

OPTIMAL SPACECRAFT TRAJECTORIES VIA VISUAL TRADE SPACE EXPLORATION

Daniel D. Jordan^{*}
David B. Spencer[†]
Timothy W. Simpson[‡]
Michael A. Yukish[§]
Gary M. Stump^{**}

Recent advances in computing power and speed allow designers to simulate thousands, if not millions, of design alternatives more cheaply and quickly than ever before. These advancements provide new opportunities to revolutionize trade space exploration for complex dynamic systems in the aerospace industry, among others. In this paper, we demonstrate our multi-dimensional data visualization software, the Applied Research Laboratory (ARL) Trade Space Visualizer (ATSV), to search for optimal impulsive trajectories in a two-burn plane change spacecraft maneuver. This problem is formulated as a multi-objective optimization problem where it is desirable to explore various competing objectives.

INTRODUCTION

The continual improvements in computing power speed allow today's engineers and designers to evaluate more design alternatives than ever before. Rapid visualization and analysis combined with model integration can allow decision makers to explore a design space quickly and efficiently. This concept of trade space visualization is particularly useful for the design of complex systems, such as automobiles, aircraft, and spacecraft.

Engineers now recognize the importance of visualizing possible designs beforehand, rather than trusting black-box algorithms to optimize multi-objective decisions. This new paradigm is dependent upon using an *a posteriori* approach to design, in which designers utilize their own experience and intuition to "shop" through a set of possible designs.

^{*} Graduate Research Assistant, Department of Aerospace Engineering, The Pennsylvania State University, 229 Hammond Bldg., University Park, PA 16802 USA

[†] Associate Professor, Department of Aerospace Engineering, The Pennsylvania State University, 229 Hammond Bldg., University Park, PA 16802 USA; Senior Member, AAS; Associate Fellow, AIAA.

[‡] Professor of Mechanical and Industrial Engineering, 314D Leonhard Building, The Pennsylvania State University, University Park, PA 16802 USA; Senior Member, AIAA.

[§] Head, Product and Process Division, Applied Research Laboratory, State College, PA 16804 USA

^{**} Research Assistant, Applied Research Laboratory, State College, PA 16804 USA

Balling [1], credited with the “Design by Shopping” paradigm, found that traditional optimization-based design processes often leave designers unsatisfied with their results. Balling states that this is most likely a consequence of the initial design problem being improperly formulated: “the objectives and constraints used in optimization were not what the owners and stakeholders really wanted...in many cases, people don’t know what they really want until they see some designs”. Furthermore, Wilson and Schooler [2] have found that people often do worse at decision making tasks when asked to analyze the reasons for their preferences. Consequently, trade space exploration intends to take advantage of a designer’s intuition and experience to allow the user to choose designs as well or better than more costly alternatives.

Within the visualization community, interactive optimization-based design methods fall mainly into the category of computational steering whereby the user (i.e., designer) interacts with a simulation during the optimization process to help “steer” the search process toward what looks like an optimal solution [3]. The designer observes some sort of a visualization of the optimization process and then uses intuition, heuristics, experience, or some other method to adjust the design space to move toward something that may not have been intuitive at the beginning of the simulation. The importance of using visualization together with optimization is discussed at length by Messac and Chen [4]: “If effectively exploited, visualizing the optimization process in real time can greatly increase the effectiveness of practical engineering optimization.” Ng [5] advocates the use of data visualization and interaction to support the designer in making informed decisions and tradeoffs during multi-objective optimization. Finally, Eddy and Mockus [6] argue that visualization should be considered as a solution tool rather than a means to present results.

This paradigm can be applied to various astronautics problems, including the optimal orbit transfers that are the focus in this paper. Throughout this work, the Applied Research Lab Trade Space Visualizer (ATSV), described next, is utilized as a tool for evaluating a design space and applied to an example in orbital mechanics.

ATSV BACKGROUND

Multi-Dimensional Trade Space Visualization in ATSV

The ATSV is a Java application that displays multi-dimensional trade spaces using glyph, histogram, scatter, scatter matrix, and parallel coordinate plots [7]. Since the ATSV is developed in Java, its cross-platform compatibility allows relatively simple model integration, making it an advantage over most commercially available software systems.

The ATSV also offers linking between plot types, which helps users recognize underlying patterns and relationships that exist in the data. The data can be brushed and preference shaded with respect to any objective function. Pareto filtering can be

implemented in multiple dimensions to reduce the data set to the most ideal solutions with respect to specified preferences of objective functions.


Static Datasets, Dynamic Datasets, and Sampling

The ATSV can explore design spaces statically or dynamically. Static data sets, generated from an external model, are read from comma-, space-, or tab-delimited text or spreadsheet files (filename.csv). Objective functions and their values for each population member are organized within these files as shown in Table 1.

Table 1: Structure of Static (.csv) Spreadsheet Data Files

Pop member	Objective Function #1	Objective Function #2	Objective Function #3	Objective Function #4	...	Objective Function #n
1	<value>	<value>	<value>	<value><value>
2	<value>	<value>	<value>	<value><value>
3	<value>	<value>	<value>	<value><value>

Dynamic datasets are generated by integrating the ATSV with external code via a “wrapper” that allows the ATSV to directly sample design inputs to generate new design alternatives and visualize objective function outputs. By clicking File / Link to Exploration Engine, this dynamic data generation and visualization can be implemented. The advantage to integrating external code with the ATSV primarily lies in the ability to specify five types of visual steering commands to generate new data by random sampling, manual sampling, attractor-based sampling, preference-based sampling, and Pareto sampling [3]. Random sampling will randomly generate values for each design input between specified limits with uniform or normal distributions. This type of sampling is designed to populate the objective space with random points, possibly to attempt to identify trends. Manual sampling allows the user to precisely pick design input values and visualize the corresponding single output in the objective space. Preference based sampling will generate points in a region of interest. This region is specified by the user by assigning weights on objectives of importance. Pareto sampling will generate solutions along a Pareto frontier. The Pareto front must be defined by the user by specifying a direction of preference for each objective function. The ATSV’s attractor based sampling creates points in the objective space close to a desired location, specified by an “attractor”.

Attractors, specified in the interface by the  icon, generate points near a specified n-dimensional location in the trade space. These attractors, also referred to as point samplers, can be used to create a cluster of points in the trade space in order to “fill gaps” or explore a specific space in more detail. An evolutionary algorithm, specifically Differential Evolution [8], is used to guide the sampling process in order generate points in the objective space from n-dimensional discrete and continuous inputs. Details on the

implementation of Differential Evolution and the visual steering commands within ATSV can be found in Ref. [3].

Plots

The three-dimensional glyph plot is capable of displaying eight dimensions simultaneously: three spatial dimensions as well as point size, color, transparency, orientation, and text overlay. Using the mouse, the user can navigate through the dataset, plotting any objective function with respect to any of the seven dimensions. The two-dimensional scatter plot is similar to the glyph plot, but allows the user to visualize only two spatial dimensions and a third dimension represented by color. The importance of these two plot types with regards to objective space exploration is the primary focus of the orbital mechanics example presented in this paper.

The ATSV can also display scatter matrix, parallel coordinates, and histogram plots. The scatter matrix shows thumbnail images of 2D scatter plots showing each objective function versus every other objective function. This allows the user to quickly locate and recognize patterns in objective function data. The parallel coordinates plot shows graphical representations of relationships between design inputs and objective functions as lines between values. It is useful for identifying connections and relationships between variables of interest that might otherwise be difficult to discover.

Statistical distributions of values of design inputs and objective functions are visualized with the histogram plot. This type of visualization is useful for understanding how design input distributions may affect objective functions of interest. The two-dimensional histogram plot allows the user to visualize distributions of two variables of interest simultaneously.

Brush and Preference Controls

For each objective function of interest, there is the ability to view only specified points between certain feasible ranges by “brushing” the dataset [9]. Preference controls allow the user to specify a desired minimization or maximization of specific objective functions, permitting the ATSV to filter the corresponding Pareto front.

As an example, the ATSV can be used to explore the decision space populated with many different automobiles. Controls can be added for the horsepower and highway efficiency objective functions. If the decision maker has knowledge of what range of these values are feasible, the minimum and maximum values can be entered to “brush” out infeasible values. For example, the user can specify reducing the visible dataset to automobiles with horsepower between 150 and 250, and highway efficiency above 20 mpg. Furthermore, the preferences on these two variables would most likely be set to maximize horsepower and highway efficiency simultaneously.

The resulting brushed dataset showing highway efficiency versus horsepower might look something like Figure 1. The gray points show designs rendered infeasible as specified by the brush controls. The red points are feasible solutions within the specified ranges. Points marked with a black + symbol show the Pareto front formulated from the preference controls. These are the solutions that mutually maximize horsepower and highway efficiency and lie within the feasible objective space.

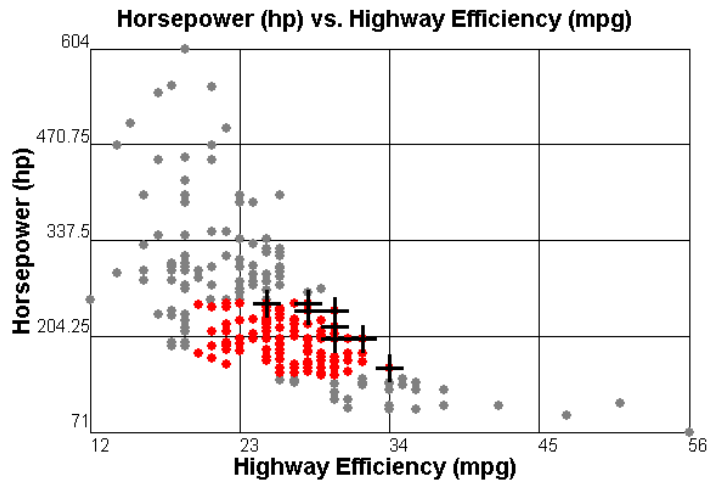


Figure 1: Brushed Car Dataset Showing Infeasible Designs (Gray), Feasible Designs (Red), and the Pareto Solutions (Marked by +).

ORBITAL PROBLEM SPECIFICATION

In order to demonstrate the ATSV as a visualization tool that aids in multi-objective decision making to the orbital mechanics community, a sample orbital transfer problem has been modeled in MATLAB. The problem models an impulsively delivered two-burn plane change spacecraft maneuver from an initial circular orbit to a higher altitude circular orbit. The impulsive changes in velocities (ΔV 's) are constrained to be delivered at the nodal crossings (see Figure 2 points A and B) to prevent alteration of the right ascension of the ascending node. A graphical illustration of this two-burn transfer is shown in Figure 2.

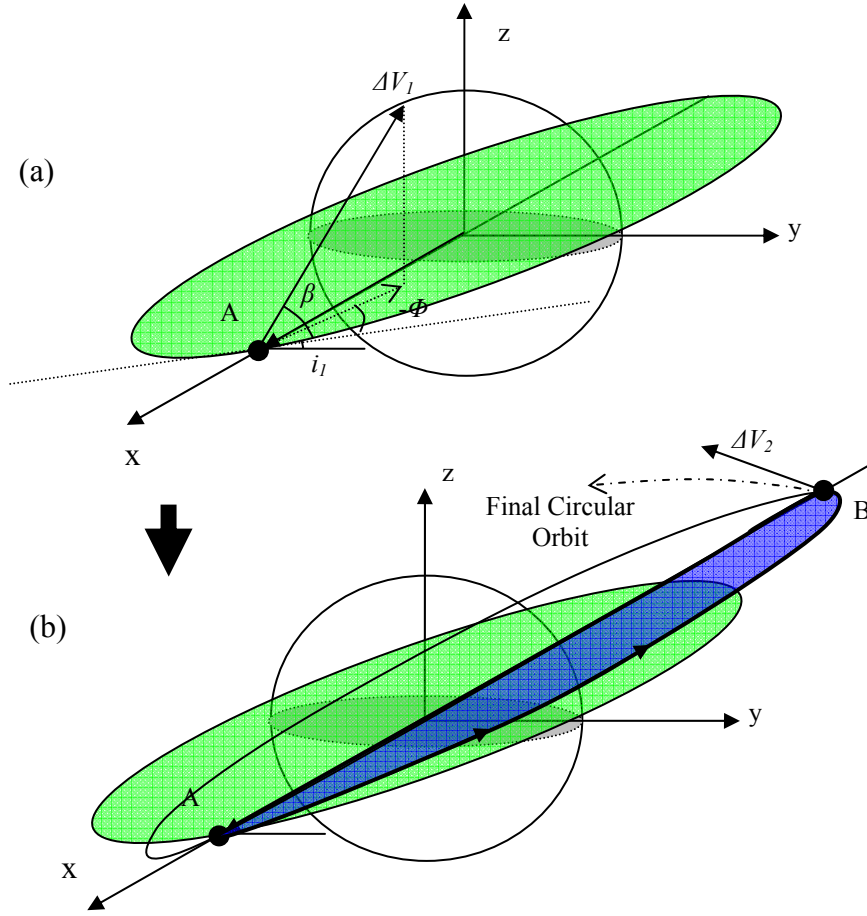


Figure 2: Graphical Illustration of Two-Burn Transfer

The green orbit is the initial circular orbit at inclination i_l shown in Figure 2(a) and (b), the blue orbit is the arc traced by the transfer ellipse at inclination i_t , and the dotted line represents the direction of the final circular orbit at inclination i_2 . ΔV_1 is the impulsively-delivered velocity change at the first nodal crossing, and it is delivered at a flight path angle of Φ with respect to the orbit tangential direction, and an out-of-plane angle β as shown in Figure 3 and Figure 2(a), where x' - y' is the plane of the initial circular orbit. The problem inputs and outputs with corresponding ranges and preferences are summarized in Table 2.

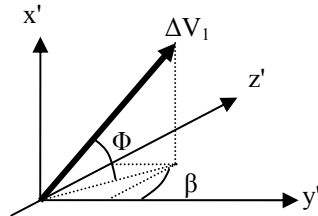


Figure 3: Illustration of Angles Φ and β .

Table 2: Orbital Problem Inputs/Outputs and Ranges/Preferences

Inputs	Range
i_1	Constant
i_2	Constant
ΔV_1	$\Delta V_1 > 0$
Out-of-plane angle, β	$-90^\circ < \beta < 90^\circ$
Flight path angle, Φ	$-45^\circ < \Phi < 45^\circ$
Outputs	Preference
ΔV_{Total}	Minimize
Time of Flight	Minimize
Final Orbital Radius r_2	Maximize
Practical	Discrete Boolean

Note: Only outputs of critical design criteria are listed in this table. All relevant variables generated in the model are included in the dataset.

The MATLAB code algorithm developed for this example is shown as a flow chart in Figure 4. First, the inclination (i_1) and altitude of the initial circular orbit are specified. Then, the essential orbital elements are calculated for this initial orbit. Next, the code randomly varies ΔV_1 , Φ , and β between the ranges specified in Table 2, and the resulting characteristics regarding the transfer and final orbits are obtained. This information, including objective functions ΔV_{Total} , time of flight, and final orbital radius, is calculated by solving Lambert's problem using Lagrange coefficients [10].

Each orbit trio is evaluated for practicality based on the rotation of the orbit planes between the initial, transfer, and final orbits. If the transfer orbit has an inclination that is not between the inclination of the initial and final orbits, the solution is tagged as impractical. Though these solutions are feasible, they represent all orbit trios aligned with a "one step backward, two steps forward" approach. Once the program has looped through all population members, all objective functions of interest are written to a comma-delimited (.csv) excel file.

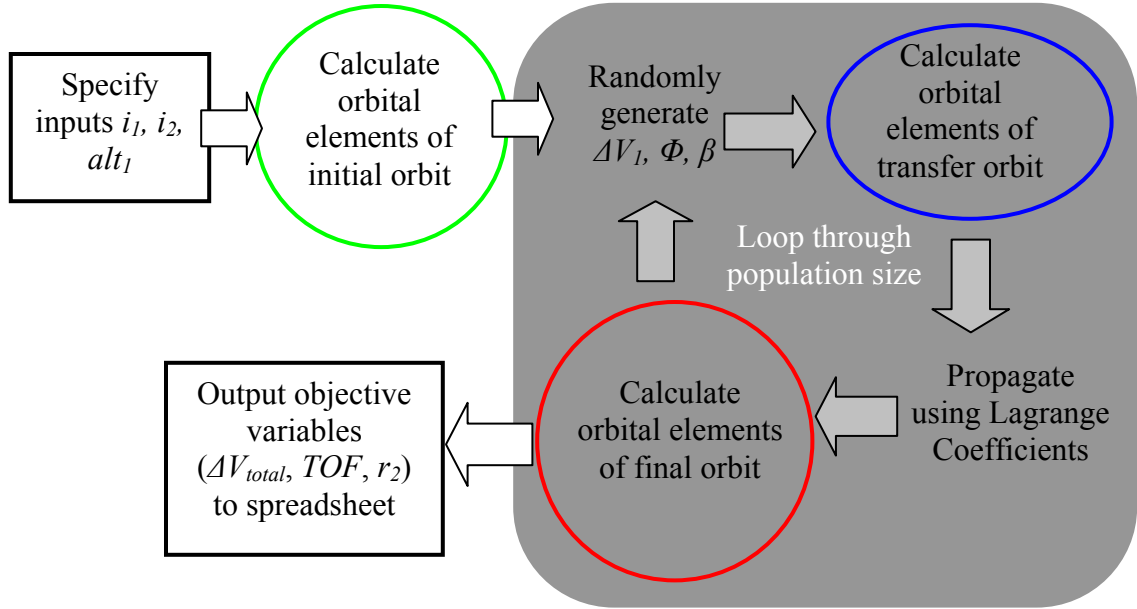


Figure 4: Flowchart Depicting the Algorithm of the MATLAB Code.

When visual steering is implemented by integrating this MATLAB code with ATSV via a Java wrapper program, the code is slightly modified such that one population member is evaluated, and then immediately exported to ATSV. This allows real-time visualization of possible solutions in the objective space as they are generated.

Common initial orbital parameters were chosen for the solutions generated in this work; they are listed in Table 3. The initial inclination i_1 , initial circular altitude, and final inclination i_2 , were chosen to represent an orbital transfer from 28.5°, 500 km low Earth orbit to equatorial geosynchronous.

Table 3: Initial and Desired Final Orbit Characteristics

Initial altitude	LEO 500 km
Initial inclination i_1	28.5°
Final altitude	GEO 35,786 km
Final inclination i_2	0°

RESULTS AND DISCUSSION

Static Dataset

Using the aforementioned MATLAB code, a static dataset consisting of 10,000 members is generated for the orbital transfer problem specified in Table 3. Plotting the three objective functions of interest (ΔV_{Total} , time of flight, and final orbital radius) on

each of the spatial axes creates a three-dimensional envelope, as shown in Figure 5. Figure 5(b) shows an iso-surface at a final orbital radius corresponding to the final desired altitude of 35,786 km. This plane shows the region of interest to be further explored in the ATSV.

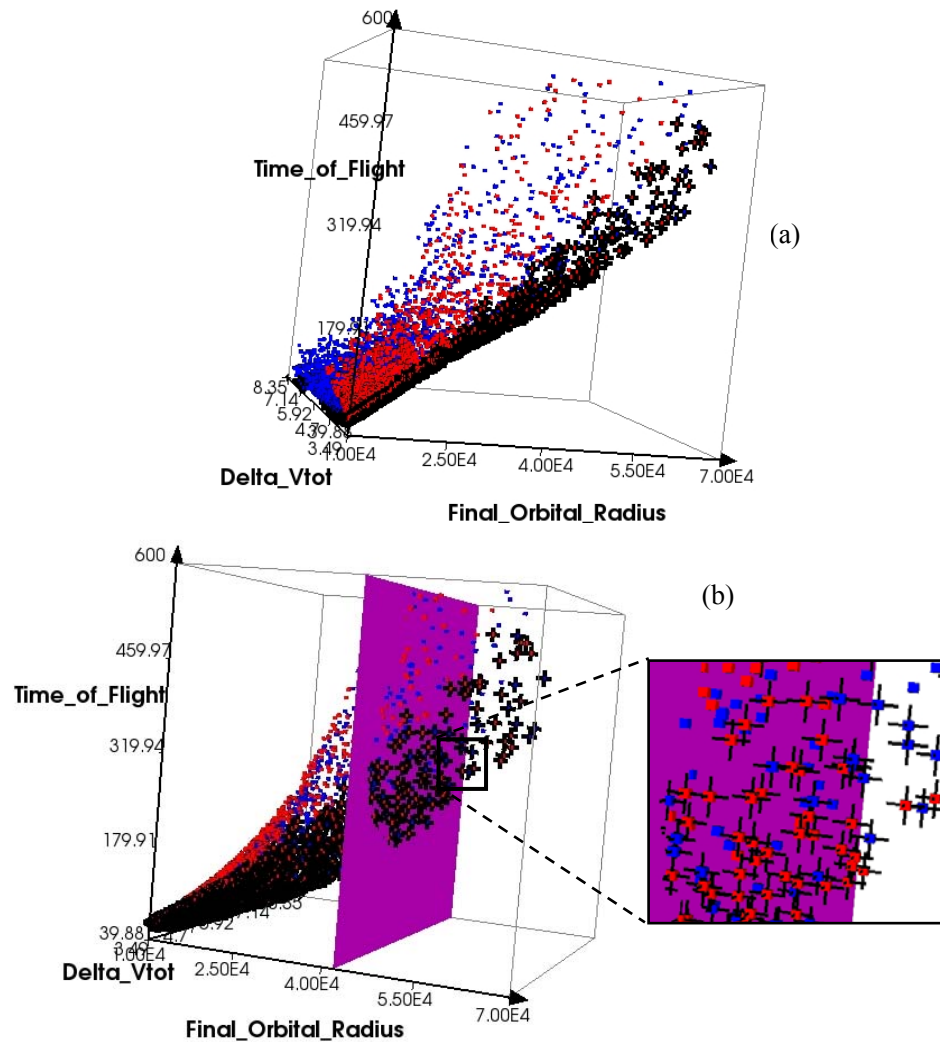


Figure 5: (a) Static Dataset Consisting of 10,000 Members. (b) Isosurface Showing Plane of Interest Near GEO Altitude and Pareto Solutions

The red and blue points are solutions that are practical and impractical, respectively, based on the aforementioned criteria. The solutions marked with a black + symbol represent the Pareto solutions as specified by the preferences shown in the Preference/Brush Control Window in Figure 6. These settings aim to highlight solutions that simultaneously minimize ΔV_{Total} and time of flight while maximizing the radius of the final circular orbit.

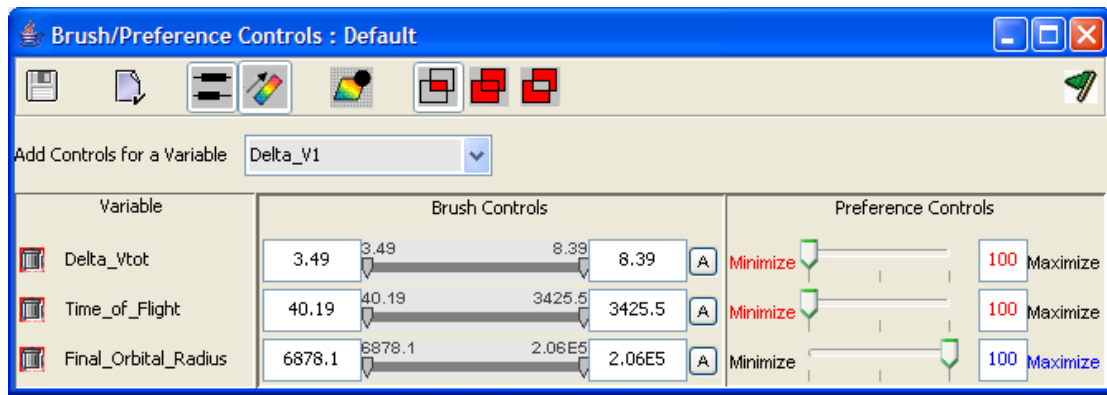


Figure 6: Brush/Preference Controls Window for Orbital Problem Solution Set

Data Exploration and Reduction

The static dataset shown in Figure 5 has been brushed to exclude all points that have a final orbital radius outside the region of interest, nominally specified to be between a final orbital radius of 40,000 km and 45,000 km. The resulting dataset is shown in Figure 7(a).

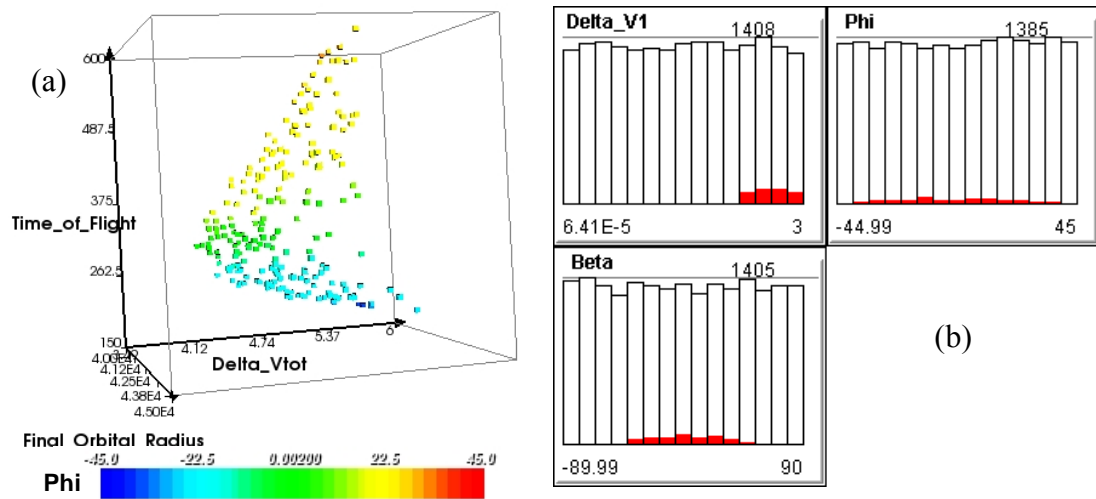


Figure 7: (a) Static Dataset Brushed to Final Orbital Radius Between 40,000 and 45,000 km. (b) Histogram plot of Inputs Brushed in the Same Manner

Figure 7(b) shows the problem input distribution for the first impulsive maneuver. The white bars show the total distribution of each input (approximately uniform), and the red bars show the distribution of inputs for the brushed dataset. Essentially, any bins without red represent inputs that create solutions far away from the area of interest. Proceeding logically, the range of inputs should be reduced by the user to increase the likelihood of generating solutions within the region of interest. From visual inspection of the histograms, reducing inputs to larger values of ΔV_I would be most effective. Also, as

shown in the color contour in Figure 7(a), non-positive values of Φ create solutions along the Pareto Front which minimizes ΔV_{Total} and flight time.

By changing the random distribution of ΔV_I from $[0, 3.0]$ km/s to $[2.0, 3.0]$ km/s and Φ from $[-45^\circ, 45^\circ]$ to $[-45^\circ, 0]$ in the model, a population of 10,000 solutions generates approximately 500 new points in the region of interest. By generating static datasets, then brushing to the region of interest, and exporting visible designs, the solutions in the region of interest are stored in a separate file. Just a few iterations of this technique result in the concatenation of all these region specific points shown in Figure 8(a) and (b).

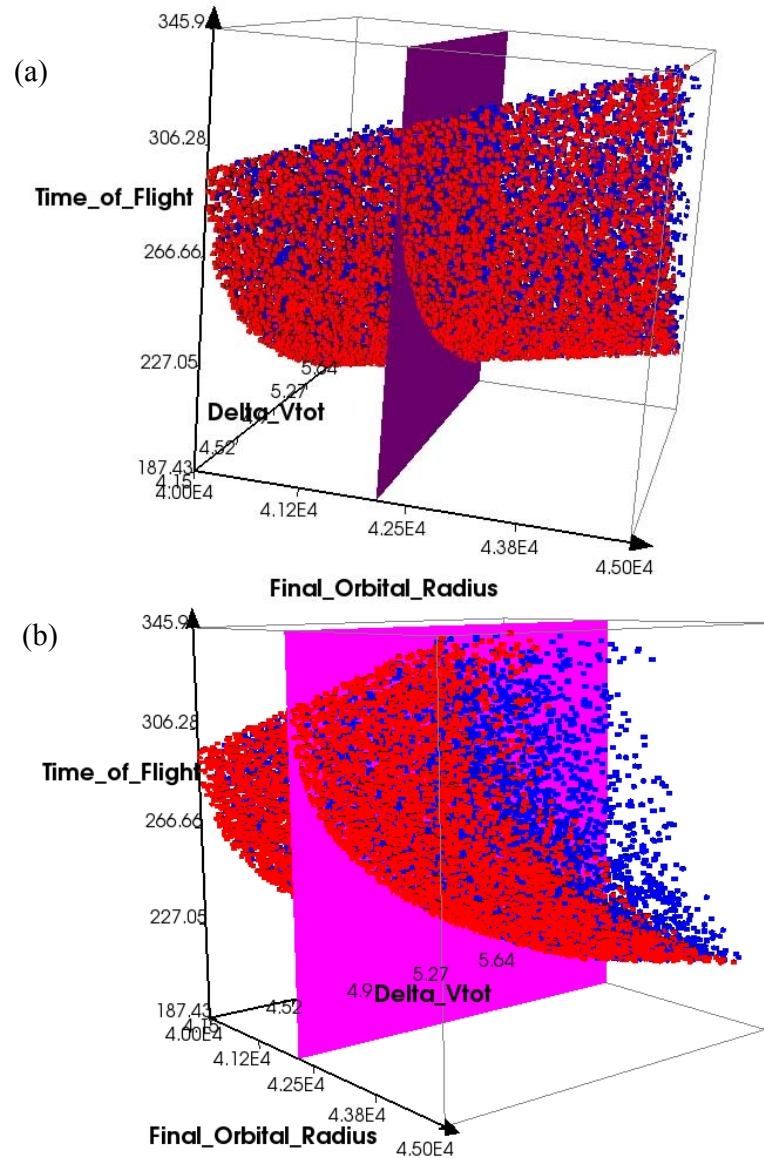


Figure 8: (a) and (b) Two Perspectives of Densely Populated Region of Interest with Iso-surface at GEO Altitude

Observable Trends

After the region of interest has been more densely populated, it is easier to more precisely recognize trends in the dataset. As shown in Figure 8, the three-dimensional envelope formed by the practical (red) points seems to be representable by a continuous Pareto surface. Based on the physics in the model, this apparent continuity is an expected result and leads to the conclusion that this problem is well suited for static datasets and may not require the use of attractors. In other words, the nature of the problem results in easily acquired Pareto solutions, and no “gaps” exist near the Pareto front that must be filled.

Using the ATSV’s Scatter Matrix Plot, one can easily view trends between variables of interest. This scatter matrix, brushed to ± 50 km from the target GEO altitude, is shown in Figure 9. Once again, gray points represent “brushed out” points, and red and blue points are the practical and non-practical solutions respectively which fall within this criterion. Because of this ± 50 km brushing, the trends that follow can be considered specific to the LEO to GEO problem parameters listed in Table 3. Though not all points fall precisely on the plane formed by the geostationary altitude, these solutions are considered in the neighborhood and would undoubtedly be subject to orbital fine tuning and phasing, but this is beyond the scope of this demonstration.

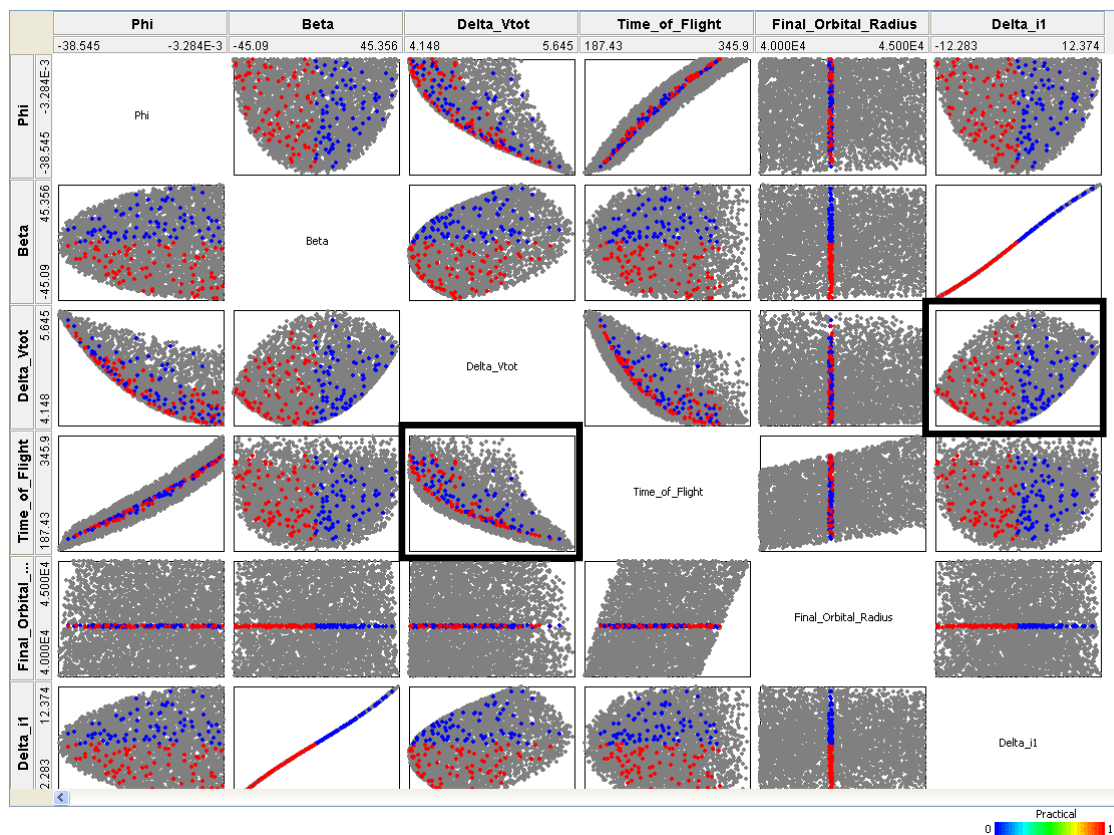


Figure 9: Scatter Matrix Plot of Variables of Interest

The two highlighted scatter plots, time of flight vs. ΔV_{Total} , and ΔV_{Total} vs. Δi_I are of particular interest. These plots are enlarged and shown in Figure 10 and Figure 11, respectively. The trend of the Pareto front in Figure 10 shows an inverse relationship between time of flight and ΔV_{Total} . Furthermore, it resembles the expected solution for a two-dimensional version of this problem with no plane change (from Lambert's problem) [10]. Intuitively, the trend's shape remains similar for different desired final orbit sizes.

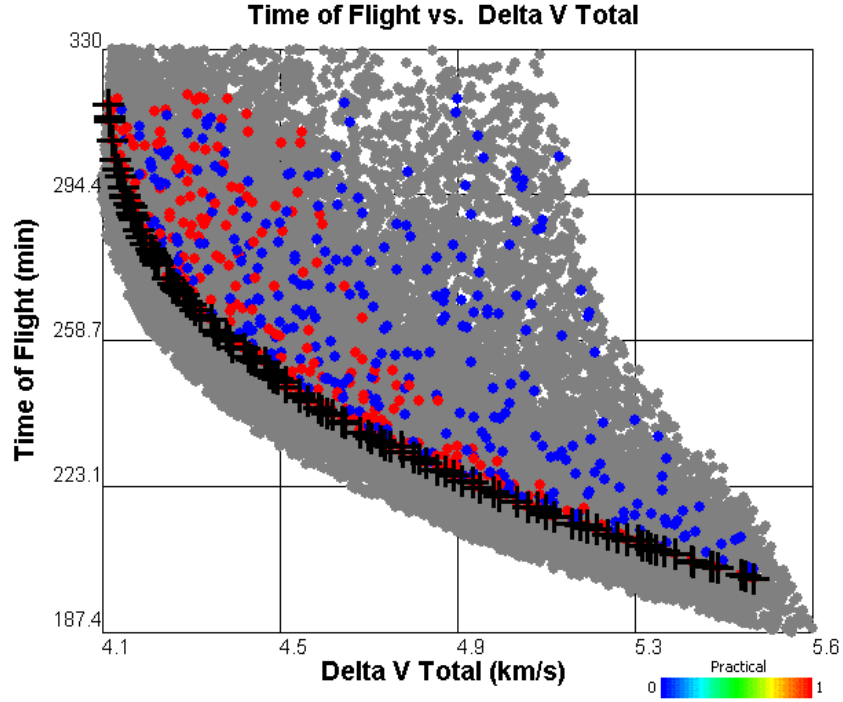


Figure 10: Solution Set and Pareto Frontier for Time of Flight vs. Total Impulsive Velocity Change (± 50 km Brush)

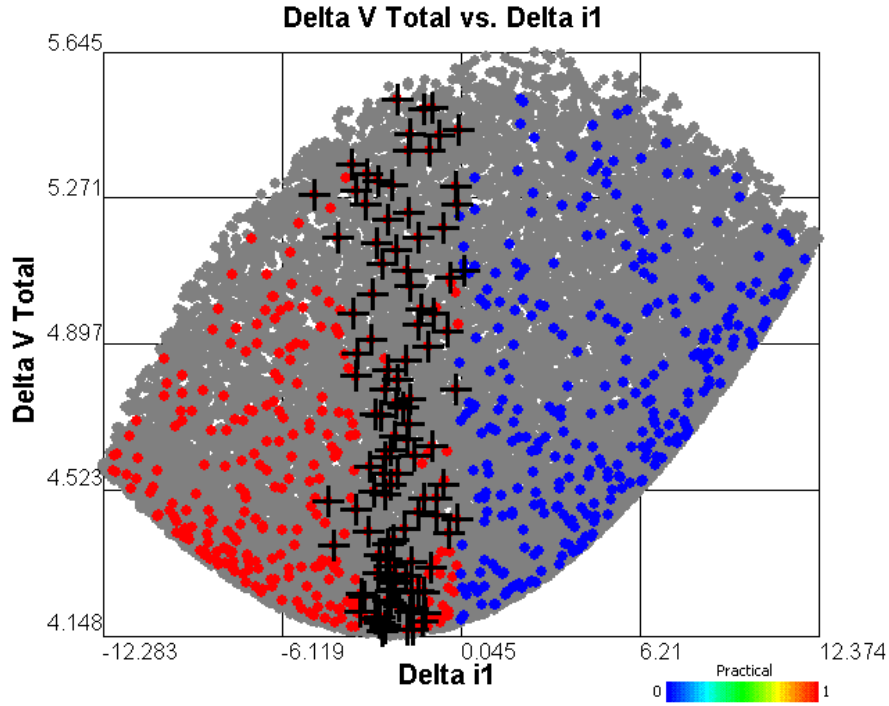


Figure 11: Total Impulsive Velocity Change vs. Inclination Change from First Burn (± 50 km Brush)

Figure 11 shows ΔV_{Total} vs. the change in inclination after the first impulsive maneuver (Δi_1). Because of the nature of this specific plane change as well as the manner in which angle β is defined, solutions must have a negative Δi_1 value to be considered practical. As orbital dynamics theory suggests, the optimal (Pareto) plane change transfers shown here are solutions which impulsively deliver the most inclination change during the second burn (at a lower orbital velocity). Consequently, the optimal solutions have small Δi_1 's, shown in Figure 11 to be within a range of 0° to 5° .

The Pareto design that most effectively minimizes ΔV_{Total} has design attributes listed in Table 4.

Table 4: Attributes of Solution with Minimum ΔV_{Total}

ΔV_I	2.403 km/s
Out-of-plane angle, β	-11.05°
Flight path angle, Φ	-0.3776°
ΔV_{Total}	4.1625 km/s
Time of Flight	316.41 min
Final Orbital Altitude	35,740 km
Δi_1	-2.645°

Comparison with Analytical Optimal Three-Dimensional Orbit Change

Vallado [10] explored a similar three-dimensional two-burn plane change orbital mechanics problem. The Δi_1 and Δi_2 which lead to the lowest ΔV_{Total} can be solved for iteratively in what Vallado labels the “minimum combined plane change” algorithm:

$$\sin(s\Delta i) = \frac{\Delta V_1 V_{final} V_{transfer,b} \sin((1-s)\Delta i)}{\Delta V_2 V_{initial} V_{transfer,a}} \quad (1)$$

where $V_{initial}$ and V_{final} are initial and final circular orbital speeds, $V_{transfer,a}$ and $V_{transfer,b}$ are speeds at points A and B as shown in Figure 2, and s is a scaling factor that represents the percentage of the total inclination change which is delivered on the first burn:

$$\Delta i_1 = s\Delta i \text{ and } \Delta i_2 = (1-s)\Delta i \quad (2)$$

Solving the equation iteratively yields an optimal s , which in turn creates the optimal distribution of plane change between the initial and final maneuvers. For the initial and final orbit characteristics explored in this paper (see Table 3) this algorithm yields the optimal transfer parameters listed in Table 5.

Table 5: Comparison between ATSV Results and Minimum Combined Plane Change Algorithm

Parameter	ATSV	Algorithm	% Difference
ΔV_{Total}	4.1625 km/s	4.1620 km/s	0.01%*
<i>Final Orbital Altitude</i>	35,740 km	35,786 km	0.13%
Δi_1	2.645°	2.262°	14.5%
Δi_2	25.855°	26.238°	-
* Negligible within uncertainty			

As shown in the last column of Table 5, the percent difference between the ATSV results and the minimum combined plane change algorithm results is small. The total impulsive velocity change is the same within the uncertainty of the algorithm calculation. The algorithm yields an optimal initial plane change of 2.645° which is 14.5% higher than what the ATSV found. However, since the ATSV did not locate a solution precisely at geostationary altitude, the comparison to the iterative algorithm serves more as a verification of the data and less as an optimality criterion. The ATSV has clear advantages over this algorithm in that it can visualize the entire design space and effectively highlight the optimal solutions with respect to any number of multiple objectives.

SUMMARY AND CONCLUSIONS

The Applied Research Lab Trade Space Visualizer (ATSV) is an effective software tool that can be applied to a range of complex systems design problems, including orbit transfer problems. Using static exploration and data reduction to explore the trade space in a sample two-burn plane change problem permits the user to observe relevant trends regarding design objectives of interest, eventually allowing the designer to choose an optimal solution from the objective space. As applied to the impulsive two-burn orbital mechanics problem in this paper, the ATSV's trade space visualization made it easy to narrow down design inputs to generate more solutions in the region of interest. Using preference controls, Pareto optimal solutions were obtained for a 28.5°, 500 km LEO to GEO impulsive maneuver with respect to total impulsive velocity change and time of flight.

In this work, the trade space exploration capabilities of the ATSV have been illustrated for a simple orbit transfer problem. This software can be used for much more complex orbital mechanics problems involving both continuous and non-continuous solutions. Future applications of the ATSV's trade space visualization capabilities include exploration of low-thrust orbital maneuvers or satellite constellation design optimization. Furthermore, attractor and/or Pareto based sampling could be applied to a somewhat less predictable orbital mechanics problem to provide a means for an effective demonstration of ATSV-model integration.

The Applied Research Lab Trade Space Visualizer is available for use in any technical field and can be acquired for free for evaluation purposes at the ATSV website: <http://www.tradespaceexploration.psu.edu/>

ACKNOWLEDGMENTS

This work is being supported by the National Science Foundation under Grant No. CMMI-0620948. Any opinions, findings, and conclusions or recommendations presented in this paper are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

1. Balling, R., "Design by Shopping: A New Paradigm?", Proceedings of the Third World Congress of Structural and Multidisciplinary Optimization (WCSMO-3), Buffalo, NY, University at Buffalo, pp. 295-297, 1999.
2. Wilson, T.D., and J.W. Schooler, "Thinking Too Much: Introspection can Reduce the Quality of Preferences and Decisions", *Journal of Personality and Social Psychology*, vol. 60(2), pp. 181-192, 1991.
3. Stump, G., S. Lego, M. Yukish, T.W. Simpson and J.A. Donndelinger, "Visual Steering Commands for Trade Space Exploration: User-Guided Sampling with

- Example”, ASME Design Engineering Technical Conferences - Design Automation Conference, Las Vegas, NV, ASME, DETC2007/DAC-34684, 2007.
4. Messac, A., and X. Chen, “Visualizing the Optimization Process in Real-Time Using Physical Programming,” *Engineering Optimization*, 32(6), 721-747, 2000.
 5. Ng, W.Y., “Generalized Computer-Aided Design System: A Multiobjective Approach”, *Computer-Aided Design*, 23(8), 548-553, 1991.
 6. Eddy, W.F., and A. Mockus, “Dynamic Visualization in Modeling and Optimization of Ill-Defined Problems: Case Studies and Generalizations”, Technical Report, Department of Statistics, Carnegie Mellon University, Pittsburgh, PA. 1995.
 7. “Applied Research Laboratory Trade Space Visualization”, <http://www.atsv.psu.edu>, date cited 1/22/08.
 8. Price, R. Storn and J. Lampinen, *Differential Evolution - A Practical Approach to Global Optimization*, Berlin, Springer, 2005.
 9. Becker, R.A., and W. S. Cleveland, "Brushing Scatterplots," *Technometrics*, vol. 29(1), pp. 127-142, 1987
 10. Vallado, V.A., *Fundamentals of Astrodynamics and Applications*, , Microcosm Press, El Segundo, CA, p. 445-468, 2001.