|  |
| --- |
|  |
| OPUS User Guide |
| OPUS (Orbital Debris Propagators Unified with Economic Systems): An Integrated Assessment Model for Satellites and Orbital Debris |

Contents

[1. Necessary Software 2](#_Toc173589534)

[2. How to Download OPUS 3](#_Toc173589535)

[3. How to Set Up Your Work Folder 4](#_Toc173589536)

[4. Specifying a Scenario 4](#_Toc173589537)

[5. Running OPUS 5](#_Toc173589538)

[6. Results 7](#_Toc173589539)

[7. Single Scenario Analytics 8](#_Toc173589540)

[8. Comparison Analytics 10](#_Toc173589541)

[List of Modifiable OPUS Parameters 12](#_Toc173589542)

[List of Output Data 14](#_Toc173589543)

[List of Single Scenario Analytics 15](#_Toc173589544)

[List of Comparison Analytics 18](#_Toc173589545)

[Appendix I: Setting Up R to Run from a Bash File 19](#_Toc173589546)

This user guide was authored by Joey Kilpatrick.

# Necessary Software

Before getting started, you will need a few pieces of software to run the program. It is best to download and set these programs up before downloading OPUS. Namely:

1. MatLab (<https://www.mathworks.com/products/matlab.html>)
   1. You will either need to download MatLab via a single sign on service through your organization or you will have to purchase MatLab before downloading.
   2. Once you download Matlab, a few additional add-ons will need to be downloaded into it for OPUS to run:
      1. Symbolic Math Toolbox
      2. Optimization Toolbox
      3. Global Optimization Toolbox
      4. Parallel Computing Toolbox
   3. To download each add-on, open Matlab and navigate to the Add-ons button to the far-right, in the main toolbar (see figure 1 below.)

|  |
| --- |
| **Figure 1: Location of Add-Ons in Matlab Toolbar** |
| A screenshot of a computer  Description automatically generated |

* 1. Click the arrow under “Add-Ons”, then click “Get Add-Ons.” A new window will open, in the upper right corner is a search box, search for each add-on in part b above and install them. Once all are installed, restart your machine to use them in Matlab.

1. R (download the program for Windows here: <https://cran.r-project.org/bin/windows/base/>)
   1. If you have not run R from a shell script before, be sure to set this up. You will need to make sure a path is set within your system settings to allow for Rscript to be called from Git or another bash program. See Appendix I for a detailed explanation of how to do this.
   2. R also needs a few packages installed into it:
      1. Patchwork
      2. Tidyverse
   3. To download each package, make sure you have the latest version of R, this guide is written using version 4.4.0.
      1. Open the R program file, within Windows this should open in Command Prompt.
      2. Type in the line: install.packages(c(‘tidyverse’,’patchwork’)) to begin the installation process.
      3. R will then open a window asking you to select a server location, select the first one and click “Ok”. The packages should then download and install themselves. Once this is done, you will need to restart your machine.
2. GIT (download here: <https://git-scm.com/download/win>) or another bash software that can run .sh file formats.
   1. If you are installing GIT for the first time: select the appropriate bit version of Git for your machine, once it is downloaded, open the file. The installer has a series of options, click “Next” without changing any of the default options presented until the program begins installation.
3. A zip extraction program.
4. Also note the number of cores within your machine’s processor [referred to as ‘workers’ throughout the program]. To find the number of cores, follow the steps below.
   1. Press Ctrl + Shift + Esc to open Task Manager.
   2. Select the Performance tab to see how many cores your machine has (see figure 2 below).

|  |
| --- |
| **Figure 2: Finding Number of ‘Workers’** |
|  |

# How to Download OPUS

1. Go to the Github webpage: <https://github.com/akhilrao/OPUS>
2. Click the green “<> Code” button and a drop-down should appear (see figure 3 below).
3. Under that drop-down, click “Download ZIP” and the file will download to where you have set your browser’s download folder.

|  |
| --- |
| **Figure 3: Downloading OPUS from Github** |
| A screenshot of a computer  Description automatically generated |

# How to Set Up Your Work Folder

1. Set aside a folder in your machine for OPUS.
2. Go to where the files were previously downloaded.
3. The files are downloaded in ZIP format, you will need to extract them. Using either your machine’s built-in ZIP extraction software or a third-party program, extract them to the file location set aside for OPUS.

# Specifying a Scenario

1. Scenarios are set up using excel for inputs. The program can run several programs in succession, see step 2 in this section. The file “OPUS-main\scenarios\parsets\example\_parameters” can be used as a starting point for parameter specification. For a complete list of changeable parameters, their descriptions, and units, see the section List of Modifiable Opus Parameters.
   1. Please take note that certain parameters are restricted to being between 0-1, units are noted in the summary of parameter variables.
   2. You do not need to put in a value for all parameters. Parameters that do not have a value will default to benchmark values.
   3. Once you have specified values for parameters of interest, be sure to delete the rows within your excel file that have no values in the “parameter\_value” column.
   4. After you have completed the file, save it as a new file in .csv format within the “OPUS-main/scenarios/parsets” folder, with a file name that summarizes the scenario parameters. Take note of the file name, it will be needed when running OPUS.
2. [Optional] To run several scenarios in succession you will need to create several parameter files. You will need to name each file uniquely and save them in the “OPUS-main/scenarios/parsets” folder.
3. Outside of the parameter file, one parameter, the number of years for the model to simulate, is changed within the conductor file. See optional step 4 under “Running OPUS” for how to change this parameter.
   1. Note: if you set a parameter that has a years unit the model horizon will need to be, at a minimum, the number of years set in your parameter value.

# Running OPUS

1. Go to the “conductor” file under the “main-OPUS” folder. Right click the file and make sure to select “Edit in [software]”. {Warning: Double clicking the file will start the program}. The software should be some form of notepad. If “Edit in [software]” is not an option, select “Open with…” and then select your notes program. Under “2. SCENARIO DEFINITIONS” you will need to change the scenario file names to run your specified scenario(s). If you have one scenario, replace the default “25yr-rule-parset” under “scenario\_files” with the name of your previously created scenario file and delete the line below this, "scenarios/parsets/25yr-rule-tax-0.5-parset.csv" (see figure 4 below).
   1. If you have multiple scenario files to run, you can add them in this section. Press enter and tab twice then add in your scenario file name in this format: “scenarios/parsets/[your scenario file name]”. Do this for each scenario file you’d like to run in succession; there is no limit to how many scenarios can be run in succession.
   2. Note: it is recommended to keep the “benchmark’ scenario here, in case you would like to run comparisons analytics against the benchmark scenario later, but it is not required for the program to run.

|  |
| --- |
| **Figure 4: Changing Default Scenarios** |
| A computer screen with a red box  Description automatically generated |

1. Further, under “2. SCENARIO DEFINITIONS” you will need to change “n\_workers” to the number of cores your processor has (see figure 5 below). If you have set up Matlab to run multiple workers over processing cores, that number can be used here as well.

|  |
| --- |
| **Figure 5: Changing Default Number of Workers** |
| A screenshot of a computer program  Description automatically generated |

1. Once you have completed steps 1 and 2, you are ready to save a new conductor file. Before doing so, review optional steps 4 below to see if it pertains to your scenario(s). Save the conductor as a new file with a name describing your scenario(s); select “All files” from the “Save as type” dropdown; and be sure to include “.sh” at the end of the file name to make it a shell script file (For example: conductor\_scenario\_0.sh). Exit notepad and double click your newly created conductor file to run the program.
   1. Note that OPUS can take a while to run, anywhere upwards of an hour, depending on your machine and the number of scenarios you are running in succession.
2. [Optional] You can change the model horizon (the number of years for the model simulation to run over) under “2. SCENARIO DEFINITIONS”. The variable is named “model\_horizon” and the current default is for the model to run over 35 years. Change the horizon by replacing 35 with the number of years desired (see figure 6 below).

|  |
| --- |
| **Figure 6: Changing Model Horizon** |
| A screenshot of a computer program  Description automatically generated |

# Results

1. Opus automatically saves results associated with your scenarios under a folder with a four-word, human readable name. {Warning: If you are running the same scenario more than once, OPUS will overwrite the results within the previously generated, four-word named folder associated with that scenario}. You will need to manually associate the four-word name with the scenario you have specified.
   1. To do so, go to the “scenarios folder” and double click the four-word folder, newly created by OPUS, to open it (see figure 7 below).
   2. Once inside this folder, there will be a file with the same name as a scenario you previously specified and ran the OPUS program with (see figure 8 below). For example, in the figures below, the “generous-turtle-tastes-Hibiscus” results come from the “25yr-rule-tax-1-parset” previously specified in an excel file under “scenarios/parsets”. Note: this scenario file is not included in the version of OPUS downloaded from GIT.
   3. It is recommended that you create a dictionary for yourself relating these four-word names with the names of the scenarios you specified in the “conductor” file.

|  |
| --- |
| **Figure 7: Four-Word, Human Readable Results Folder Location** |
| **A screenshot of a computer  Description automatically generated** |

|  |
| --- |
| **Figure 8: Scenario Associated with Four-Word, Human Readable Name** |
|  |

1. Once you have noted the four-word name that the results for your specified scenarios are saved under, navigate to its results folder. In this folder will be a set of excel files with output data, see the “List of Output Data” section for a list of outputs and descriptions. There are further R programs that will conduct analyses on this output, both for a single scenario and for pair-wise comparisons of scenarios. See the section below, “List of Single-Scenario Analytics” and “List of Comparison Analytics” to see the output from these programs. For how to run these analyses programs, see the sections immediately succeeding this section, “Single Scenario Analytics” and “Comparison Analytics.”

# Single Scenario Analytics

1. Locate the file “Single Scenario Analytics” under your “OPUS-main” folder, the same location as the “conductor” file. Right click the file and make sure to select “Edit in [software]”. {Warning: Double clicking the file will start the program}. The software should be some form of notepad. If “Edit in [software]” is not an option, select “Open with…” and then select your notes program. A few things will need to be changed in this file to produce analytics from each of your previously produced OPUS results.
   1. If you changed the model\_horizon, the model horizon will need to be updated to match what you specified in your previously created conductor file. Under “1. INITIAL DEFINITIONS”, locate “model\_horizon=35” and change the 35 to match the model horizon value you previously specified (see figure 9 below).

|  |
| --- |
| **Figure 9: Changing Model Horizon to Match Your Conductor File** |
|  |
|  |

1. Under “2. SCENARIO DEFINITIONS” you will need to change the four-word names to the four-word names associated with your previously produced scenarios results. If you have one scenario, replace the default “staying-coral-stew-Soybean” under “scenario\_names” with the four-word name of your previously created scenario file, be sure to include quotes around the name, and delete the lines below this, "generous-turtle-tastes-Hibiscus" and "loving-dingo-vocalize-Horseradish" (see figure 10 below).
   1. If you have multiple scenarios that you would like analytics for, there is no need to create multiple single scenario analytics files, you can add each four-word name in this section. Click after the right quote on your first scenario, press enter and tab twice, then add in your scenario file name in the same format as the first scenario: “four-word-generated-scenario”. Do this for each four-word name; there is no limit to how many scenarios you can run analytics on.

|  |
| --- |
| **Figure 10: Changing Default Four-Word, Human Readable Scenarios** |
| A screenshot of a computer program  Description automatically generated |
|  |

1. Once you have completed steps 1 and 2, you are ready to save a new single scenario analytics file. Save as a new file with a name describing your scenario(s); select “All files” from the “Save as type” dropdown and be sure to include “.sh” at the end of the file name to make it a shell script file (For example: Single Scenario Analytics Test Run.sh). Exit notepad and double click your newly created single scenario analytics file to run the program.
   1. The analytics output for each of your scenarios will be saved under the associated four-word name folder. See “List of Single Scenario Analytics” for a description of the output.

# Comparison Analytics

1. The comparison analytics file will compare two scenarios whose output data was previously generated by OPUS. Before running a comparison, be sure to run each scenario you would like to compare through the single scenario analytics file. Necessary data for the comparison is produced from the single scenario file. See the “List of Comparison Analytics” section for a list of graphs that will be produced.
2. Locate the file “Comparison Analytics” under your “OPUS-main” folder, the same location as the “conductor” file. Right click the file and make sure to select “Edit in [software]”. {Warning: Double clicking the file will start the program}. The software should be some form of notepad. If “Edit in [software]” is not an option, select “Open with…” and then select your notes program.
3. First, locate “1. INITIAL DEFINITIONS”. Directly above “2. SCENARIO DEFINITIONS”, you will need to specify a file name where you want the comparison graphs to be saved. Replace “comparison-benchmark-eqm-satfeedback”, see figure 11, with a descriptive name of your comparison, making sure to include quotations around the folder name.

|  |
| --- |
| **Figure 11: Changing Comparison Output Location** |
| A screenshot of a computer program  Description automatically generated |
|  |

1. Throughout this step, refer to figure 12. Locate the section headed “2. SCENARIO DEFINITIONS.” Below is a list of parameters that will need to be updated according to what you would like to compare between scenarios.
   1. First, you will need to define the length of the model horizon previously specified within the conductor file for your scenarios. Replace the number after “model\_horizon=” with the length of the model horizon in both scenarios.
   2. Below where you specified the model horizon, you will need to specify the first scenario for comparison. Replace "generous-turtle-tastes-Hibiscus" after “scenario\_name\_1=” with the randomly generated four-word name of your first scenario. Make sure there are quotations around the scenario’s name.
   3. Below where you specified the first scenario, you will need to specify the second scenario for comparison. Replace "loving-dingo-vocalize-Horseradish" with the randomly generated four-word name of your second scenario, making sure there are quotations around the name.
   4. Below where the second scenario name is specified, you will need to specify the kind of satellite data you would like to use for the first scenario, either “sat\_feedback” for satellite feedback data or “equilibrium” for the equilibrium data. After “sat\_data\_1=” replace “equilibrium” with “sat\_feedback” if you would like to use satellite feedback data or leave it if you want to compare equilibrium data. Be sure to include quotations around the data type.
   5. Below where you specified the data for the first scenario, you will need to specify the data type for the second scenario. After “sat\_data\_2=” you can either leave “equilibrium” or replace it with “sat\_feedback”. Quotations are need around the data type here too.
   6. After specifying the satellite data for both scenarios, below should be “Description for 1st scenario”. After “description\_1=” should be “Satellite feedback behavior”, replace “Satellite feedback behavior” with a description of the first scenario, including quotations.
   7. Below the description of the 1st scenario should be a place for the description of the 2nd scenario. Replace “Open-access behavior” after “description\_2=” with a description of your second scenario. Again, be sure to include quotations around the description.
   8. Finally, we need to specify the name of the folder where the comparison output will be saved. After “compare\_folder=” replace "comparison-benchmark-eqm-satfeedback", with the name of the folder specified in step 2 of this section. Be sure to include quotations around the folder name.

|  |
| --- |
| **Figure 12: Updating Parameters for Scenario Comparisons** |
|  |
|  |

1. Once you have completed steps 3 and 4, you are ready to save a new comparison analytics file. Save as a new file with a name describing your comparison; select “All files” from the “Save as type” dropdown and be sure to include “.sh” at the end of the file name to make it a shell script file (For example: Comparison Analytics Test Run.sh). Exit notepad and double click your newly created comparison analytics file to run the program.
   1. Note that unlike the single scenario analytics file, this file cannot run multiple comparisons. You will need to create a new comparison file and run it for each comparison.

# List of Modifiable OPUS Parameters[[1]](#footnote-2)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter Type | Parameter Name | Parameter Description | Benchmark Value | Units |
| Economics | satLifetime | Active satellite lifetime | 5 | Years |
| Economics | disposalTime | Disposal regulation, number of years at which the satellite must be disposed of | 5 | Years |
| Economics | discountRate | Discount rate, opportunity cost of funds | 0.05 | Percent per Year |
| Economics | intercept | Revenue of satellite with no competition, maximum willingness-to-pay for service of satellite | 750000 | Dollar per Year |
| Economics | coef | Linear revenue coefficient of satellite, willingness-to-pay from a marginal satellite | 100 | Dollar per Satellite per Year |
| Economics | tax | Shell-specific orbital-use fee in each period | 0.0 | Percent per Satellite per Year |
| Economics | delta\_v\_cost | Cost of delta-v at a given altitude, the total dollar cost required to maintain a satellite in its target orbit, including any necessary maneuvers over its lifetime. | 1000 | Dollar per Meter per Satellite |
| Economics | lift\_price | Cost per kilogram of accessing low earth orbit, multiplied by the mass of the satellite. | 5000 | Dollar per kilogram |
| Constellation | n\_constellations | Number of satellite constellations (such as Starlink or OneWeb) | 2 | No unit |
| Constellation | location\_index | Indicates altitude location of constellations. Each shell is 35km wide. | [10,29] | Altitude (km) = 200 + 35\*location\_index |
| Constellation | final\_size | The targeted final size of constellations. | [3000,300] | No unit |
| Constellation | linear\_rate | The maximum launch rates the constellations can achieve, reflecting launch capacity available. | [1500,150] | No unit |

# List of Output Data

|  |  |
| --- | --- |
| File Name[[2]](#footnote-3) | Description |
| x-x-x-x --collision-parameters | Probabilities of collision for both fringe and constellation satellites across location indices. |
| x-x-x-x -MOCAT-equilibrium-35yrs\_\_out-data | Number of fringe and constellation satellites across model horizon and altitude, launch rates for fringe and constellation satellites across model horizon and altitude, and amount of debris and number of derelict satellites across horizon and altitude for the “equilibrium” condition, which uses open-access conditions to determine launch patterns. |
| x-x-x-x -MOCAT-sat\_feedback-35yrs\_\_out-data | Number of fringe and constellation satellites by altitude, launch rates for fringe and constellation satellites by altitude, and amount of debris and number of derelict satellites by altitude for the “satellite feedback” condition, which maintains the initial population of satellites. |
| x-x-x-x--parameters | A list of parameters that were used to calculate results in the model. |
| fragmentation-parameters | A matrix of the number of fragments generated in each collision between fringe, constellation, derelict satellites and debris, specific to a generated scenario. |

# List of Single Scenario Analytics

|  |  |  |
| --- | --- | --- |
| File Name[[3]](#footnote-4) | Type | Description |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_ecob | Graph | Environmental Consequence of Orbital Breakup Index over your model horizon timeframe using equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_heatmap-time-plots-D | Graph | Location and number of derelict satellites over your model horizon timeframe using equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_heatmap-time-plots-N | Graph | Location and amount of debris over your model horizon timeframe using equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_heatmap-time-plots-S | Graph | Location and number of constellation satellites over your model horizon timeframe using equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_heatmap-time-plots-Su | Graph | Location and number of fringe satellites over your model horizon timeframe using equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_heatmap-time-plots | Graph | 6 charts, number of constellation and fringe launches over your model horizon timeframe and by altitude, number of fringe and constellation satellites over your model horizon timeframe and by altitude, and number of debris and derelict satellites over your model horizon timeframe and by altitude using equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_line-cost | Graph | Line plot of cost in millions of dollars of a satellite, across altitudes using equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_scatter-accounting-profits-0 | Graph | Account rate of return, in percent, of a satellite across altitudes in year 0 using equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_scatter-accounting-profits-all | Graph | Account rate of return, in percent, of a satellite across altitudes and over your model horizon timeframe using equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_scatter-cost | Graph | Scatter plot of cost in millions of dollars of a satellite, across altitudes using equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_scatter-lifetime-loss-cost | Graph | Lifetime loss cost of a satellite across altitudes using equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_scatter-stationkeeping-cost | Graph | Station keeping cost of a satellite across altitudes using equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_ssr-index | Graph | Space sustainability rating index of derelict, constellation, and fringe satellites over your model horizon timeframe using equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_total-collision-probability | Graph | Number of expected collisions of derelict, constellation, and fringe satellites over your model horizon timeframe using equilibrium data |
| Su-D-x-x-x-x-MOCAT-equilibrium-35yrs\_\_heatmap-time-plots | Graph | Two plots, one of fringe satellites and another of derelict satellites, across altitudes and over your model horizon timeframe using equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_aggregate-metrics-paths | Excel file | Number of objects, collisions, mean chance of collision, ECOB index, expected mass loss, SSR Index, and normalized SSR Index by derelict, fringe, and constellation satellite types over your model horizon timeframe, for equilibrium data |
| x-x-x-x-MOCAT-equilibrium-35yrs\_\_summary-stats | Excel file | Summary statistics by each satellite type, for cost, gross revenue, amongst other parameters, for equilibrium data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_ecob | Graph | Environmental Consequence of Orbital Breakup Index over your model horizon timeframe using satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_heatmap-time-plots-D | Graph | Location and number of derelict satellites over your model horizon timeframe using satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_heatmap-time-plots-N | Graph | Location and amount of debris over your model horizon timeframe using satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_heatmap-time-plots-S | Graph | Location and number of constellation satellites over your model horizon timeframe using satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_heatmap-time-plots-Su | Graph | Location and number of fringe satellites over your model horizon timeframe using satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_heatmap-time-plots | Graph | 6 charts, number of constellation and fringe launches over your model horizon timeframe and by altitude, number of fringe and constellation satellites over your model horizon timeframe and by altitude, and number of debris and derelict satellites over your model horizon timeframe and by altitude using satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_line-cost | Graph | Line plot of cost in millions of dollars of a satellite, across altitudes using satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_scatter-accounting-profits-0 | Graph | Account rate of return, in percent, of a satellite across altitudes in year 0 using satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_scatter-accounting-profits-all | Graph | Account rate of return, in percent, of a satellite across altitudes and over your model horizon timeframe using satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_scatter-cost | Graph | Scatter plot of cost in millions of dollars of a satellite, across altitudes using satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_scatter-lifetime-loss-cost | Graph | Lifetime loss cost of a satellite across altitudes using satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_scatter-stationkeeping-cost | Graph | Station keeping cost of a satellite across altitudes using satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_ssr-index | Graph | Space sustainability rating index of derelict, constellation, and fringe satellites over your model horizon timeframe using satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_total-collision-probability | Graph | Number of expected collisions of derelict, constellation, and fringe satellites over your model horizon timeframe using satellite feedback data |
| Su-D-x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_heatmap-time-plots | Graph | Two plots, one of fringe satellites and another of derelict satellites, across altitudes and over your model horizon timeframe using satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_aggregate-metrics-paths | Excel file | Number of objects, collisions, mean chance of collision, ECOB index, expected mass loss, SSR Index, and normalized SSR Index by derelict, fringe, and constellation satellite types over your model horizon timeframe, for satellite feedback data |
| x-x-x-x-MOCAT-sat\_feedback-35yrs\_\_summary-stats | Excel file | Summary statistics by each satellite type, for cost, gross revenue, amongst other parameters, for satellite feedback data |

# List of Comparison Analytics

|  |  |
| --- | --- |
| File Name | Description |
| expected-maximum-welfare-comparison | Expected maximum welfare in millions of dollars across time. |
| normalized-ssr-comparison | Normalized SSR index across time, for derelict, constellation, and fringe satellites across time. |
| ssr-comparison | Aggregate SSR index across time, for derelict, constellation, and fringe satellites across time. |
| total\_sats-comparison | Number of derelict, constellation, and fringe satellites across time. |

# Appendix I: Setting Up R to Run from a Bash File

* + - 1. Navigate to where you have downloaded R, then click into the bin folder (R\R-4.4.0\bin). Copy the pathway to this folder.
      2. Once you get the folder path, search for “environmental variables” under the Windows start menu search function. Click “Edit the system environment variables.”
      3. A new window should pop up. Click “Environmental Variables” in the bottom right-hand of the window (see Figure 9 below).

|  |
| --- |
| **Figure 9: Environmental Variables** |
| A screenshot of a computer program  Description automatically generated |

* + - 1. Locate ‘Path’ under System Variables and click “Edit” (see figure 10 below).

|  |
| --- |
| **Figure 10: Locating Path** |
|  |

* + - 1. A new window will open. Check to see if the pathway you copied to the bin file of R is in the list of environment variables. If not, you will need to add it. Click “New” and then paste the copied path (see figure 11 below). Once it is added to the list, click “Ok” on each of the three windows that have been opened and restart your machine.

|  |
| --- |
| **Figure 11: Adding a New Environmental Variable to Path** |
|  |

1. In addition to the variables listed here, the MOCAT model has several variables that are editable in the same way as those listed here, via excel file parameter specification. Please see the Matlab “MOCAT4S\_VAR\_Cons” file under the MOCAT4S folder for what parameters can be changed. For additional information on MOCAT4S and its parameters, please see the MIT ARCLab MOCAT Github: <https://github.com/ARCLab-MIT/MOCAT-SSEM> . [↑](#footnote-ref-2)
2. The x-x-x-x prefix on each file name represents the randomly generated four-letter name OPUS has given your scenario. See the ‘Results’ section for more detail on the randomly generated name. [↑](#footnote-ref-3)
3. The x-x-x-x prefix on each file name represents the randomly generated four-letter name OPUS has given your scenario. [↑](#footnote-ref-4)