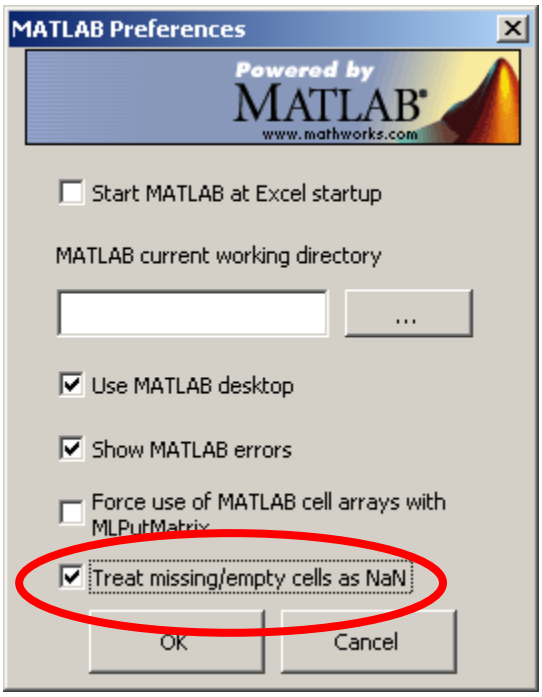
Users that are experienced with MATLAB programming have the option to modify the back end code that does the bulk of calculations. If you fall into this category, please feel free to examine the various Excel macros and MATLAB m-files that you will be installing in the next section. You should also refer to the companion document to this guide, the Algorithm Description Document.

Installation

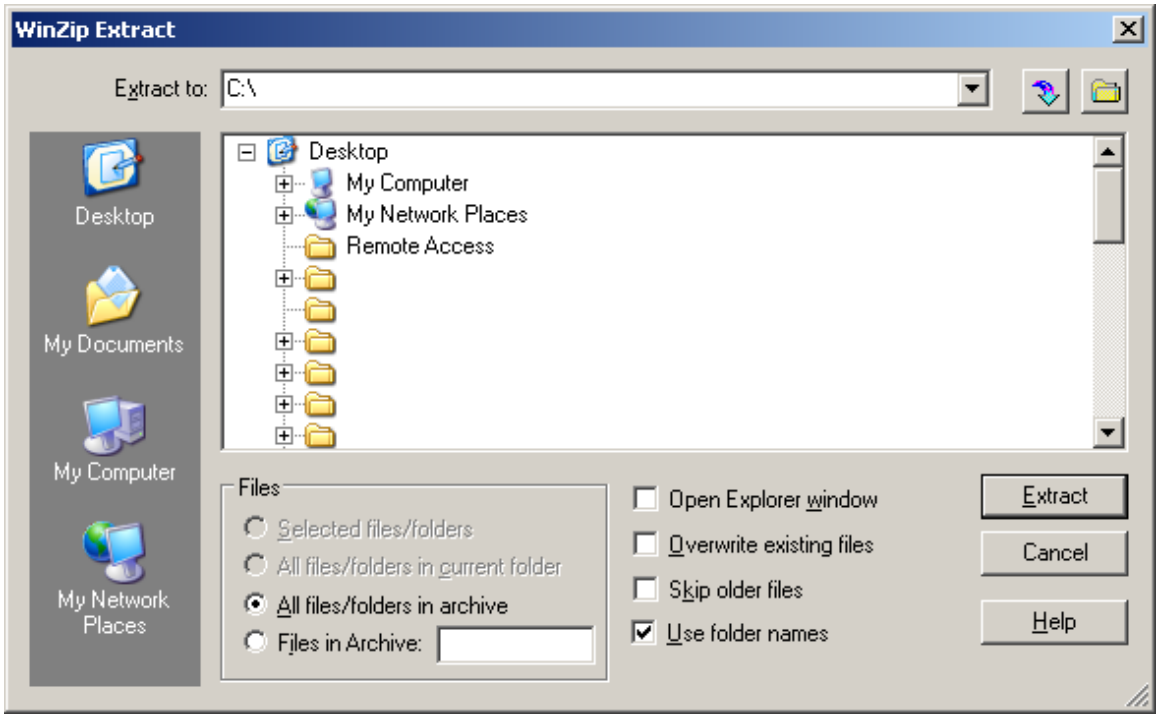
1) To use SVMTool, you will need Excel (2003 or higher) and MATLAB (2007b or higher) installed on your PC. You will also need to install the MATLAB Excel (or Spreadsheet) Link and Statistics toolboxes.

2) Once you've installed MATLAB, follow the instructions in the MATLAB Help on configuring Excel Link (aka Spreadsheet Link EX).

Important: you will need to check the "Treat missing/empty cells as NaN" box in the MATLAB preferences window inside **Excel** (see figure below).

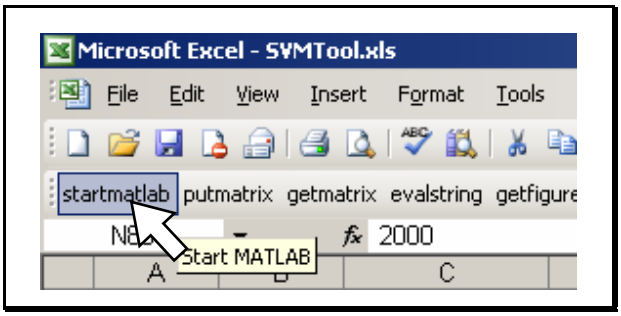


3) Extract the "SVMTool.zip" file to a folder on your computer:



The location is not important.

4) Open the SVMTool.xls file in Excel, and then click the “startmatlab” button. MATLAB may take a while to open – be sure you don’t click on “startmatlab” more than once:



If everything has worked to this point, then you are ready to begin using the SVMTool.

SVMTool Capabilities

The SVMTool was designed for high-level space mission engineering, with a focus on fractionated spacecraft concepts. These are some of the key features:

S/C Cost Modeling
Multiple Cost Models
Learning Curve
Discounted Costs
Inflation

S/C SWaP Modeling
Automatic Sizing of Subsystems

Performance Models
System Availability
Revisit Rate

Launch Vehicle Selection

S/C Development Schedule

Life-Cycle Dynamics
Launch Failures
On-Orbit Failures (Infancy and End-of-Life)
Development Delays

How those features are implemented is the focus of the Algorithm Description Document. Figure 1, which you'll also see in the ADD, shows just what's going on under the SVMTool hood when it runs:

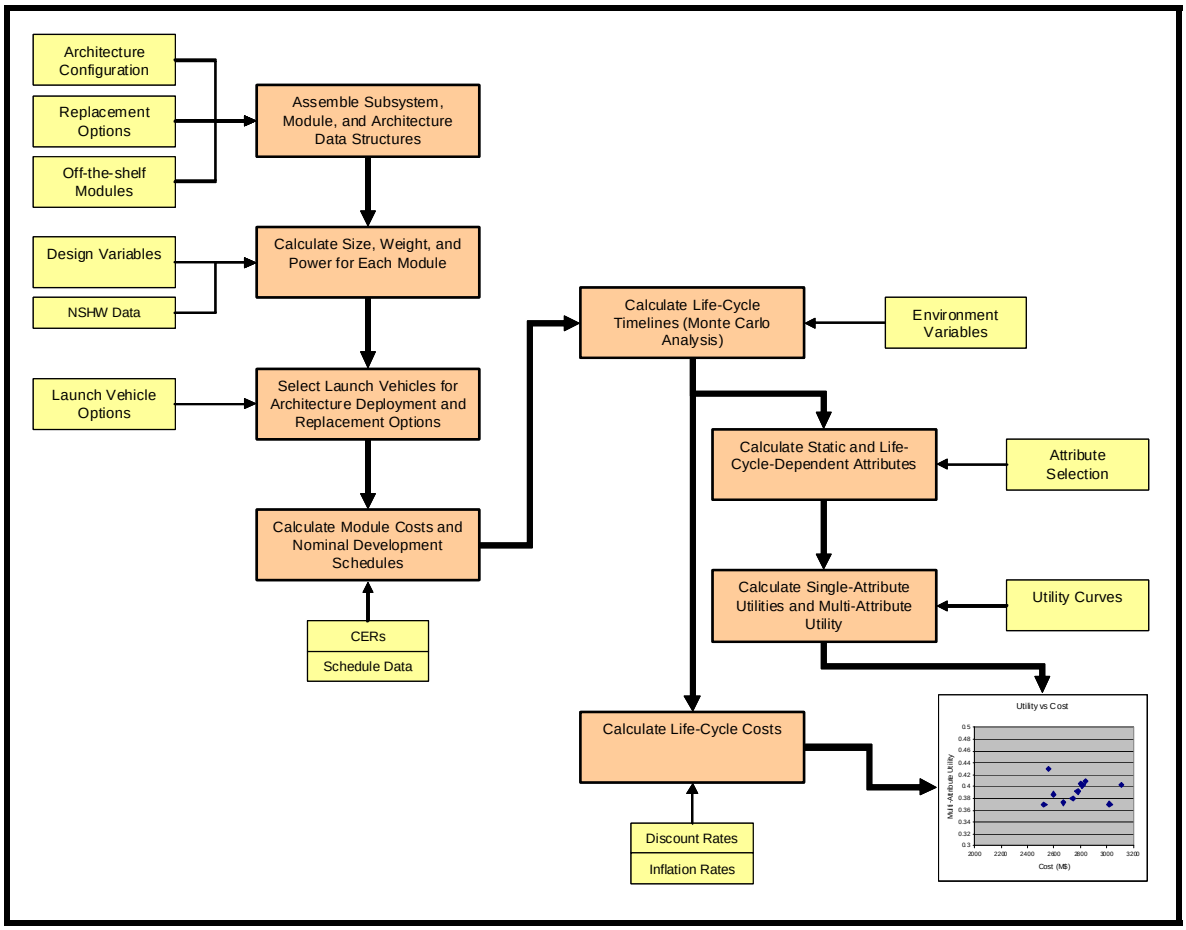
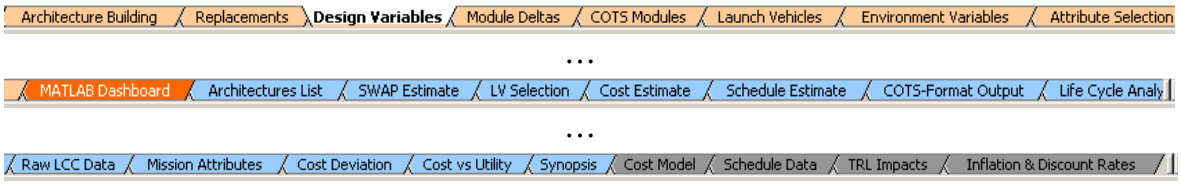


Figure 1. SVMTool data processing overview

As you're going through the examples that follow, it might help to refer back to this diagram.

SVMTool Inputs

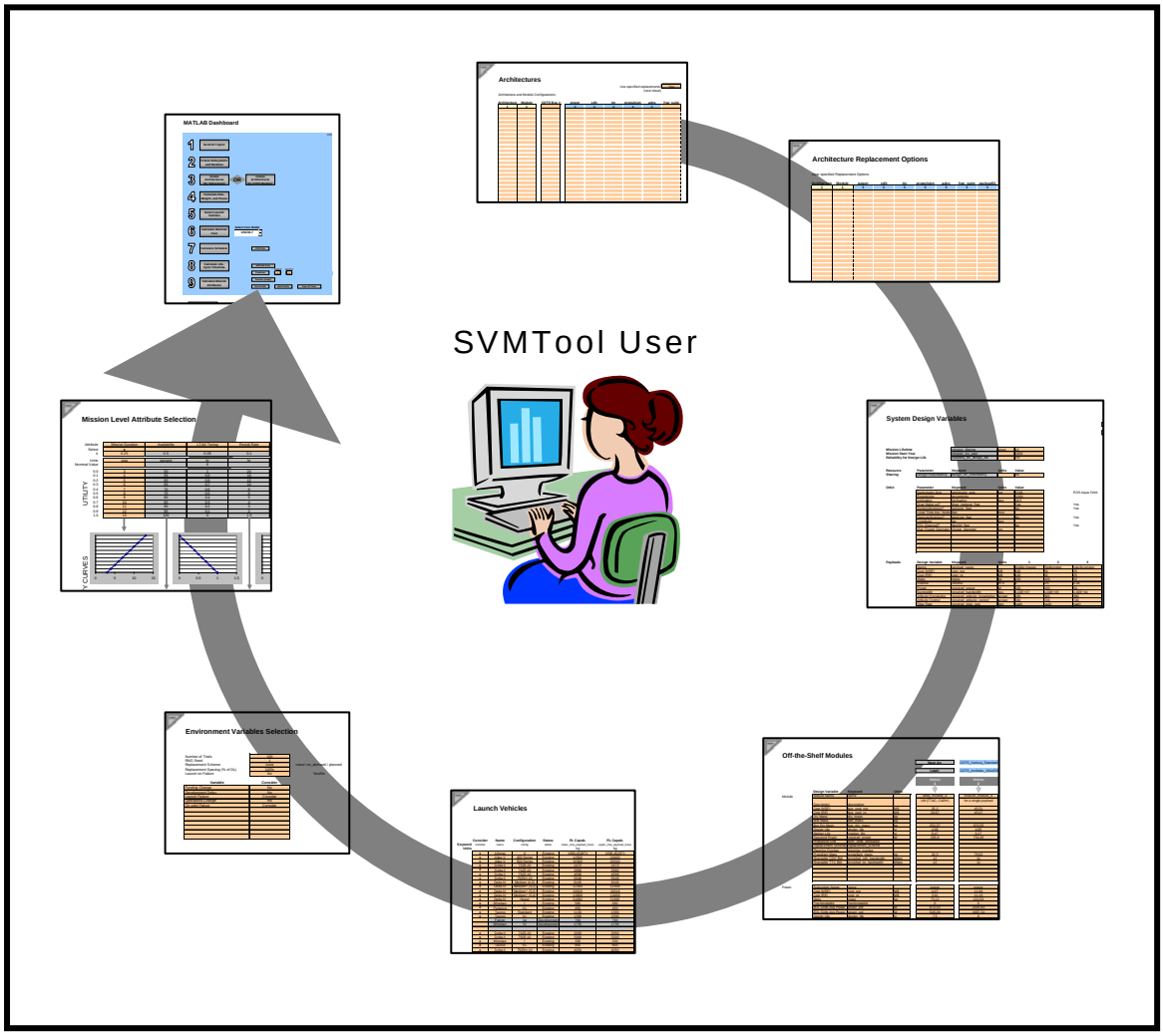
You will mostly interact with SVMTool through the orange-tabbed input sheets, the "MATLAB Dashboard" sheet, blue-tabbed output sheets, and gray-tabbed ancillary data sheets:



The various input sheets allow you, for example, to:

- Select mission-level attributes of interest

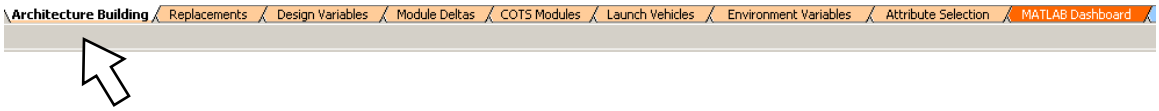
- Configure spacecraft at the subsystem level
- Change design parameter values
- Configure launch vehicles
- Modify Cost Estimating Relationship (CER) coefficients
- Select life-cycle perturbations of interest



Analysis Example: LEO Monolith (Test Case #1)

The best way to learn how to use the tool is to just start using it. Let's walk through the tool by setting up and running an example case of a hypothetical low-earth orbit spacecraft. This hypothetical spacecraft will perform an Earth-observing mission with three scientific payloads.

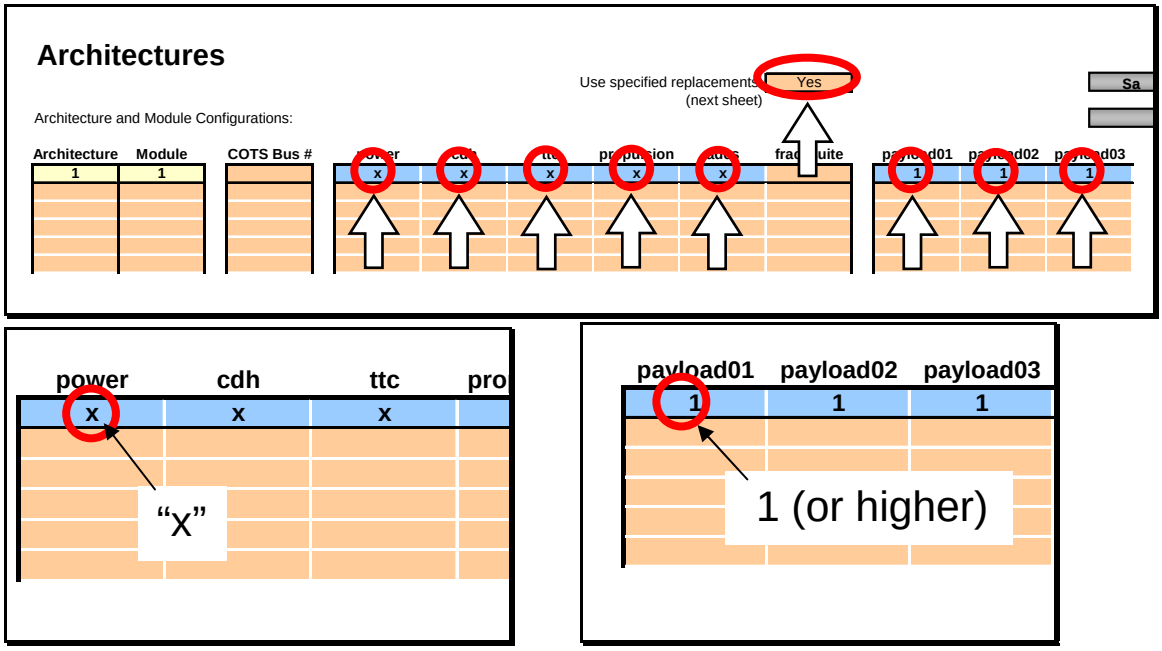
Step 1: Architecture Building Sheet



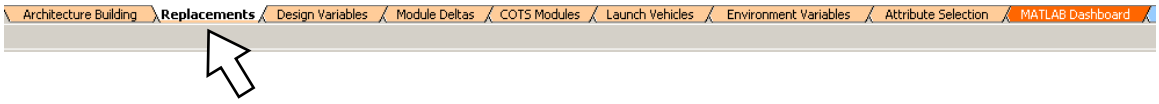
This sheet allows you to perform high-level design of the system, by inputting the number of free-flying modules that comprise the architecture, and the subsystems that are present on each module. Those subsystems and payloads are in turn defined at a lower level on the "Design Variables" sheet, which will be described below.

It is possible to set up more than one architectures on this sheet (see figure below). Each architecture will be evaluated independently when you actually perform the analysis.

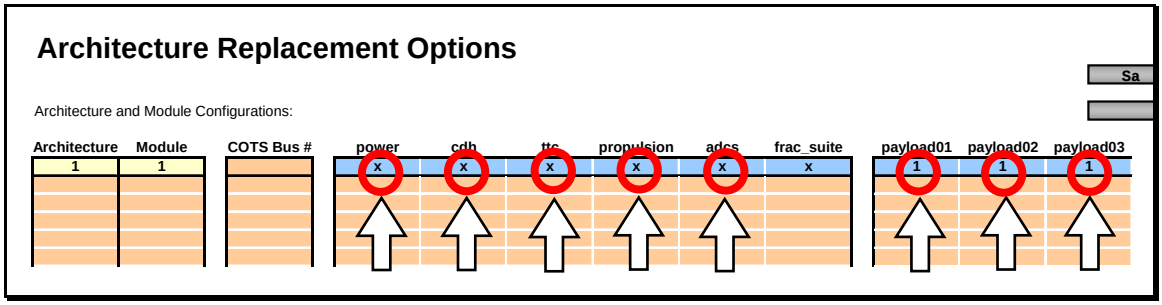
You will probably first notice several color-coded areas on this sheet. In general, throughout the SVMTool spreadsheet, orange-shaded cells represent areas for user input and gray-shaded cells represent areas that are either currently not selected for user input, or cannot be modified by the user.



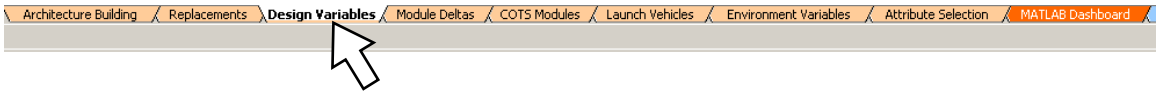
Step 2: Replacements Sheet



This sheet allows you to specify on-orbit replacement options for each module in each architecture. Specification is done as in the previous step, with specific architectural replacements being listed under their architecture.



Step 3: Design Variables Sheet



This sheet allows you to describe the various payloads and subsystems via design constants and design variables. The reference orbit for the system is also defined on this sheet.

System Design Variables

Mission Lifetime	mission_lifetime	years	10
Mission Start Year	mission_ioc_year		2009
Reliability for Design Life	reliability_for_design_life		0.9

Resource Sharing	**Parameter**	**Keyword**	**Units**	**Value**
	Design Redundancy	design_for_redundancy		No

Orbit	**Parameter**	**Keyword**	**Units**	**Value**
	Semi-major Axis	semimajor_axis	km	6748
	Eccentricity	eccentricity		0.001
	Inclination	inclination	deg	28.5
	Drag Make-up?	drag_makeup_flag		Yes
	Sunsynchronous?	sunsync_flag		No
	Local Time Asc. Node	ltan	hour	
	Geosynchronous?	geosync_flag		No
	Longitude	lan	deg	
	EOL Disposal?	deorbit_flag		No
	Max Cluster Diameter	cluster_diameter	km	

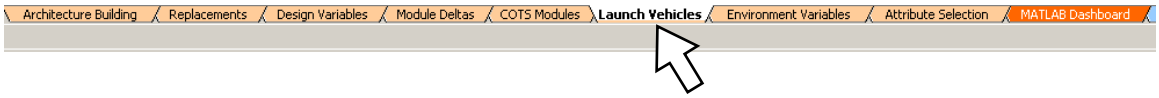
Payloads

Design Variable	**Keyword**	**Units**	**1**	**2**	**3**
Name	payload_name		Visible Imager	Radiometer	Interferometer
Cost (NRE)	cost_nre	M$	100	50	25
Cost (RE)	cost_re	M$	100	50	25
Mass	mass	kg	200	100	50
Volume	volume	m^3	1	0.5	0.25
Power	required_power				
Bandwidth	required_bandwidth				
Attitude Knowledge	required_attitude_knowledge				
Attitude Control	required_attitude_control	arcsec	360	720	720
Slew Rate	required_slew_rate	deg	nadir	nadir	nadir
FOR Half-angle	for_half_angle	deg	60	45	45
Design Life	design_life	yr	6	6	6
TRL	trl		9	8	7
Fractionatable	fractionatable		Yes	Yes	Yes
Prob. Infant Mortality	prob_infant_mortality		0.03	0.04	0.05

See Inputs In Next Figure

Units	1	2	3
	Visible Imager	Radiometer	Interferometer
M$	100	50	25
M$	100	50	25
kg	200	100	50
m^3	1	0.5	0.25
W	200	100	50
bps	1.00E+07	1.00E+05	1.00E+04
arcsec	180	360	360
arcsec	360	720	720
deg	nadir	nadir	nadir
deg	60	45	45
yr	6	6	6
	9	8	7
	Yes	Yes	Yes
	0.03	0.04	0.05

Step 4: Launch Vehicles Sheet



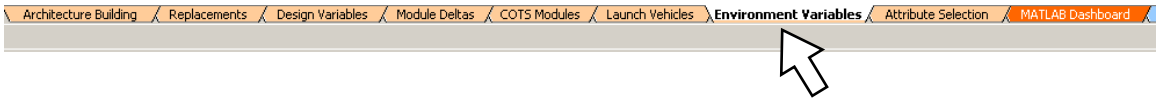
This sheet contains the launch vehicle cost, performance, and reliability characteristics. Note that at this time payload capacity is relative to a single reference orbit – if you want to consider different orbit regimes, other than the ones listed, you will need to insert new launch vehicle data for each orbit.

You can choose to consider any launch vehicle by placing an “x” in its consider column.

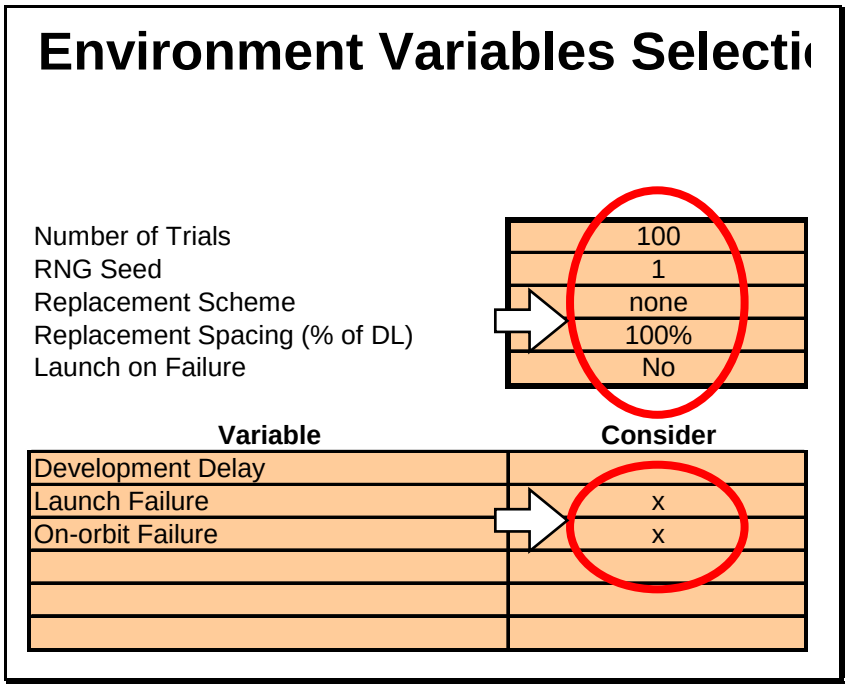
Launch Vehicles

Consider consider	Name name	Configuration config	Status status	PL Capab. lower_max_payload_mass kg	PL Capab. upper_max_payload_mass kg
x	Athena	II	Existing	1696.452871	1696.452871
x	Atlas V	400 Series	Existing	12500	18400
x	Atlas V	500 Series	Existing	10300	20520
x	Delta II	7326-10	Existing	2472	2472
x	Delta II	7425-10	Existing	2880	2880
x	Delta II	7925-10	Existing	4598	4598
x	Delta II	7925H-10	Existing	5648	5648
x	Delta IV	Medium (4,0)	Existing	9106	9106
x	Delta IV	Medium+ (4,2)	Existing	12300	12300
x	Delta IV	Medium+ (5,2)	Existing	10616	10616
x	Delta IV	Medium+ (5,4)	Existing	13869	13869
x	Delta IV	Heavy	Existing	21892	21892
x	Minotaur	I	Existing	580	580
x	Pegasus	XL	Existing	450	450
x	Taurus	Standard	Existing	1130	1380
x	Taurus	XL	Existing	1390	1590

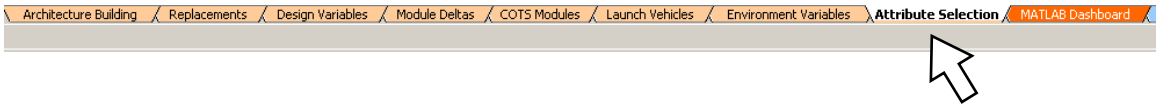
Step 5: Environment Variables Sheet



This sheet sets up the life cycle analysis tool. The Number of Trials and RNG Seed set up the Monte Carlo analysis. The Replacement Scheme defines how replacement spacecraft are developed and launched. On Demand starts development when the previous one fails. Planned starts development such that the replacement spacecraft is ideally launched as the previous spacecraft reaches its design life.

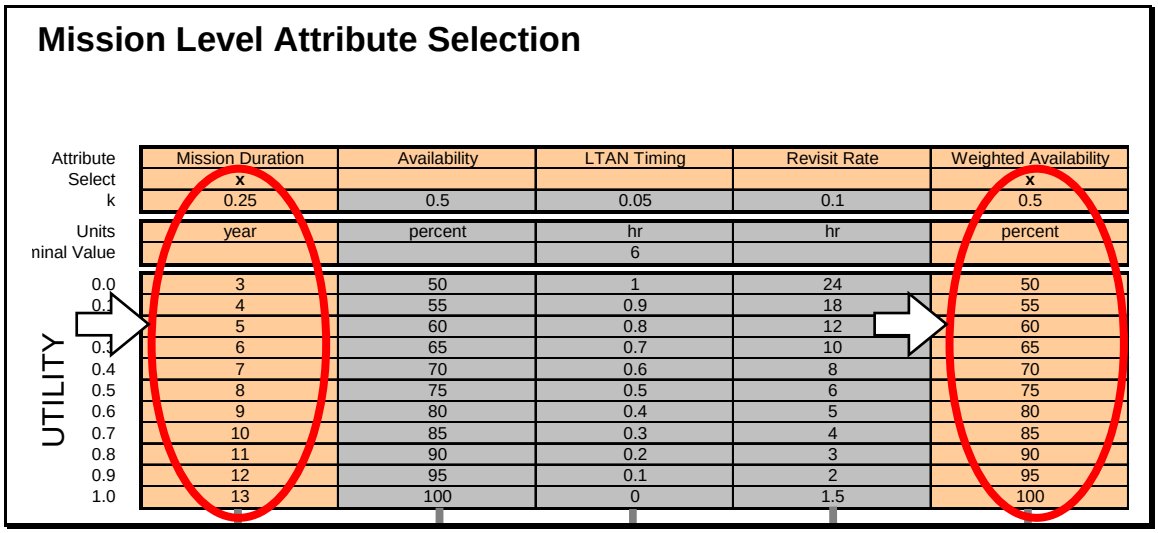


Step 6: Attribute Selection Sheet



On this sheet, you can choose which attributes (from a given set) are important for the evaluation of your system. In addition, you can assign relative weights (k values) to the attributes, and input utility curves that map attribute values to normalized utility values. These utility values are used later on in Multi-Attribute Utility (MAU) calculations.

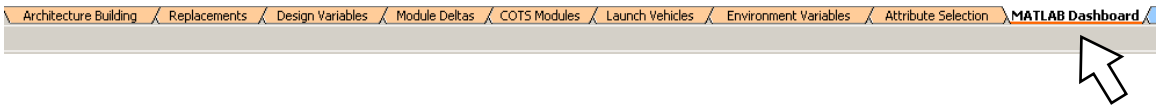
For this example, we will select “Mission Duration” and “Weighted Availability” as our two mission-level attributes of interest. Enter an “x” in the “Select” row for those two attributes. You will notice that the column changes from gray to orange, indicating that it is now being considered as a user input.



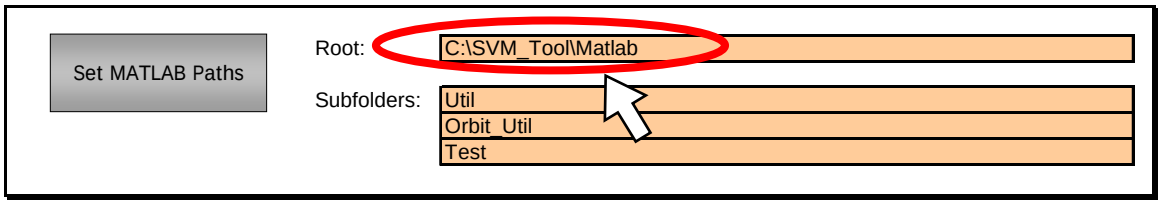
Mission Level Attribute Selection

Attribute	Mission Duration	Availability	LTAN Timing	Revisit Rate	Weighted Availability
Select	x				x
k	0.25	0.5	0.05	0.1	0.5
Units	year	percent	hr	hr	percent
ninal Value			6		
UTILITY 0.0	3	50	1	24	50
0.1	4	55	0.9	18	55
	5	60	0.8	12	60
0.3	6	65	0.7	10	65
0.4	7	70	0.6	8	70
0.5	8	75	0.5	6	75
0.6	9	80	0.4	5	80
0.7	10	85	0.3	4	85
0.8	11	90	0.2	3	90
0.9	12	95	0.1	2	95
1.0	13	100	0	1.5	100

Final Step: MATLAB Dashboard



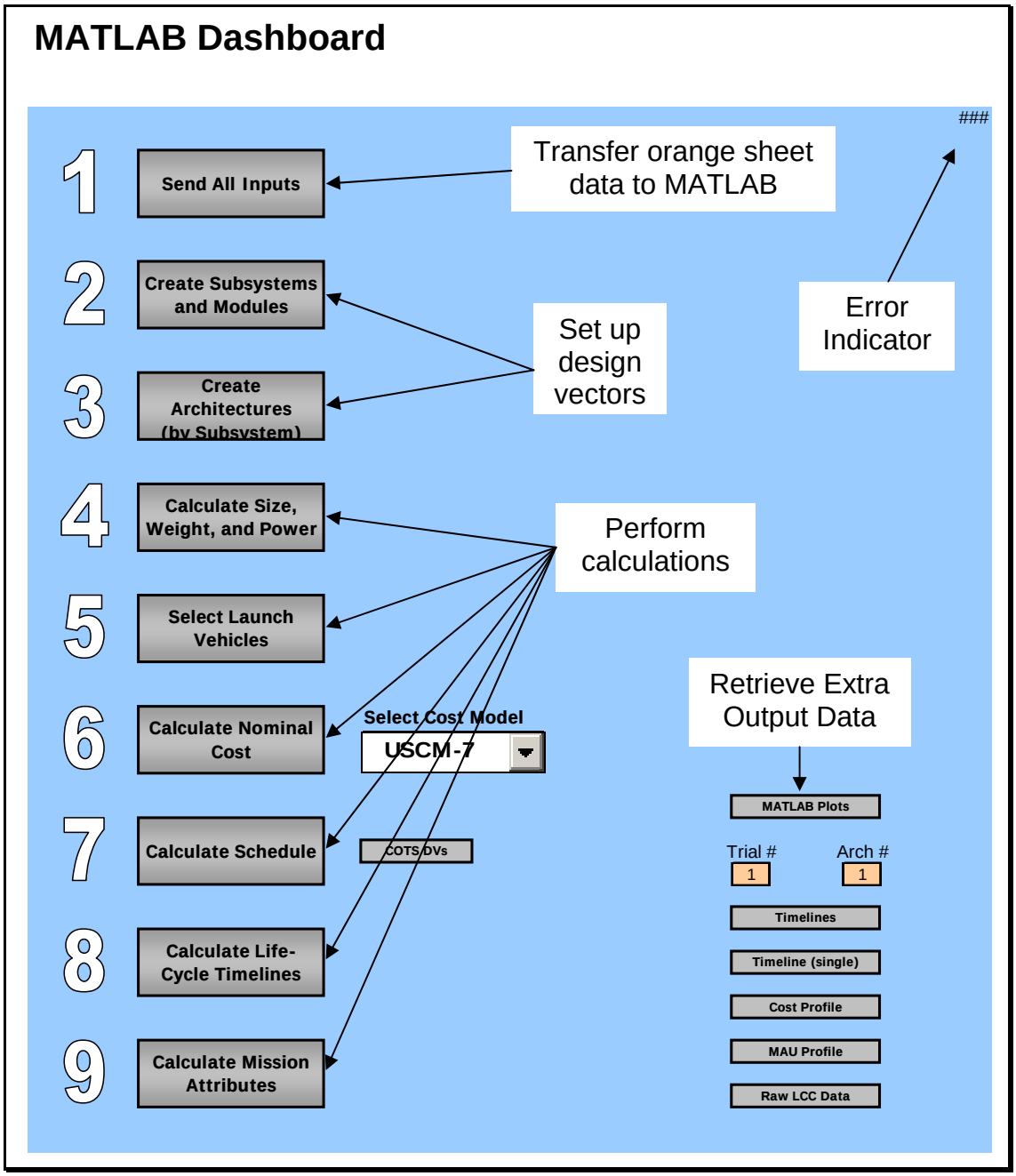
Before using the SVMTool, you must first tell it where your files reside on your computer. The Root folder is the location of the MATLAB folder that contains the main SVMTool m-files, and the Subfolders contain various utility files. Click the "Set MATLAB Paths" button once you've entered the correct paths:



Please note that this will only work if MATLAB has already been opened via the "startmatlab" button in Excel. The MATLAB directory window should display the Root folder.

Once the inputs are set and the MATLAB interface is configured, the tool is ready to be run. **To receive valid output, the buttons must be executed sequentially**.

Click Buttons 1 through 9, in order, to generate results. If the hourglass appears, you will need to wait until MATLAB finishes before clicking the next button.



Understanding the Outputs

Now let's examine some of the blue-tabbed output sheets…

SWAP Estimate Sheet

The SWAP Estimate sheet provides mass breakdowns for each module in each architecture, including propellant mass, and also a couple key design parameters like solar array area and beginning-of-life power:

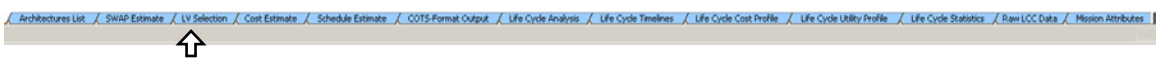
Mass/Size/Power Estimates

				kg			W	W	m^2	A-h
Architecture	**Variation**	**Arch Mass**	**Module**	**Propellant**	**Module Dry**	**Module Wet**	**BOL Power**	**EOL Power**	**Array Area**	**Batt Cap**
1	1	2000	1	576	1423	2000	2301	1867	18	29

For our monolith example, the total wet mass of the spacecraft is estimated to be 2000 kg, 576 kg of which is the propellant required to maintain the relatively low altitude for 10 years.

kg										
payload01	**payload02**	**payload03**	**power**	**adcs**	**ttc**	**cdh**	**thermal**	**propulsion**	**structure**	**frac_suite**
200	100	50	298	99	20	25	49	87	495	

LV Selection Sheet

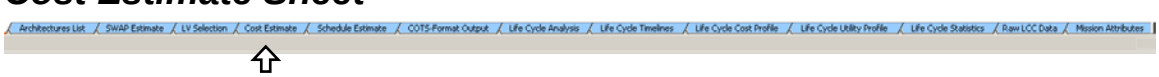


This sheet displays the lowest-cost launch vehicle options for each module, and for the entire architecture, if it were to be launched together

Launch Vehicle Selection

		kg			kg	
Architecture	**Variation**	**Arch Mass**	**Arch LV**	**Module**	**Mod Mass**	**Mod LV**
1	1	2000	Delta II 7326-10	1	2000	Delta II

Cost Estimate Sheet



This sheet shows the breakdown of costs by subsystem for each module in each architecture. These are nominal costs, as predicted by the CERs chosen on the Dashboard. For this monolith example, the total spacecraft cost, in FY2009 terms, is $656M.

Cost Estimates

Production Unit: 1

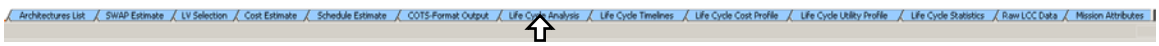
FY2009$M

Architecture	Variation	Arch	Arch + LV	Module	Bus	ATLO	PM/SE	Tot NRE	Tot RE	Total
1	1	655.89	703.89	1	83.99	94.30	127.60	304.71	351.18	655.89

Note that the payload costs are identical to the input values – they will always be treated as pass-through costs.

payload01	payload02	payload03	power	adcs	ttc	cdh	thermal	propulsion	structure	frac_suite
100.00	50.00	25.00	8.04	12.94	1.13	1.34	2.58	0.78	16.78	

Life-Cycle Analysis Sheet



This sheet displays the results of the life-cycle dynamics simulation, both in terms of the environment perturbations and the life-cycle costs. The mean LCC for our monolith, including development delay costs, is $763M. This is the mean of all LCCs across all 100 trials. The mean discounted LCC is lower, at $732M, which is to be expected. For this example, where replacements were not an option, the only variation in LCC comes from development delays. This variation is represented by the LCC Std Dev value.

Life Cycle Statistics

FY2009$M

Architecture	Variation	Num Trials	LCC Mean	LCC Std DEv	LCC Min	LCC Max	LCC 5th	LCC 95th	Disc. LCC	c. LCC Std D
1	1	100	704	0	704	704	704	704	680	0

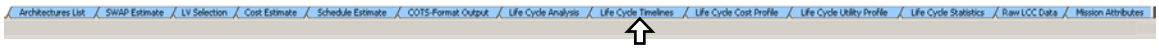
Disc. LCC Mi	Disc. LCC Ma	Disc. LCC 5th	Disc. LCC 95t	Mod Fail	Mod Fuel	Mod Delay	Launch Fail	# Mods	
680	680	680	680	1	0.2	0	0.03		

The failure rates represent the mean number of failures of that particular subsystem, across all modules and all trials. Subsystem failures that occur after the host module has lost functionality are not counted against this statistic

Failure Rates

payload01	payload02	payload03	power	adcs	ttc	cdh	thermal	propulsion	structure	frac_suite	propellant
0.21	0.21	0.16	0.07	0.2	0.18	0.12	0	0.4	0	0	0.2

Life-Cycle Timelines Sheet



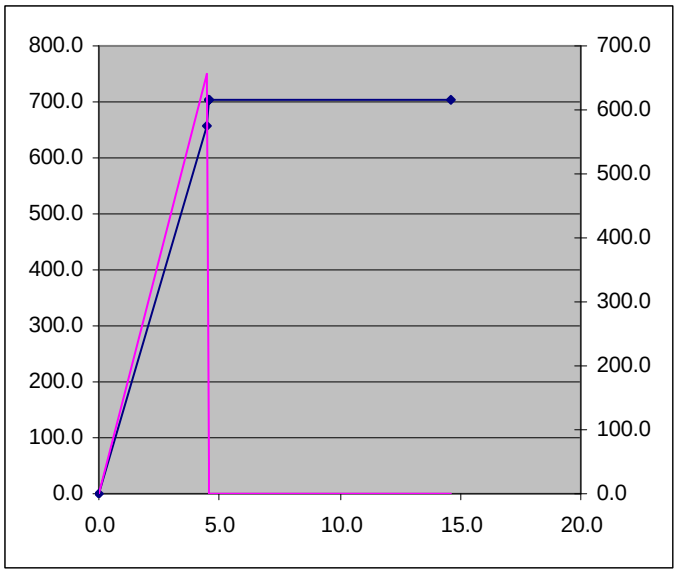
This sheet attempts to tell the story of individual timelines as they were played out in the life-cycle dynamics simulation in SVMTool. Each module's timeline corresponds to the specific trial # selected on the Dashboard, and can be updated by changing the Trial # and then clicking the "Timelines" button.

The monolith's timeline is pretty straightforward – it was built, delivered, launched, inserted into orbit, and declared dead after the Power subsystem failed in infancy. The times (and costs, when applicable) are listed for each event:

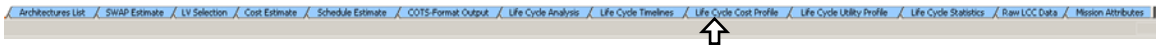
Architecture	Variation		1	2	3
1	1	Event	atp_event	delivery_event	launch_event
		Module	1	1	1
		System	Architecture	Architecture	Delta II 7326-10
		Time	0.0	4.5	4.6
		Cost	0.0	656.0	48.0
		Cum Cost	0.0	656.0	704.0

4	5	6	7
eginning_of_mission_eve	orbit_insertion_event	on_orbit_failure_event	end_of_mission_event
0	1	1	0
Architecture	Architecture	power	Architecture
4.6	4.6	4.6	14.6
0.0	0.0	0.0	0.0
704.0	704.0	704.0	704.0

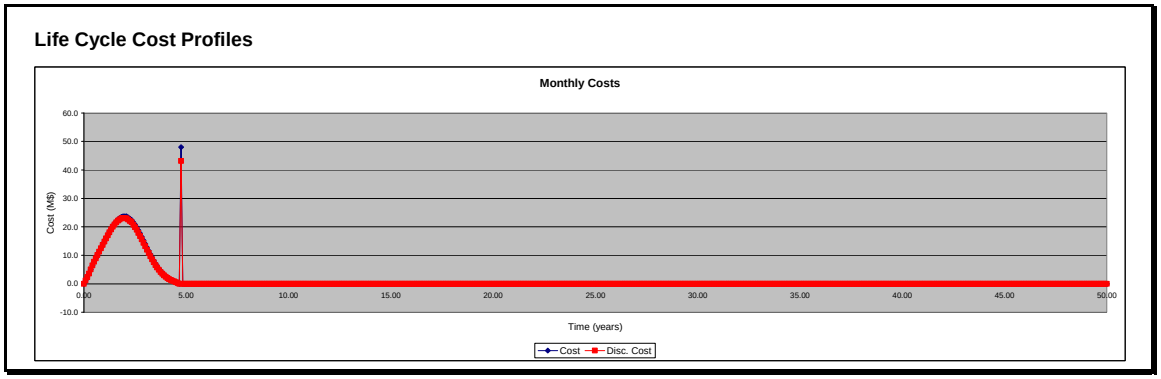
The costs associated with some of the events are displayed in the accompanying graph, where the blue line is the cumulative LCC, and the magenta line is the individual event costs. Keep in mind that YOUR trial #1 may have played out differently during your MATLAB run, owing to the stochastic nature of the life-cycle simulation.



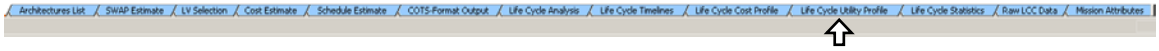
Life-Cycle Cost Profile Sheet



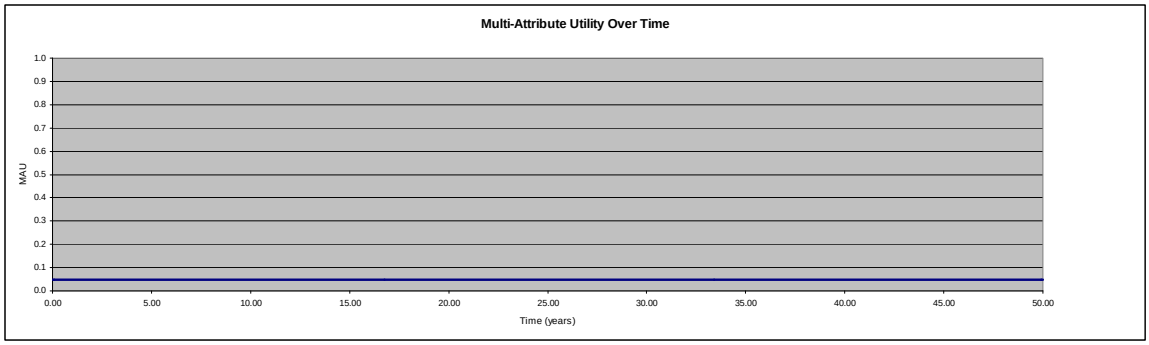
This sheet shows LCC costs as a continuous profile over the entire mission duration, for the specific trial # selected on the Dashboard. For the monolith example, we can see the "pig-in-a-python" cost hump of the development phase, along with a spike for the launch cost.

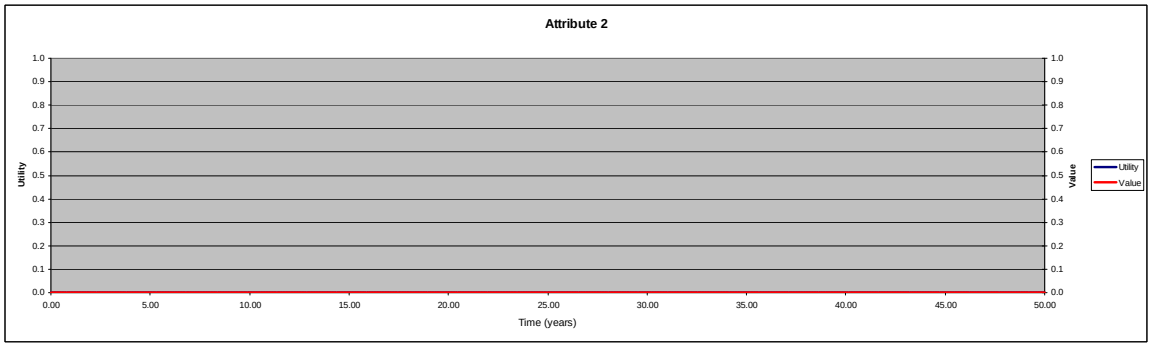


Life-Cycle Utility Profile Sheet



This sheet shows utility and performance over the mission duration, again for the specific trial # selected on the Dashboard. Remember from the Timelines output that the spacecraft's Power subsystem failed in infancy – this means that the mission essentially failed from the start, resulting in zero availability across the board.


Multi-Attribute Utility Over Time
MAU
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0.00
5.00
10.00
15.00
20.00
25.00
30.00
35.00
40.00
45.00
50.00
Time (years)


Attribute 2
Utility
Value
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0.00
5.00
10.00
15.00
20.00
25.00
30.00
35.00
40.00
45.00
50.00
Time (years)
Utility
Value

Analysis Example: Fractionated Architecture (Test Case #2)

For the next example, let's examine a fractionated architecture that consists of three wirelessly connected modules flying in close formation. One of the modules will host the same three payloads from the previous example The other two modules will be identical copies of a COTS infrastructure module, both of which will provide all the data processing and downlinking capabilities for the architecture.

In this example, we will only change a few of the sheets from the previous example (with minimal explanation). Start by going back to the original Architecture Building sheet.

Step 1: Architecture Building Sheet

Architectures

Use specified replacements: Yes
(next sheet)

Sa

Architecture and Module Configurations:

Architecture	Module	COTS Bus #	power	cdh	ttc	propulsion	adcs	frac_suite	payload01	payload02	payload03
1	1		x			x	x	x	1	1	1
1	2	1									
1	3	1									

Step 2: Replacements Sheet

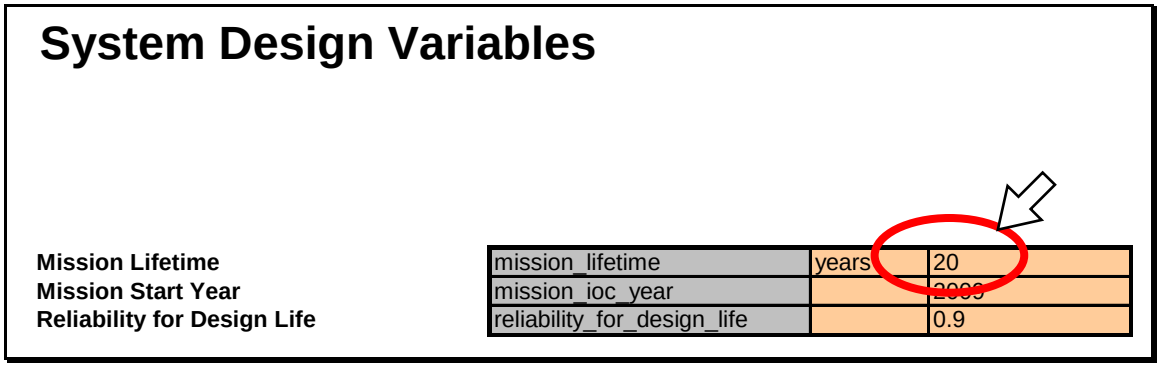
Architecture Replacement Options

Sa

Architecture and Module Configurations:

Architecture	Module	COTS Bus #	power	cdh	ttc	propulsion	adcs	frac_suite	payload01	payload02	payload03
1	1		x			x	x	x	1	1	1
1	2	1									
1	3	1									

Step 3: Design Variables Sheet

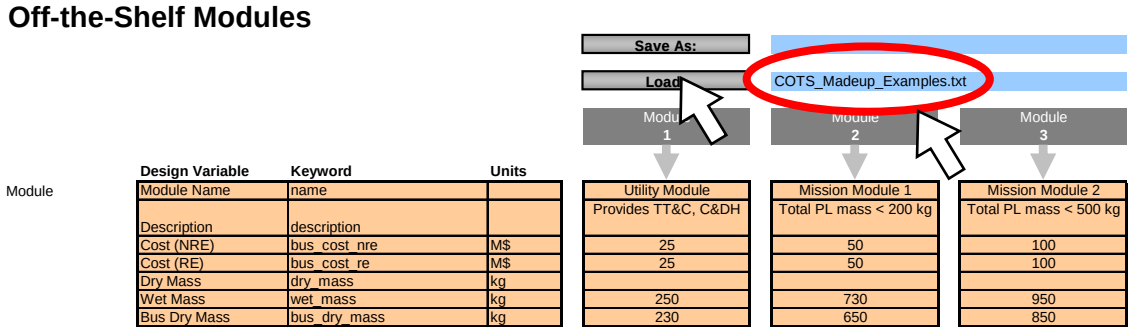


System Design Variables

Mission Lifetime	mission_lifetime	years	20
Mission Start Year	mission_ioc_year		2009
Reliability for Design Life	reliability_for_design_life		0.9

Step 4: COTS Modules Sheet

The next step, on the "COTS Modules" sheet, involves loading in data from a text file (which should already exist in the base SVMTool folder). Type "COTS_Madeup_Examples.txt" into the blue cell circled in the figure below, then click the "Load:" button. You might not see any changes to the data that is already in the sheet, but the data is being automatically entered nonetheless.



Off-the-Shelf Modules

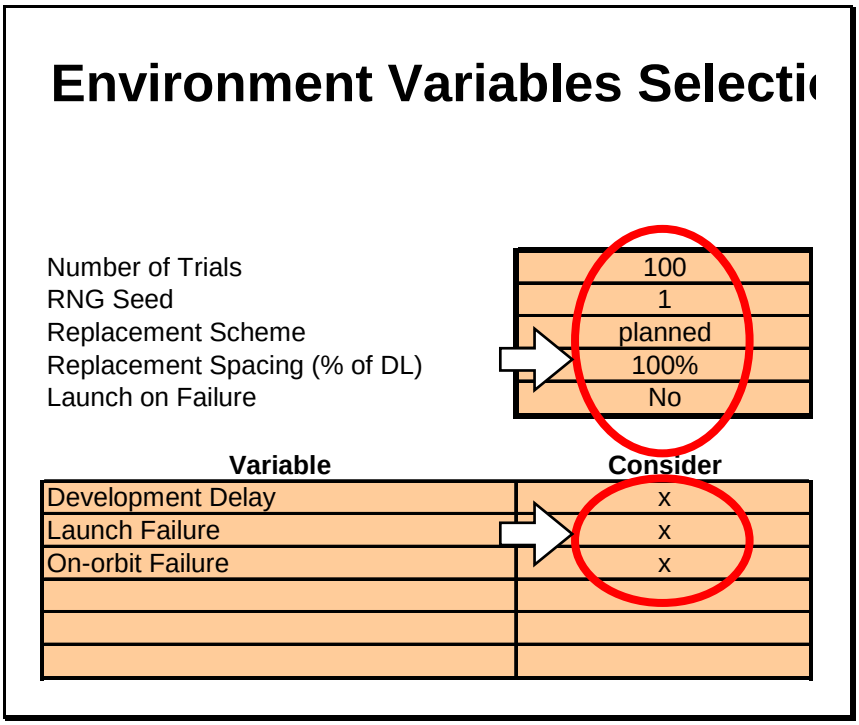
Save As:

Load:

COTS_Madeup_Examples.txt

Module	Design Variable	Keyword	Units	Module 1	Module 2	Module 3
	Module Name	name		Utility Module	Mission Module 1	Mission Module 2
	Description	description		Provides TT&C, C&DH	Total PL mass < 200 kg	Total PL mass < 500 kg
	Cost (NRE)	bus_cost_nre	M$	25	50	100
	Cost (RE)	bus_cost_re	M$	25	50	100
	Dry Mass	dry_mass	kg			
	Wet Mass	wet_mass	kg	250	730	950
	Bus Dry Mass	bus_dry_mass	kg	230	650	850

Step 5: Environment Variables Sheet



Final Step: MATLAB Dashboard

For the final step, click through the MATLAB Dashboard buttons just like in the previous example, then examine the output sheets.

Examining the Output Sheets:

Compare your output with the following expected results. Your values should be within 1%.

SWAP Estimate:

Architecture	Variation	Arch Mass	Module	Propellant	Module Dry	Module Wet	BOL Power	EOL Power	Array Area	Batt Cap
1	1	2287	1	515	1272	1787	1998	1621	16	25
			2	25	225	250	700	500	-99999	-99999
			3	25	225	250	700	500	-99999	-99999

payload01	payload02	payload03	power	adcs	ttc	cdh	thermal	propulsion	structure	frac_suite
200	100	50	261	90			44	79	442	5
			65	25	20	25	5	15	65	5
			65	25	20	25	5	15	65	5

Launch Vehicle Selections:

Architecture	Variation	Arch Mass	Arch LV	Module	Mod Mass	Mod LV
1	1	2287	Delta II 7326-10	1	1787	Delta II
				2	250	Minotaur
				3	250	Minotaur

Cost Estimate:

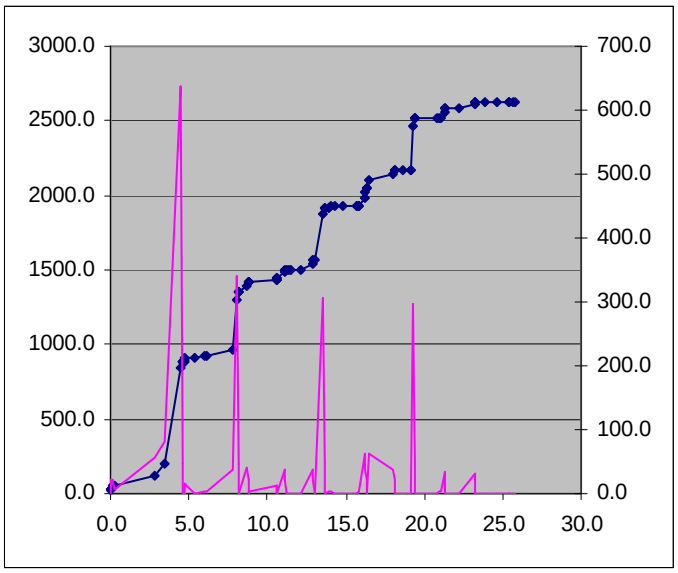
Architecture	Variation	Arch	Arch + LV	Module	Bus	ATLO	PM/SE	Tot NRE	Tot RE	Total
1	1	777.57	825.57	1	71.17	91.54	124.43	295.62	341.51	637.14
				2	50.00	11.74	20.98	37.95	44.77	82.72
				3	50.00	11.74	20.98	37.95	44.77	82.72

payload01	payload02	payload03	power	adcs	ttc	cdh	thermal	propulsion	structure	frac_suite
100.00	50.00	25.00	7.05	11.98			2.40	0.74	15.28	0.00
			-99999.00	-99999.00	-99999.00	-99999.00	-99999.00	-99999.00	-99999.00	-99999.00
			-99999.00	-99999.00	-99999.00	-99999.00	-99999.00	-99999.00	-99999.00	-99999.00

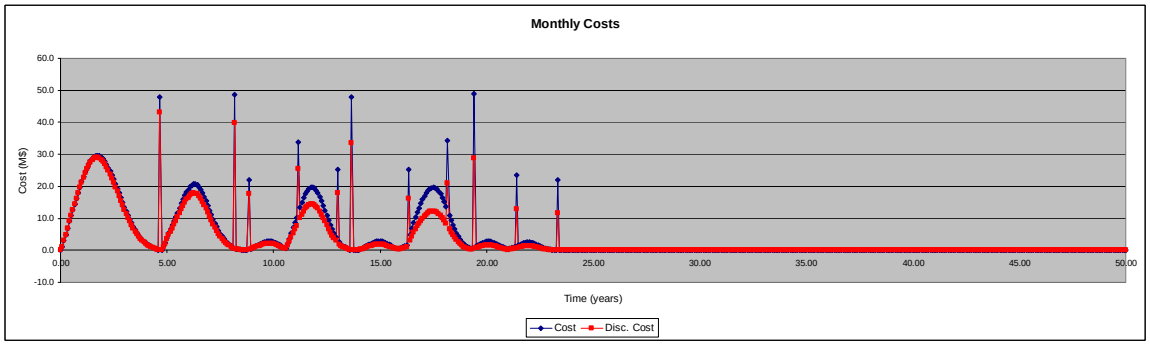
Life-Cycle Analysis:

Architecture	Variation	Num Trials	LCC Mean	LCC Std DEv	LCC Min	LCC Max	LCC 5th	LCC 95th	Disc. LCC	ic. LCC Std D
1	1	100	2620	156	2368	3213	2441	2941	2066	106

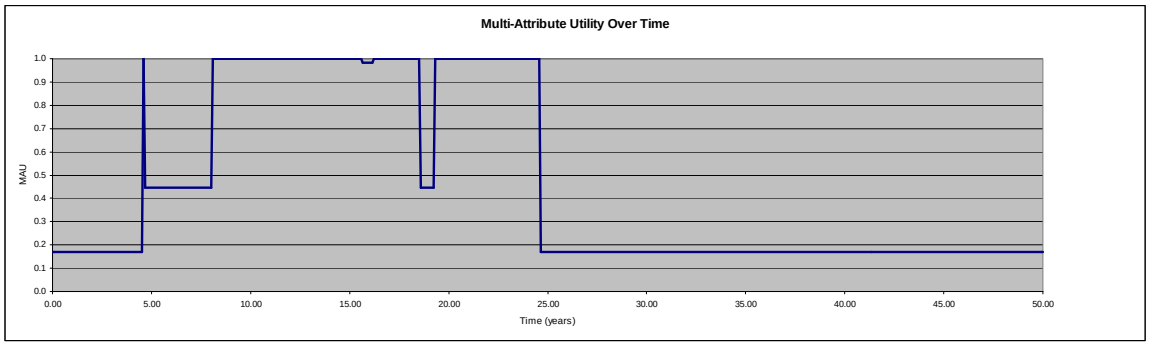
Life-Cycle Timelines:



Life-Cycle Cost Profile:



Life-Cycle Utility Profile:



Hopefully your results agree with these, accounting for variations in the stochastic variables.

The next few sections will describe some of the functions that were skipped or ignored in our two examples.

Design Variables Sheet

Saving and Loading DV Data:

The "Save As:" button allows you to save all the data on the sheet to a text file, and the "Load:" button allows you to load that data back into the sheet:

Save As:	Example_Monolith.txt
Load:	Example_Monolith.txt

This is useful for saving different configurations that correspond to specific missions or spacecraft. Depending on your version of the tool, there may be several save files in the SVMTool folder that represent real NGST spacecraft. The Save As and Load buttons on the other sheets work in much the same way.

Design for Redundancy:

Resource	Parameter	Keyword	Units	Value
Sharing	Design Redundancy	design_for_redundancy		No

Subsystem Design Variables:

Subsystem	Design Variable	Keyword	Units	Value	
Power	Subsystem Name	name		power	
	Cost (NRE)	cost_nre		determine	
	Cost (RE)	cost_re		determine	
	Mass	mass		determine	
	Fractionatable	fractionatable		No	Yes/No
	Solar Cell Type	solar_cell_type		GaAs	Si/GaAs/Multijur
	Battery Type	battery_type		NiH2	NiCd/NiH2/Li
	Bus Voltage	bus_voltage	V	26	

Each subsystem has a set of design variables that (for the most part) can be changed to modify system performance or cost. Some of the DVs pertain to all subsystems, including payloads; other DVs only pertain to one or several specific subsystems. This guide does not include descriptions for every DV. It's hoped that the name for each is sufficiently descriptive, but you can always refer to the Algorithm Description Document, or, as a last result, the MATLAB code.

Subsystem Design Life:

Design Life	design_life	yr	6

reliability_for_design_life		0.9

The design life defines the length of time the subsystem is expected to survive, with a certain probability. The following figure shows how design life is input for each subsystem. The input value of 6 is the subsystem design life, with input units of years. The defined percentage to which the design life is tied comes from the Reliability for Design Life variable. These two values are used to define a reliability curve versus time that the model uses to compute module failures.

Subsystem Probability of Infant Mortality:

Prob. Infant Mortality	prob_infant_mortality		0.01

In addition to failure around design life, a small percentage of subsystems may fail in the first month after launch. The Probability of Infant Mortality defines the likelihood that the subsystem will fail at the beginning of operations, and is a discrete probability.

COTS Modules Sheet

This sheet allows you to input design variables for "off-the-shelf" modules, where the cost, size, weight, and capability are specified instead of being calculated by the tool. The SVMTool will still calculate the Nth production unit cost based on the input Theoretical First Unit (TFU) cost. The SVMTool logic assumes that all propulsion,

ADCS, and Power subsystems are appropriately sized for all of the payload requirements across the architecture, and will only check for the absence of those subsystems.

The following figure shows three example COTS modules:

Off-the-Shelf Modules

Save As: COTS_Examples.txt

Load: COTS_Examples.txt

	Design Variable	Keyword	Units	Module 1	Module 2	Module 3
Module	Module Name	name		Utility Module	Mission Module 1	Mission Module 2
	Description	description		Provides TT&C, C&DH	Total PL mass < 200 kg	Total PL mass < 500 kg
	Cost (NRE)	bus_cost_nre	M$	25	50	100
	Cost (RE)	bus_cost_re	M$	25	50	100
	Dry Mass	dry_mass	kg			
	Wet Mass	wet_mass	kg	250	730	950
	Bus Dry Mass	bus_dry_mass	kg	230	650	850
	Design Life	design_life	yr	5	5	5
	Median Life	median_life	yr	6.5	6.5	6.5
	Required Power	required_power	W	200	700	1000
	Production Unit	num_units_produced				
	Replacement Scheme	replacement_scheme				
	Planning Overlap	planning_overlap				
	Propellant Mass	propellant_mass	kg	20	80	100
	Shareable CDH BW	provided_cdh_bandwidth	Mbps	10	10	0
	Shareable TTC BW	provided_ttc_bandwidth	Mbps	10	10	0

Note that both mass and cost are specified for each module. When the tool performs launch vehicle selection and life-cycle cost calculations, it will use these values. Each module's subsystems are also described farther down the sheet, though subsystem costs and masses will not be used in subsequent calculations.

Launch Vehicles Sheet

This sheet contains the of the launch vehicle cost, performance, and reliability characteristics. Note that at this time payload capacity is related to a single reference orbit. You can choose to consider any launch vehicle by placing an "x" in its consider column.

Environment Variables Sheet

This sheet sets up the life cycle analysis tool. The Number of Trials and RNG Seed set up the Monte Carlo analysis. The Replacement Schedule defines when replacement spacecraft are developed and launched. Perfect replacement puts another spacecraft in at the moment a previous one fails. On Demand starts development when the previous one fails. Planned starts development such that the replacement spacecraft is ideally launched as the previous spacecraft reaches its design life.

See the Algorithm Description Document for more information on how these variables are used in the life-cycle dynamics simulation.

Number of Trials	100	
RNG Seed	1	
Replacement Scheme	planned	none / on_demand / planned
Replacement Spacing (% of DL)	100%	
Launch on Failure	No	Yes/No

Variable	Consider
Development Delay	x
Launch Failure	x
On-orbit Failure	x

Cost Model Sheet

The Cost Model sheet contains CERs for two standard cost models: SSCM07 and an NGST-specific version of USCM8. Any of the numerical values can be modified. Changing any keywords in the coefficient cells (e.g., “battery_type” or “att_control”) will have no effect at this time. You will make the decision as to which cost model to use on the MATLAB Dashboard.

Cost Model Inputs

SSCM - Small Satellite Cost Model

Fiscal Year: 2007

				c	* x1	^ b1	* a2	^ b2	* a3	^ b3
subsystem	**subsystem type**	**keyword**	**type**	**c**	**x1**	**b1**	**a2**	**b2**	**a3**	**b3**
Power	Body-Mounted	power	body_mounted	1975.844	sub_mass	0.175	1.472	battery_type	1	0
Power	Deployed - Fixed	power	deployed_fixed	1548.209	power_bol	0.235	1	0	1	0
Power	Sun-Pointing	power	sun_pointing	847.462	power_bol	0.193	1.587	solar_cell_type	1.989	battery_type
Structure		structure		364.291	sub_mass	0.541	1.498	material	1	0
ADCS	Spin	adcs	spin	21.437	wet_mass	0.927	att_control	-1.184	1	0
ADCS	3-axis	adcs	three_axis	1205.096	sub_mass	0.128	att_control	-0.125	1.628	star_tracker
Propulsion		propulsion		727.231	sub_dry_mass	0.183	2.102	prop_type	3.13	mono_or_bi
C&DH		cdh		1399	sub_mass	0.211	transmit_power	0.306	1	0
TT&C		ttc		720	sub_mass	0.211	transmit_power	0.306	1	0
Thermal	Earth-Orbiting	thermal		164.957	sub_mass	0.482	power_bol	0.13	0.758	array_type
Frac Suite		frac_suite		1000	0	0	0	0	0	0
ATLO		atlo		79.207	wet_mass	0.723	design_life	0.092	1	0
PM/SE	Earth-Orbiting	pmse		0.15	bus_atlo_cost	0.875	bus_dry_mass	0.294	1	0

Schedule Data Sheet

The data in this sheet represents the nominal, normalized spending profile for all modules. The actual profiles for each module will be scaled according to the Minimum Development Time and Development Time Multiplier (i.e., scaling factor):

Minimum Development Time	36	mo
Development Time Multiplier	1.00E-07	mo/$
RE Fraction	0.75	
Schedule Offset for Batch Production	0%	

The RE Fraction value is used in the calculation of schedule length for the second production unit and beyond.

MATLAB Dashboard

Before using the SVMTool, you must first tell it where your files reside on your computer. The Root folder is the location of the MATLAB folder that contains the main SVMTool m-files, and the Subfolders contain various utility files. Click the "Set MATLAB Paths" once you've entered the correct paths:

Set MATLAB Paths	Root:	C:\SVM_Tool\Matlab
	Subfolders:	Util
		Orbit_Util
		Test

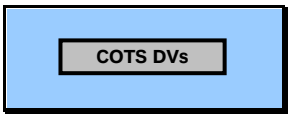
NB: this only works if MATLAB has already been opened via the "startmatlab" button in Excel. The MATLAB directory window should display the Root folder.

Once the inputs are set and the MATLAB interface is configured, the tool is ready to be run. To receive valid output, the buttons must be executed sequentially. Steps three and six give options on how to build the architecture and evaluate the cost, and only one option per step should be executed. Steps three through nine produce output into the blue Excel sheets, which can be analyzed individually.

The plot button will produce three graphs. The first graph is average life-cycle cost of an architecture versus normal standard deviation of that cost. The second graph is a plot of individual life-cycle cost of every trial of each architecture. The final graph is a histogram of those individual life-cycle costs.

The timeline buttons produce output on the Life-Cycle Timelines sheet. This output shows all the events in the specified timeline, the time at which they occurred, and the cost that was incurred by them.

<u>COTS DVs Print-out:</u>

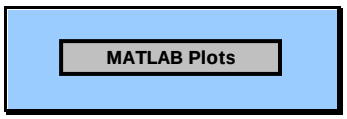


The COTS DVs button will produce the outputs on the COTS-Format Output sheet. These outputs can then be re-inserted into the COTS Modules input sheet, to represent new COTS Modules. This is a round-about process of first designing, and then using user-defined off-the-shelf modules. Some tweaking of the output values may be required to make the data compatible with the COTS Modules input sheet format:

Off-the-Shelf Modules

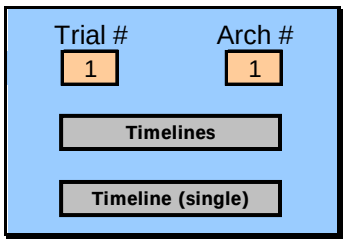
	Design Variable	Keyword	Units	Module 1
Module	Module Name	name		
	Description	description		
	Cost (NRE)	bus_cost_nre	M$	85.73569359
	Cost (RE)	bus_cost_re	M$	82.25812931
	Dry Mass	dry_mass	kg	2626.628498
	Wet Mass	wet_mass	kg	2804.997098
	Bus Dry Mass	bus_dry_mass	kg	1666.146654
	Design Life	design_life	yr	4.967589445
	Median Life	median_life	yr	6.458443183
	Required Power	required_power	W	1739.453406
	Production Unit	num_units_produced		0
	Replacement Scheme	replacement_scheme		
	Planning Overlap	planning_overlap		
	Propellant Mass	propellant_mass	kg	178.3686005
	Shareable CDH BW	provided_cdh_bandwidth	Mbps	12.0548
	Shareable TTC BW	provided_ttc_bandwidth	Mbps	12.0548

MATLAB Plots:



The plot button will produce four graphs. The first graph is average life-cycle cost of an architecture versus normal standard deviation of that cost. The second graph is a plot of individual life-cycle cost of every trial of each architecture. The third graph is a histogram of those individual life-cycle costs. The final graph is a plot of the individual multi-attribute utility of every trial of each architecture against the life cycle cost of that trial.

Timeline Plots:



Known Issues

Not all run-time exceptions in MATLAB are handled and passed through to Excel. In those cases, the run may fail silently.

Some examples of input errors that may not be caught or reported:

- Module designs that do not close
- Modules that exceed the maximum capacity of any launch vehicle
- Zero values that result in divide-by-zero errors
- Numeric values in place of text values or vice versa

SVMTool-Recognized Units, with Abbreviations

Note that SVMTool input units are NOT case-sensitive:

Measurement	**Unit**
Angle	radian (default), deg, degree, degrees, arcsec
Distance	meter (default), km, kilometer, kilometers, cm, centimeter, centimeters
Mass	kilogram (default), g, gram, grams, lb, lbs, lbm
Time	second (default), min, minute, minutes, hr, hour, hours, dy, day, days, mo, month, mos, months, yr, year, years
Speed	m/s (default), km/s, kps
Volume	m^3 (default), l, liter, liters
Data Rate	bps, kbps, mbps, gbps
Data Storage	bits, kb, mb, gb
Frequency	hz, khz, mhz, ghz
Cost	$, k$, m$, g$
Fractions/Ratios	Unitless (default), percent, %