The Pennsylvania State University

The Graduate School

College of Engineering

MULTI-CRITERIA DESIGN OPTIMIZATION USING MULTI-DIMENSIONAL DATA VISUALIZATION

A Thesis in

Mechanical Engineering

by

Gary M. Stump

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

May 2004

Gary M. Stump

We approve the thesis of Gary M. Stump.

Timothy W. Simpson
Associate Professor of Mechanical and Industrial Engineering
Thesis Advisor

Mary Frecker
Associate Professor of Mechanical Engineering

Mark T. Traband
Head of the Manufacturing Systems Division
Applied Research Laboratory

Richard C. Benson
Professor/Department Head of Mechanical Engineering

ABSTRACT

Two graphical user interfaces, the ATSV (Advanced Trade Space Visualizer) and an I-beam graphical design environment, have been developed to explore trade spaces of multi-criteria problems. The ATSV is a tool used to explore trade spaces in the preliminary design stage of a project, where there exist great potential cost benefits. Conceptual models, based on open literature and domain expertise, are used to rapidly populate trade spaces. Such trade spaces consist of a large datasets, and the ATSV allows a decision-maker to visualize this data using multi-dimensional visualization techniques. Optimization within the trade space occurs when decision-makers form a preference *a posteriori* and use this preference structure to select a preferred design. Along with using multi-dimensional visualization techniques, the ATSV can utilize virtual reality hardware, which includes stereoscopic projection on desktop monitors and projection screens. The following functionality was incorporated into the ATSV:

1. Visualize complex datasets using multi-dimensional visualization techniques
2. Assign variables to glyph, histogram, and parallel coordinates plots
3. Specify upper and lower bounds of an n-dimensional design space
4. Implement dynamic brushing within glyph, parallel coordinates, and histogram plots to uncover relationships in the dataset (linked views)
5. Visualize different regions of interest, using preference shading and corresponding Pareto frontier identification
6. Create multiple views of glyph, histogram, and parallel coordinates plots of the same trade space
7. Select a design from the glyph plot to display quantitative information, 3D geometries, and other files such as images and documents
8. Use advanced visualization hardware to view graphs and 3D geometries in stereo mode

Also, a study of an I-beam design problem was performed to assess the effectiveness, efficiency, and satisfaction of graphical user interfaces. This Human-

Computer Interaction (HCI) experiment, studied the effect that time delays have on the performance of finding an optimal design within a 2D trade space. The ATSV extends this work by both visualizing multi-dimensional data and allowing a decision-maker to form a preference after viewing the trade space.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

The author gratefully acknowledges support from the National Science Foundation under NSF Grant No. DMI-0084918 and the Office of Naval Research under Contract No N00014-00-G-0058.

Chapter 1

Motivation

The first chapter discusses the motivation for developing the Advanced Trade Space Visualizer (ATSV) that allows a decision-maker to implement a Design by Shopping paradigm. First, multi-criteria decision-making problems can be classified into different categories, each characterized by when the preference structure is specified by the decision-makers in the optimization problem. Then, the Design by Shopping paradigm is introduced, discussing its key concepts within the decision-making process. Simulation-Based Design (SBD) is used to support this paradigm by rapidly populating trade spaces using discrete sampling techniques. The ATSV, using multi-dimensional visualization techniques and advanced visualization hardware, visualizes the resulting trade space. Finally, a summary and overview of the thesis are given.

1.1 Classification of Multi-criteria Decision-Making Problems

Hwang and Masud [1] classify multiple criteria decision making problems into the following four categories.

1. *No Articulation* of Preference: No relative preferences are needed once all the objectives and constraints have been formulated.

2. *A Priori* Articulation of Preference: A relative preference structure is given before the optimization process.
3. *A Progressive* Articulation of Preference: Preference structure is specified during the optimization process, and the decision-maker gives tradeoff information during iterations of the optimization process.
4. *A Posteriori* Articulation of Preference: No preference structure is specified until after a rich set of non-dominated designs is found.

A difficulty in using an *a priori* articulation of preference is deciding on a weighting vector between performance variables [1]. Different relative preferences in multi-criteria trade spaces can lead to different optimal solutions. A preference structure specified *a priori* will find to a Pareto optimal solution but may not lead to the most preferred solution.

An advantage of an *a posteriori* articulation of preference is that no preference structure is needed until a variety of designs are presented to the decision-maker. Additionally, Pareto optimal designs can be calculated using the preference structure specified by the decision-maker. As a result, the decision-maker formulates tradeoff information after viewing the Pareto frontier [1]. Within this set of optimal designs lies the most preferred solution.

1.2 Design by Shopping Paradigm

Balling proposes the Design by Shopping paradigm [2], where a decision-maker first explores a set of non-dominated designs and then chooses a preferred solution. The three steps of this paradigm are as follows:

1. Allow a decision-maker to visualize a variety of designs
2. Then *form* a preference structure
3. Select the most preferred design based on the formed preference.

This paradigm introduces more control to the decision-makers compared to a traditional optimization problem by allowing them to first form their preference based on data visualization, and then choose a preferred design based on their preference. The incorporation of a Design by Shopping paradigm can be classified as an *a posteriori* articulation of preference to solve a multi-objective optimization [1]. This approach is more attractive to decision-makers than an *a priori* approach, because design exploration is followed by a selection process in which the decision-makers have control [2].

The motivation of using Balling's Design by Shopping paradigm lies in the ability to rapidly generate feasible designs. Balling states the need to visualize a set of non-dominated designs; however, Pareto optimal designs only visualize a small subset of feasible solutions. One can extend this paradigm by visualizing the entire trade space along with the Pareto frontier, thereby allowing a decision-maker to visualize an extensive mapping of the feasible design space. This is accomplished through simulation-based design as described in the next section.

1.3 Simulation Based Design

Conceptual models are used to rapidly populate trade spaces, which can consist of numerical information, virtual prototypes, and images of design alternatives. Such models are based on design rules captured from textbooks and domain experts. The trade space is mapped by discretely sampling points in the input space, and a conceptual model calculates designs based on the initial sampled points. This process is repeated until an extensive mapping of the trade space is generated, as shown in Figure 1-1.

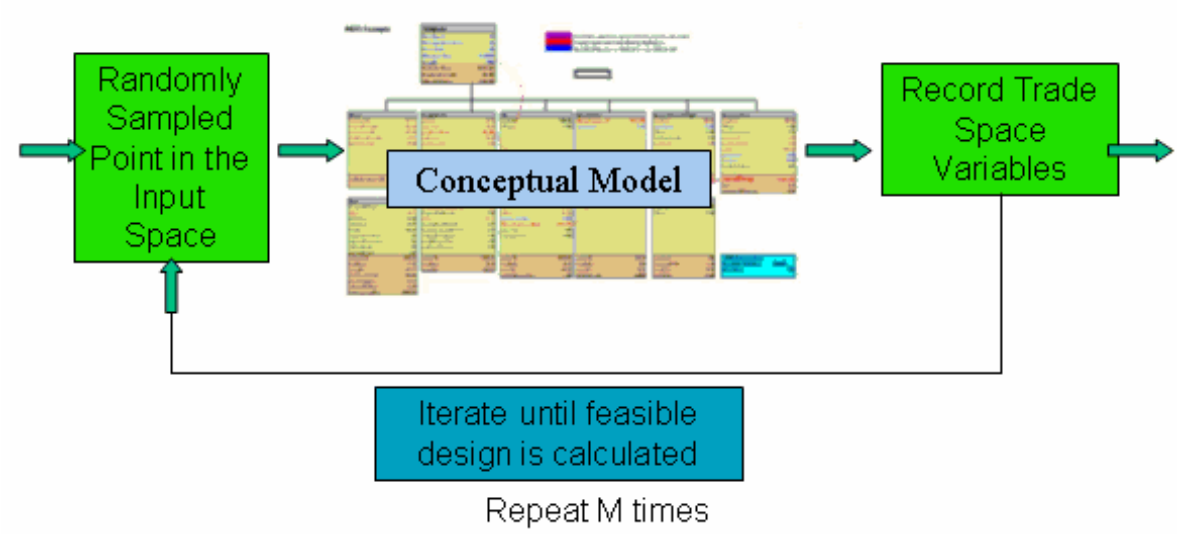


Figure 1-1: Design Space Mapping

Within the conceptual models, a decision-maker can select variables of interest, which are recorded and saved to a dataset. An example dataset is displayed in Figure 1-2 where each row represents a design and each column represents a variable. Additional data to generate virtual prototypes and images can be recorded into the data file; COTS

(Commercial off the Shelf) programs, such as a CAD programs, could then use this parametric information to generate virtual prototypes.

	A	B	C	D	E	F	G
1	RxnWheelIdx_Input	pellantType_In	Delta_V_OI_Input	Slew_Time	Total_Mass	Cost	Mission_Life_Input
2	6	N2O4/MMH	1337.89	100.11	2184.3	3619.74	2.45
3	6	N2O4/N2H4	1431.08	101.08	1901.37	2805.44	4.42
4	5	N2O4/N2H4	1231.7	115.04	1580.84	2203.26	3.84
5	13	N2O4/N2H4	1525.82	52.18	2463.23	4427.46	4.35
6	8	N2O4/N2H4	1308.18	121.43	1747.02	2502.43	4.84
7	13	N2O4/MMH	1568.74	56.94	3362.19	7239.67	4.28
8	1	N2O4/N2H4	1228.35	148.33	1621.95	2276.06	4.69
9	1	N2O4/MMH	1258.26	147.08	1960.26	3121.14	2.97
10	5	N2O4/N2H4	1204.53	123.27	1647.65	2285.03	4.27
11	12	N2O4/MMH	1107.46	33.1	2188.34	3411.6	4.23
12	12	N2O4/MMH	1522.56	33.54	2725.84	5401.61	2.82
13	7	N2O4/MMH	1300.09	109.24	2144.27	3526.75	2.67
14	3	N2O4/N2H4	1216.17	96.47	1573.69	2152.97	2.47
15	11	N2O4/N2H4	1253.53	107.68	1718.3	2416.08	4.48
16	3	N2O4/MMH	1282.06	105.02	2110.45	3445.5	2.45
17	10	N2O4/N2H4	1409.64	71.65	1812.05	2629.02	2.99
18	1	N2O4/N2H4	1430.07	143.53	1695.74	2483.71	3.02
19	3	N2O4/N2H4	1377.19	103.83	1808.94	2622.68	4.03
20	14	N2O4/N2H4	1135.46	104.59	2195.38	2860.59	3.93
21	1	N2O4/MMH	1193.93	157.96	2046.08	3275.72	4.64
22	13	N2O4/N2H4	1261.49	48.54	2063.7	2860.97	4.77
23	9	N2O4/N2H4	1371.16	172.21	1734.28	2495.78	3.14
24	13	N2O4/N2H4	1542.57	52.33	2498.76	4608.05	4.68
25	6	N2O4/MMH	1437.13	102.38	2359.83	4147.46	3.18
26	9	N2O4/N2H4	1532.99	176.21	1865.81	2779.12	2.04

Figure 1-2: Example Dataset

The conceptual models used to generate trade spaces have a low-level of detail, thereby presenting a decision-maker with trade studies and not detailed designs. Figure 1-3 displays the high value of making a good decision early in the design stage [3]. The Advanced Trade Space Visualizer (ATSV) was developed for use in the preliminary design stage of a project, analyzing different configurations, performance metrics, materials, cost, and constraints of a design. The ATSV is intended to guide the decision-maker to a preferred design; this process would be followed by further detailed conceptual models of the preferred design or nearby designs.

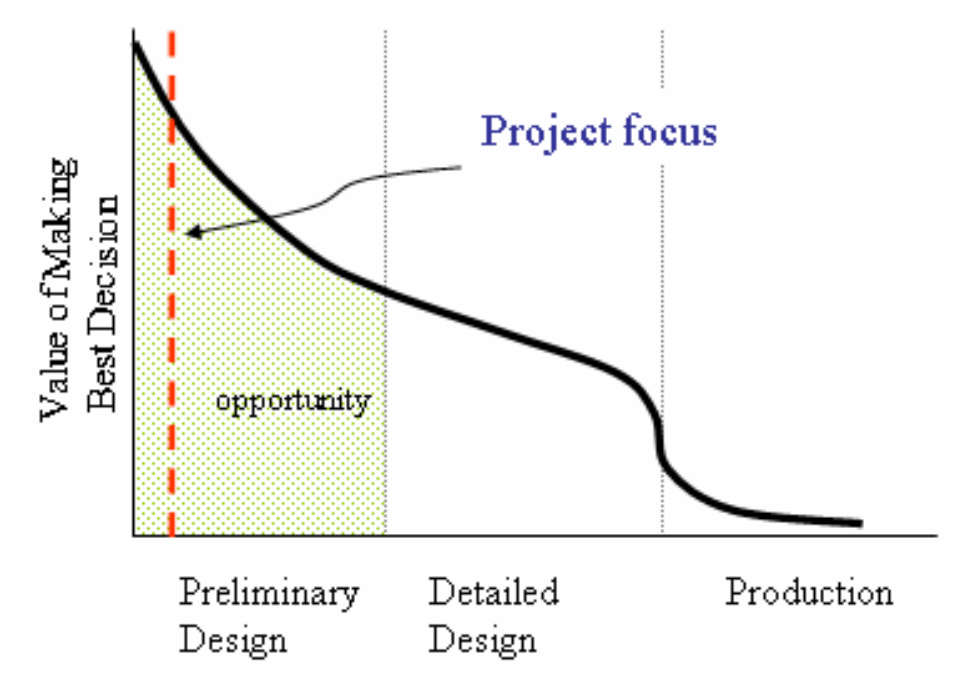


Figure 1-3: Preliminary Design Stage Value [3]

1.4 Data Visualization

Balling [2] states the need for research in the following two areas :

1. interactive graphical computer tools to assist the decision-makers in the shopping process, and
2. efficient methods for obtaining Pareto frontiers.

The Advanced Trade Space Visualizer (ATSV) addresses the first area of research by visualizing datasets using multi-dimensional visualization techniques, quantitative information windows, 3D geometry display, and other files associated to

individual designs. The ATSV allows decision-makers to experiment with different preference structures, and the resulting Pareto frontier is displayed, as they "shop" for the most preferred design. Additionally, the ATSV uses advanced visualization environments that display glyph plots and 3D geometries in stereo mode. The second area of research is addressed by finding and displaying Pareto optimal designs in a discretely sampled trade space.

1.5 Thesis Roadmap

The ATSV allows decision-makers to implement a Design by Shopping paradigm, which can by classified as an a *posteriori* articulation of preference. This paradigm can be implemented using Simulation-Based Design, which is used to rapidly populate the feasible design space. These sampled points are recorded into data files and visualized within the ATSV, using multi-dimensional visualization techniques and virtual reality.

Chapter 2 discusses previously developed methods for visualizing multi-dimensional datasets, along with general guidelines for graphical user interface development. The chapter concludes with a description of virtual reality environments. Chapter 3 discusses an I-beam design problem that analyzes the effectiveness and efficiency of using graphical user interfaces. This interface analyzed an I-Beam design problem with two variables and a known *a priori* preference structure. The ATSV extends this work in the multi-dimensional case by analyzing datasets using an *a posteriori* articulation of preference. Chapter 4 presents the ATSV and demonstrates its

use through (1) an extended version of the I-Beam problem, and (2) a satellite design example. Chapter 5 summarizes contributions from the research and future work.

Chapter 2

Background Research

This chapter addresses graphical user interface (GUI) development, multi-dimensional visualization techniques, and advanced visualization environments. Principles from the field of Human-Computer Interaction (HCI) can be used as a guide to develop user-friendly, efficient, and effective software. An overview of previously developed methods to visualize multi-dimensional data is discussed. Additionally, a summary of advanced visualization environments is given.

2.1 Graphical User Interface Development for Optimization

Graphical user interfaces (GUIs) can aid decision-makers in the design process by using data visualization methods. Ng [4] advocates the use of data visualization and interaction to support the designer in making informed decisions and tradeoffs during multi-criteria design and optimization. Jones [5] argues that design optimization is more than just algorithm development; appropriate representations (i.e., visualization strategies) are needed to better understand the models, algorithms, data, and solutions obtained during the design optimization process. Finally, Eddy and Mockus [6] argue that visualization should be considered as a solution tool rather than simply a means to present results.

The field of Human-Computer Interaction (HCI) studies the interaction between people and computer technology. HCI aids in the development of computer technologies that are usable by end-users of graphical user interfaces, hardware, and interaction devices. A graphical user interface should be easy to learn, remember, and use. In addition, the aesthetics of the interface should communicate information effectively without overloading the user with information. Table 2-1 list several principles one can follow to improve usability and layout of a graphical user interface [7,8].

Table 2-1: Graphical User Interface Development

Principles for graphical user interface usability	**Principles for graphical user interface layout**
• Create an intuitive interface • Incorporate consistency in layout, graphic vocabulary, and commands • Always display important functions • Develop a simple navigation system • Allow a user to stop or break the program. • Create an interface that is responsive and fast • Only display information that is relevant to the task at hand • The interface should not exceed a user's limited working memory	• Minimize screen clutter • Use an easily readable text style (size and font) • Use color schemes that do not distract a user's attention • Customize interface for users with disabilities • Minimize search time – information should be as few keyboards or mouse clicks away as possible • Information should organized in groups and ordered to aid the user in finding information • Focus the user's attention on key information by positioning and highlighting items

2.2 Multi-dimensional Visualization Techniques

This section contains an overview of multi-dimensional visualization techniques, including glyph plots, parallel coordinates, scatter matrices, brushing, linked displays, isosurfaces, dimensional stacking, grand tour, projection pursuit, trellis plots, and histograms.

2.2.1 Glyph Plots

Glyphs are graphical icons that display multivariate information using their physical characteristics, such as size, shape, orientation, color, texture, and transparency. According to Kraus, interactive exploration of multivariate data with glyph plots requires new graphical interfaces that allow users to generate many displays of the same design space [9]. Many previous studies [10-12] examine which encoding techniques are the most beneficial; however, this question still remains unanswered [13]. Figure 2-1 displays two glyph plots, where each design in the dataset is represented by an individual glyph. The right glyph plot, in Figure 2-1, can display 7-dimensional information using the position of the glyph cube to represent three dimensions, and the other four dimensions can be represented by the size, color, orientation, and transparency of the glyph cubes. The left glyph plot, in Figure 2-1, displays seven dimensions by using the spatial position of the glyph to represent three dimensions, and the length of the four glyph arms to represent an additional four dimensions.

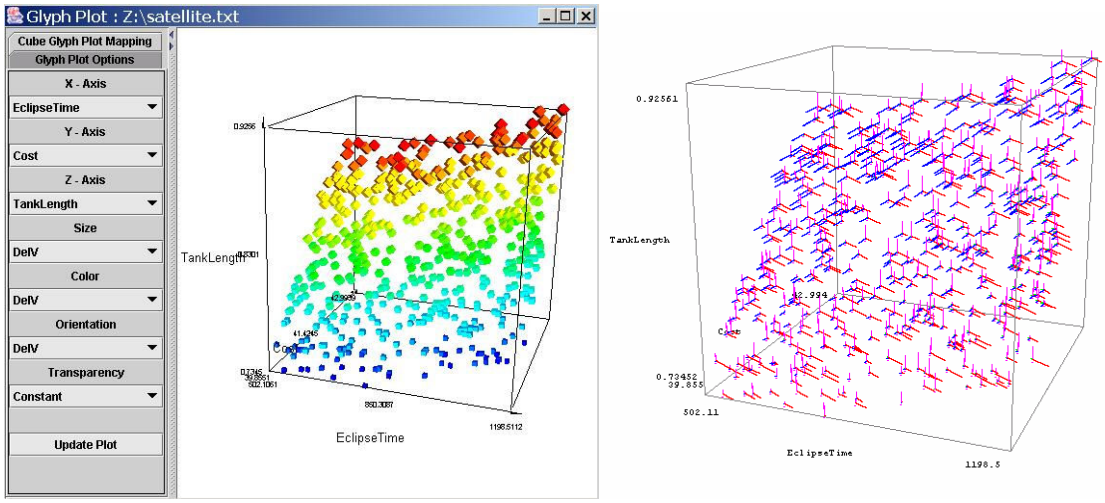


Figure 2-1: Glyph Plots

2.2.2 Parallel Coordinates

Parallel coordinates, proposed by Inselberg [14], displays multivariate designs by using a polyline that intersects equally-spaced axes. Each polyline in the parallel coordinates display represents one design; a polyline's n-intersections with the horizontal axes of the parallel coordinates plot represent the *n* variables of a design. The resulting layout of a parallel coordinates plot allows a decision-maker to visualize all variables of a design. The orientations of multiple polylines reveal qualitative information between adjacent variables in the parallel coordinate display. For example, diverging and intersecting line segments display relationships such as positive and negative trends, respectively. Figure 2-2 displays two parallel coordinate plots, an individual n-dimensional design and a set of n-dimensional designs.

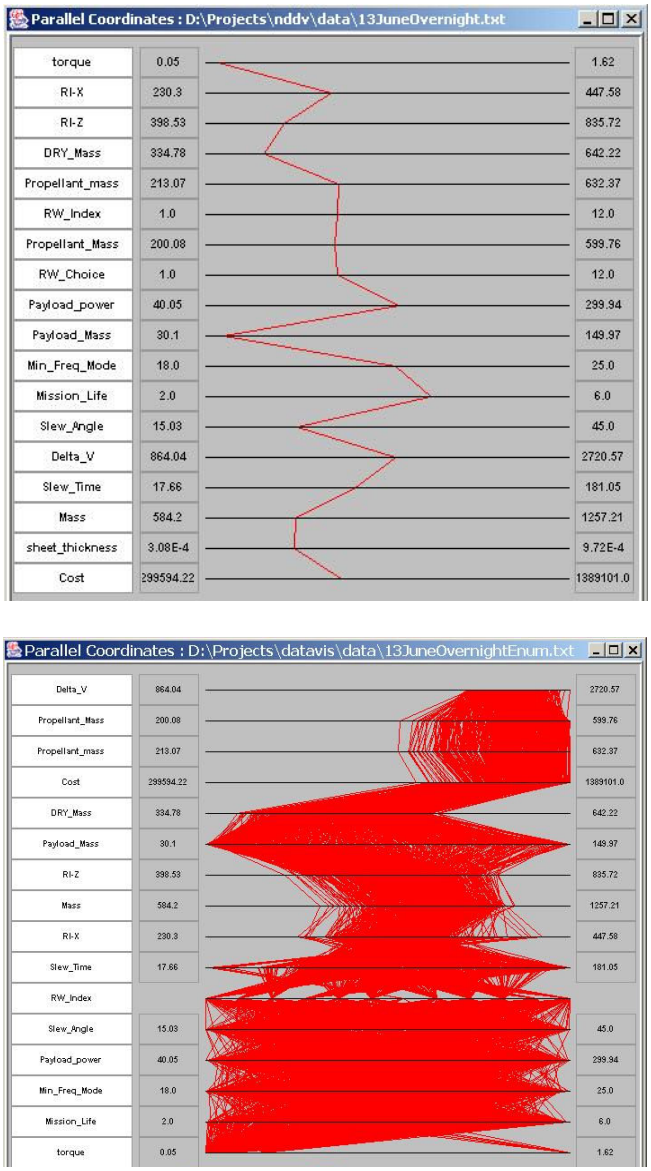
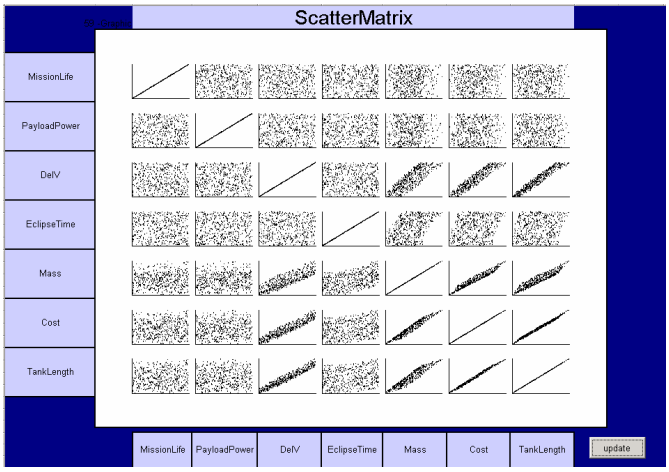


Figure 2-2: Parallel Coordinates Plot

2.2.3 Scatter Matrices and Scatter Cubes

A scatter matrix is a symmetric array of scatter plots, in which all variables are plotted against each other using 2D scatter plots. Each graph in row has the same variable

on the y-axis, and each graph in a column has the same variable on the x-axis. A scatter cube extends a scatter matrix into the third dimension, by creating an array of 3D scatter plots [15, 16]. Figure 2-3 displays a scatter matrix and a scatter cube, respectively.



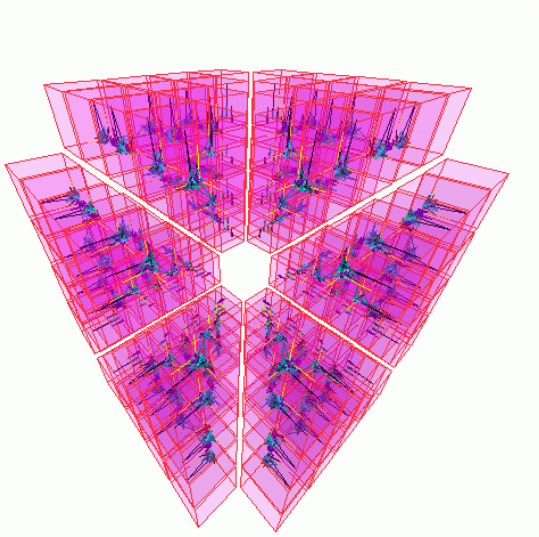


Figure 2-3: Scatter matrix and a Scatter cube [16]

2.2.4 Brushing

A brush is a user-defined region within a multivariate data set in which designs that fall within this region are highlighted, deleted, or masked [17]. A highlighting brush distinguishes a subset of designs from the rest of the dataset. Deleting refers to the removal of all designs that fall within the brush; in contrast, masking only displays designs that fall within the brush. Brushing also allows a decision-maker to set limits within the trade space and visualize correlations between variables. Figure 2-4 illustrates a brush, where the right plot displays the entire data set and the left plot displays a brush with y values between -1 and 0.

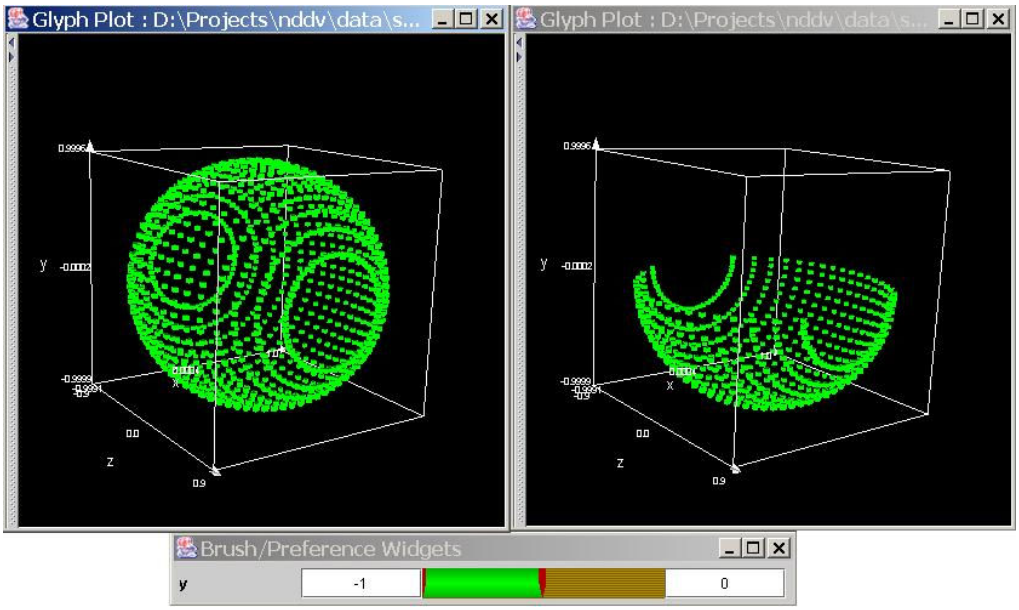


Figure 2-4: Illustration of Brushing

2.2.5 Linked Displays

Linking is the process of displaying information across multiple views of data [18]. Brushing commonly links data visualization plots, allowing a user to select, delete, or mask a subset of designs in one view; as a result, this corresponding action updates other views of the data by selecting, deleting, or masking the same designs. Figure 2-5 demonstrates brushing and linked views by only displaying satellite designs in both plots that have low ΔV values.

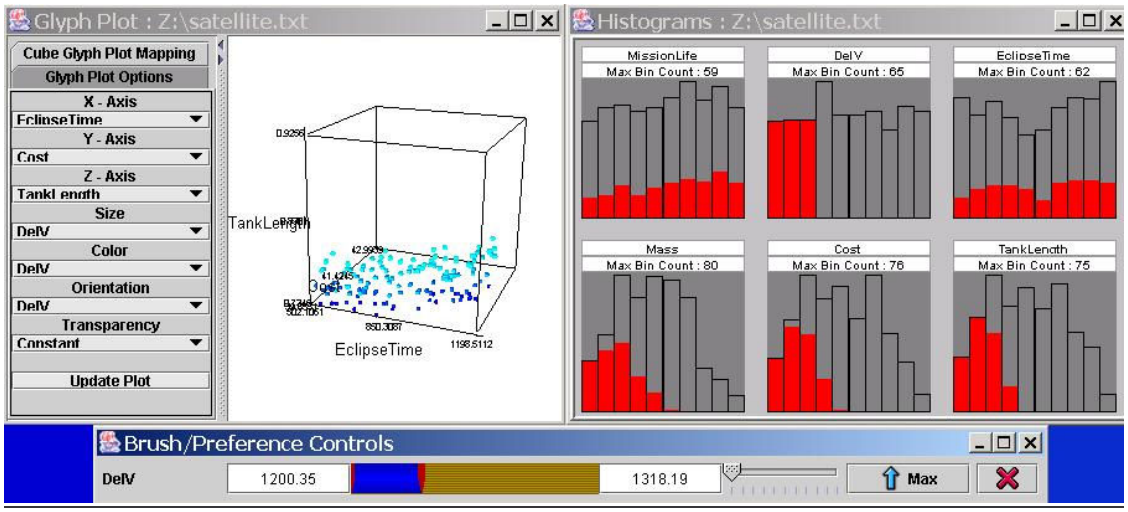


Figure 2-5: Brushing/Linking

2.2.6 Isosurfaces

To display multivariate data in a 2-dimensional or 3-dimensional display, several variables of a dataset can be held constant, and a resulting line or surface can be plotted.

$F(x_1, x_2, x_3, x_4, x_5)$: Cannot plot using 3D surface

$F(x_1, x_2, c_3, c_4, c_5)$: Can plot using a 3D surface, where c_i are constants

Figure 2-6 displays three different isosurfaces.

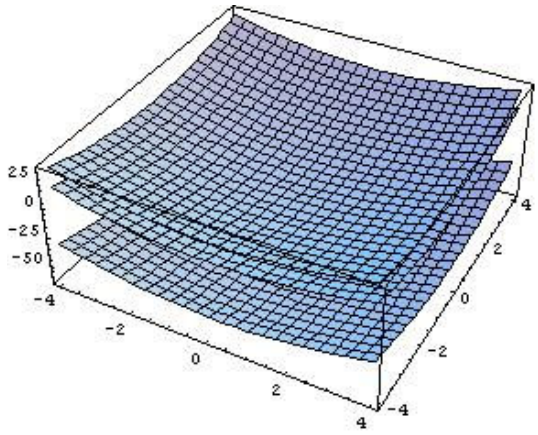


Figure 2-6: Isosurfaces

2.2.7 Dimensional Stacking

Dimensional stacking is a technique in which multivariate data is displayed by embedding graphs within an outer space [19]. The location of the embedded graphs can represent up to three variables of a design. The process can be repeated for a specified number of times until a point, line, or surface can be plotted, due to the reduction of dimensions.

For example, N-vision consists of inner worlds and outer worlds [20, 21]. The location of the inner world determines a 3D vector for the variables represented by the outer world, and by changing the location of these inner plots, the corresponding inner graph in updated. A screenshot of N-Vision is displayed in Figure 2-7.

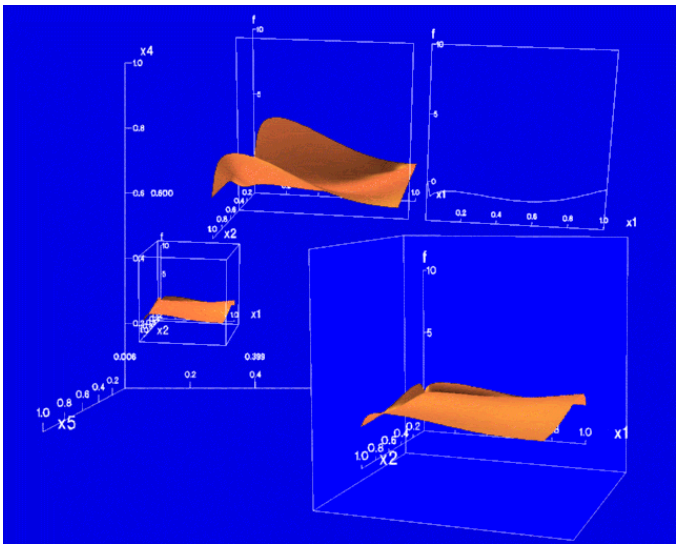


Figure 2-7: N-Vision Screenshot

2.2.8 Grand Tour/Projection Pursuit

The Grand Tour illustrates multivariate data by displaying an interpolation of projections through a time series. Randomly sampled projections are selected in an n-dimensional space, and the Grand Tour interpolates from an initial projection to the next projection. This cycle is repeated.

Projection pursuit seeks to find interesting projections of an n-dimensional space by optimizing an index function [22, 23] Usually, these index functions seek to find projections in an n-dimensional space by finding projections that are non-Gaussian, whether these function search for holes or clusters. Once an optimized projection is found, the projected dataset can be plotted using histograms or scatter plots.

XGobi/GGobi has implemented projection pursuit along with a Grand Tour display [24]. XGobi/GGobi initially selects a random projection and optimizes a projection index; at the same time, a Grand Tour plot is updated for each iteration in the optimization process. As a result, XGobi/GGobi allows the user to visualize the projection pursuit optimization process. Figure 2-8 shows a screenshot of GGobi, where a Grand Tour animation is shown on the left and the corresponding projection pursuit calculations are shown on the right. Once a maximum is reached, a new randomly sampled point is chosen, and the optimization process is repeated.

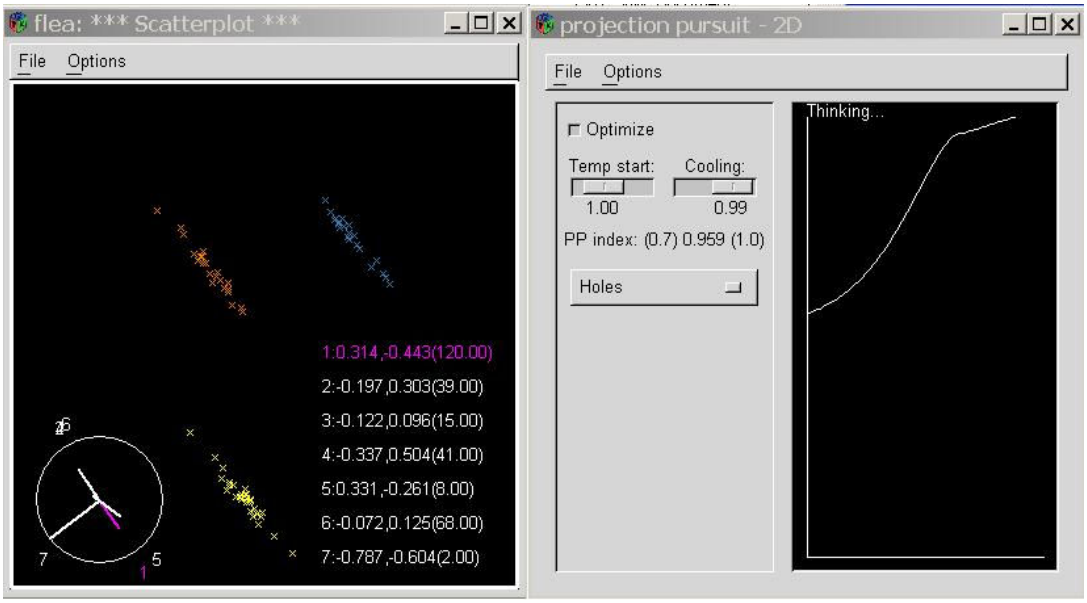


Figure 2-8: GGobi Screenshot of the GrandTour/Projection Pursuit

2.2.9 Trellis Plots

A trellis plot (shown in Figure 2-9) consists of an arrangement of panels, in which each panel displays a subset of the original dataset by slicing variables in the multi-dimensional space [25]. Slicing of a variable refers to the reduction of a dataset, which have values between specified limits. The behavior of these variables is displayed by arranging panels within the trellis plots in an increasing order from bottom to top and left to right.

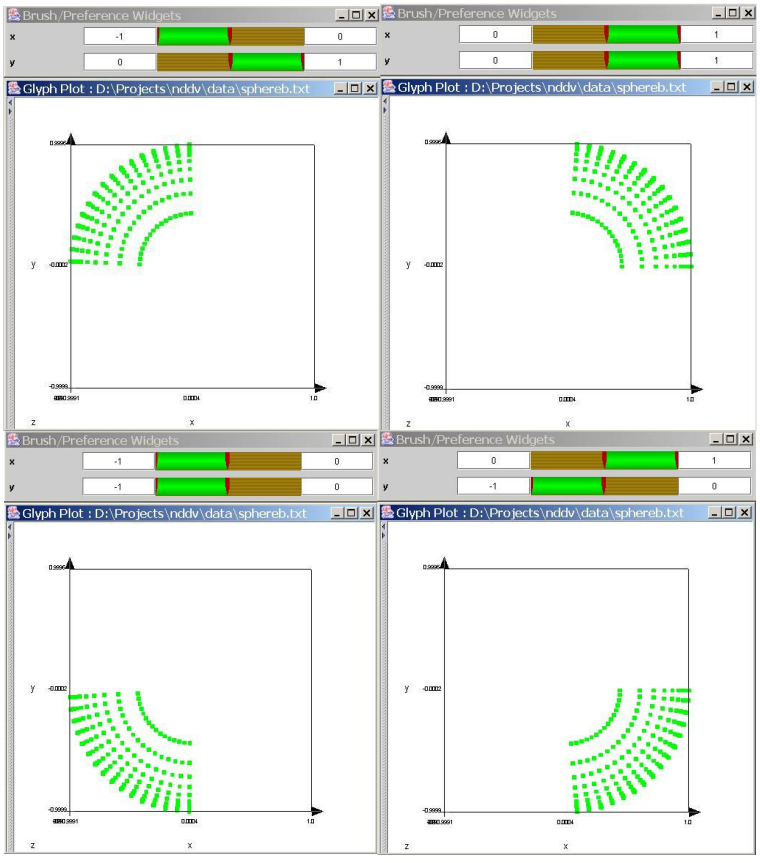


Figure 2-9: Trellis Plot

2.2.10 Histogram Plots

Histograms partition the ranges of a dataset's variables and count the total number of occurrences in each partition. Histograms visualize the mean, skewness, distribution, outliers, and variance of variables within a dataset. Figure 2-10 displays 11 histogram plots, each histogram represents one variable's distribution in a dataset.

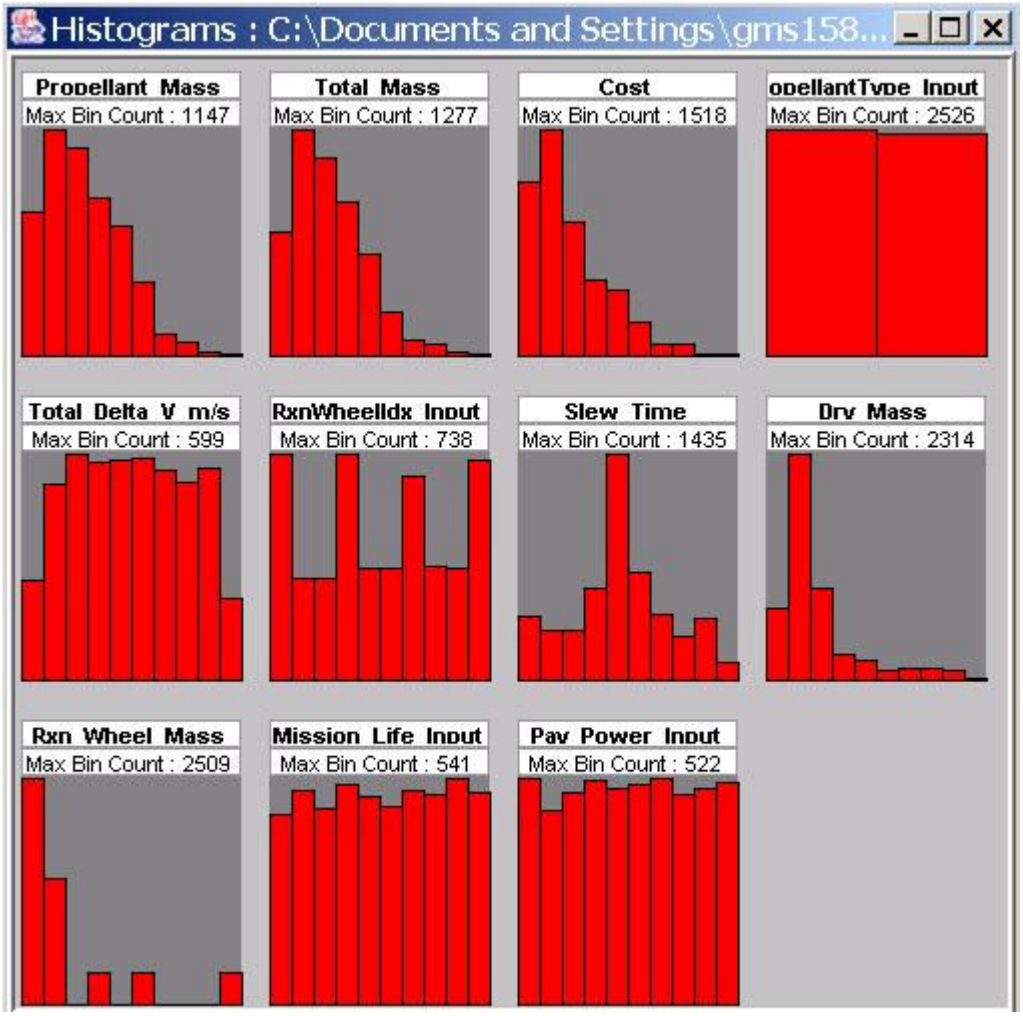


Figure 2-10: Histogram Plots

2.3 Previously Developed Software Used to Visualize Multi-dimensional Data

Several freeware and commercially available software packages, listed in Table 2-2, have been developed to visualize multi-dimensional datasets, and these existing interfaces incorporate many widely used multi-dimensional visualization techniques that include scatter matrices, glyph plots, parallel coordinates, dimensional stacking, reduction of dimensions, linked displays, and brushing.

Table 2-2: Existing Software that Implements Multi-dimensional Visualization Techniques

Software	Visualization Techniques
3DVDM [26]	3D glyph plots, Ability to assign glyph attributes to variables, 3D scatter plot matrix, 3D grand tour
CVis [27]	Cloud visualization, Brushing, Linked displays, Child/Parent windows
Glyphmaker [28]	3D glyph plots, Customizable glyphs, Ability to assign glyph attributes to variables, Linked displays, Brushing
Influence Explorer [29]	Histograms, Brushing, Linked displays, Glyphs
Miner3D [30]	3D glyph plots, Ability to assign glyph attributes to variables, Brushing, Stereoscopic visualization
Mondrian [31]	Parallel coordinates/BoxPlots, Brushing, Linked displays, Scatter plots, Bar charts, Histograms, Mosaic plots
N-Vision [20][21]	Isosurfaces, Child/Parent windows, Multiple displays, Glyphs
Partek Pro [32]	3D glyph plots, Brushing
Spotfire DecisionSite [33][34]	3D glyph plots, Ability to assign glyph attributes to variables, Brushing, Parallel coordinates, Trellis plots, Linked displays, Histograms, Bar charts, 2D and 3D scatter plots
Virtual Data Visualizer [35]	3D glyph plots, Customizable glyphs, Ability to assign glyph attributes to variables, Brushing
VisDB [36]	Pixel oriented techniques, Glyphs (spiral, axes, and grouping techniques), Parallel coordinates, Stick figure glyphs
Visual Mine [37]	3D glyph plots, Ability to assign glyph attributes to variables, Brushing, External charts display additional dimensions
XdmvTool [38]	Scatter matrix, Star glyphs, Assign glyph attributes to variables, Brushing, Parallel coordinates, Dimensional stacking, Linked displays
Xgobi/Ggobi [39]	Scatter plots, Scatter matrix, Parallel coordinates, Glyphs, Ability to assign glyph attributes to variables, Grand tour, Projection pursuit, Brushing, Linked displays, Labeling

2.3 Stereoscopic Visualization/Virtual Reality

2.3.1 Stereoscopic Visualization

Eye offset, called parallax, causes each eye to see different images, and the mind combines the two images to perceive depth. To achieve stereoscopic visualization on a computer screen or projected image, the image must be separated into two different views, allowing each eye to only see its corresponding image. Hardware such as projection screens, projection walls, desktop monitors, panoramic walls, projection tables, L-shaped projection tables, and immersive four-sided rooms have been used to visualize stereoscopic models and images.

2.3.1.1 Active Stereoscopic Visualization

Computer monitors and projectors can display stereoscopic images, by rapidly alternating left and right eye images in synchronization with LCD shutter glasses, as shown in Figure 2-11. An emitter, connected to a stereo-compatible graphics card, sends a signal, indicating which image is displayed on the screen, to the shutter glasses. When the monitor displays the right-eye image, shutter glasses block the left eye, allowing the right eye to view the image. The process is reversed when the monitor displays the left-eye image. Figure 2-12 displays an example stereo signal, where the right-image and left-image are displayed in succession. The stereo signal should be greater than 90 Hz to avoid screen “flickering”.

Figure 2-11: Active Stereoscopic Visualization Using LCD Shutter Glasses

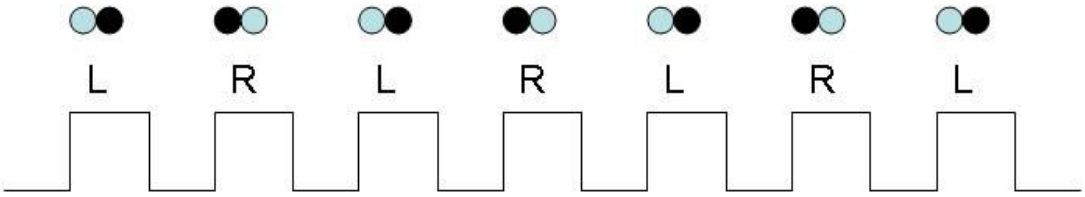


Figure 2-12: Stereo Signal Synchronized with LCD Shutter Glasses

2.3.1.2 Passive Stereoscopic Visualization

A passive stereoscopic visualization setup is displayed in Figure 2-13. A converter takes a frame sequential stereo signal and splits this signal into two separate signals, one for the left eye and one for the right eye. The two separate signals are then sent to two projectors, in which each has a polarized filter placed in front of the lens. The left and right eye images are projected simultaneously onto a polarization-preserving screen, where each eye image is displayed using two different polarized light orientations, with a 90^{o} angle of separation. Users wear polarized glasses that filter unwanted light, allowing each eye to only see its appropriate image.

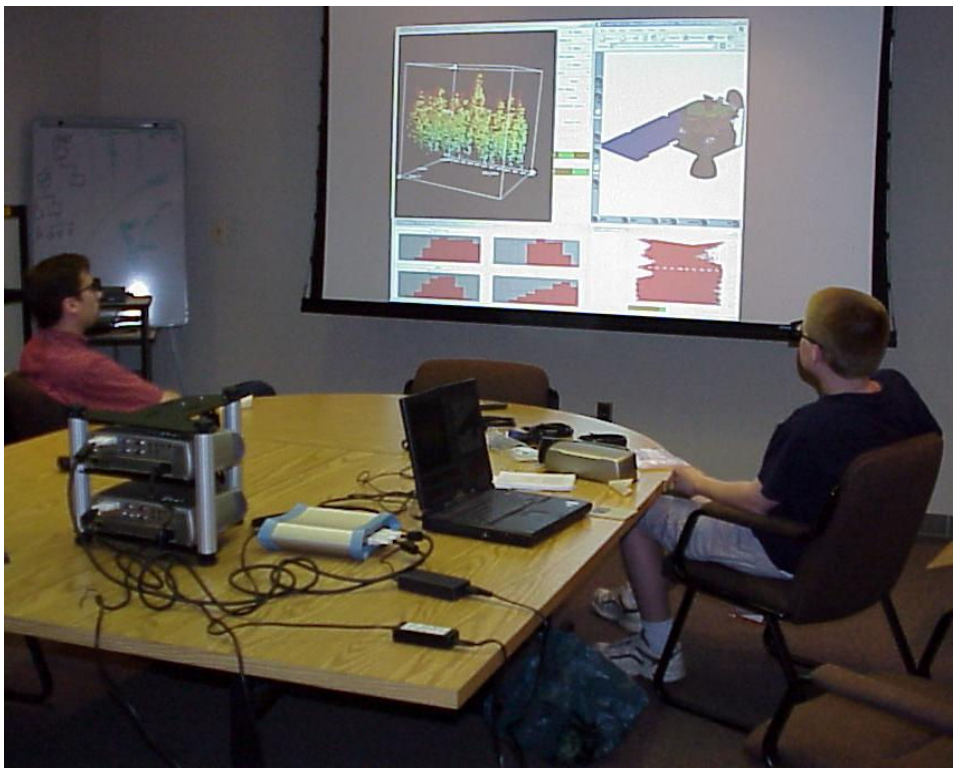


Figure 2-13: Passive Stereoscopic Setup

2.3.2 3D Immersive Virtual Reality Environments

Users, wearing LCD shutter glasses, are placed within a four-sided room (some use six sides), with each side displaying a projected stereoscopic image. Advanced hardware, such as gloves, head trackers, and wands are used within these environments to interact with virtual worlds. Shown in Figure 2-14, the Applied Research Laboratory at Penn State has an immersive virtual reality environment, the Synthetic Environment Applications Laboratory (SEA Lab) [40]. The dimensions of the room are 10’ x 10’ x 9’, and its primary functions are advanced visualization, simulation, and collaborative technologies.

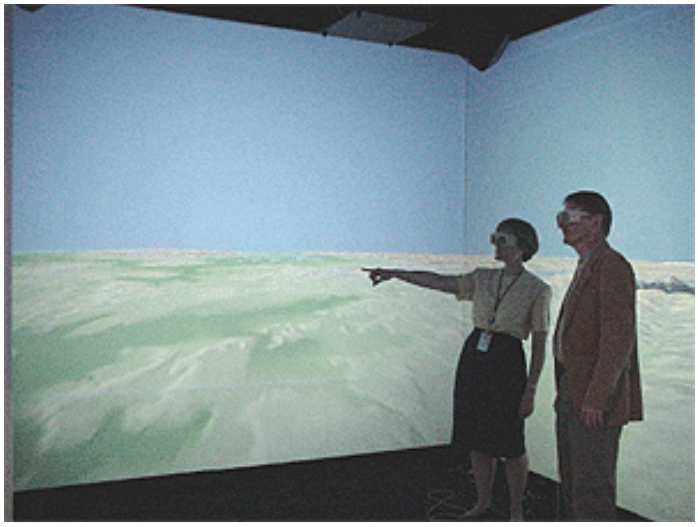


Figure 2-14: SEA Lab Illustration [40]

2.4 Summary

This chapter has discussed previous work on graphical user interface development, multi-dimensional visualization techniques, and stereoscopic visualization. These areas of research provide background information pertaining to the ATSV and I-Beam experiment. The next chapter discusses a Human Computer Interaction (HCI) experiment that analyzes the tradeoff between stress and cross-sectional area of an I-Beam design problem.

Chapter 3

I-beam Design Exercise

This Human Computer Interaction (HCI) study analyzes the effectiveness, efficiency, and satisfaction graphical users interfaces (GUI) have on an I-beam design problem. A brief introduction to the I-beam design problem, a description of the experimental setup, and results of the experiment are discussed. Finally, conclusions and a summary are given at the end of the chapter.

3.1 Introduction

The purpose in this study is to investigate user interaction with a graphical design environment (GDE) with varying response times and to measure the impact of using a GDE on user learning about resolving design tradeoffs. This example was adapted from a problem by Haftka and Gürdal [41] where the users attempt to determine the optimal design for an I-beam subject to a bending stress. The GDE was developed using Visual Basic 6.0 to visualize the effect of changing the geometry of the I-beam cross-section on the stress and cross-sectional area. The design exercise was completed by students in two graduate courses at Penn State: *Optimal Structural Design* (ME 597I) and *Using Simulation Models for Engineering Design* (IE 578). In the I-beam GDE, users are able to change both the height (h) and width (w) of the I-beam, thus changing the cross-sectional area (A) and the imposed bending stress (σ), which are calculated using Eq. 3.1

and Eq. 3.2. The thickness (t) and moment (M) of the I-beam remain constant. As the user changes ***h*** and ***w*** using slider bars, the corresponding ***A*** and $\boldsymbol{\sigma}$ response appear in the performance space (*xy*-plot of ***A*** vs. $\boldsymbol{\sigma}$). Since ***A*** and $\boldsymbol{\sigma}$ are competing design objectives, the user seeks to resolve tradeoffs between them by finding the best combination of ***h*** and ***w*** that minimize both ***A*** and $\boldsymbol{\sigma}$. This portion of this experiment is called the free-form case.

$$\mathrm{A} = 2wt + t(h - 2t) \tag{3.1}$$

$$\sigma = \frac{M\frac{h}{2}}{\frac{wh^3 - (w-t)(h-2t)^3}{12}} \tag{3.2}$$

In the second portion of the experiment, the user seeks to resolve tradeoffs between ***A*** and $\boldsymbol{\sigma}$ using a weighted sum approach, as shown in Eq. 3.3, where normalized measures of ***A*** and $\boldsymbol{\sigma}$ are used in order to avoid scaling problems in the plot. Here, ***F*** is a weighted sum of the normalized objectives, Eq. 3.3, $\boldsymbol{\alpha}$ is a scalar weighting factor ranging from 0 to 1, $\boldsymbol{A_{max}}$ and $\boldsymbol{A_{min}}$ are the maximum and minimum possible areas, respectively, and $\boldsymbol{\sigma_{max}}$ and $\boldsymbol{\sigma_{min}}$ are the maximum and minimum possible stresses, respectively. Users seek to determine the best combination of ***h*** and ***w*** that minimize ***F*** for a particular value of $\boldsymbol{\alpha}$. The weighted sum is appropriate for handling the multi-criteria optimization problem in this case because the problem is convex.

$$\min_{\substack{0.2\le h\le 10\\ 0.1\le w\le 10}} F = \alpha \frac{(A - A_{min})}{(A_{max} - A_{min})} + (1-\alpha)\frac{(\sigma - \sigma_{min})}{(\sigma_{max} - \sigma_{min})} \qquad 3.3$$

Users can investigate the quality of a particular design by using the mouse to click on a point in the performance space. The objective contour line (line of constant F with slope α) through that point appears, and the corresponding value of F is displayed. A contour line lies tangent to the optimal design point. The interface is pictured in Figure 3-1 for $\alpha = 0.9$.

In this portion of the experiment, the impact of the delay time in the display of the performance response is also assessed. The delay time for the response to appear in the performance space is varied during the experiment to be either 0.0, 0.1, or 0.5 seconds. It is important to note that for this simple I-beam example, the analysis is virtually instantaneous, making it easy to study the effect of response delay, since delay can be artificially imposed. For large complex systems, detailed analyses dictate large response delays, and rapid response can only be achieved by using approximations called *metamodels* [42, 43].

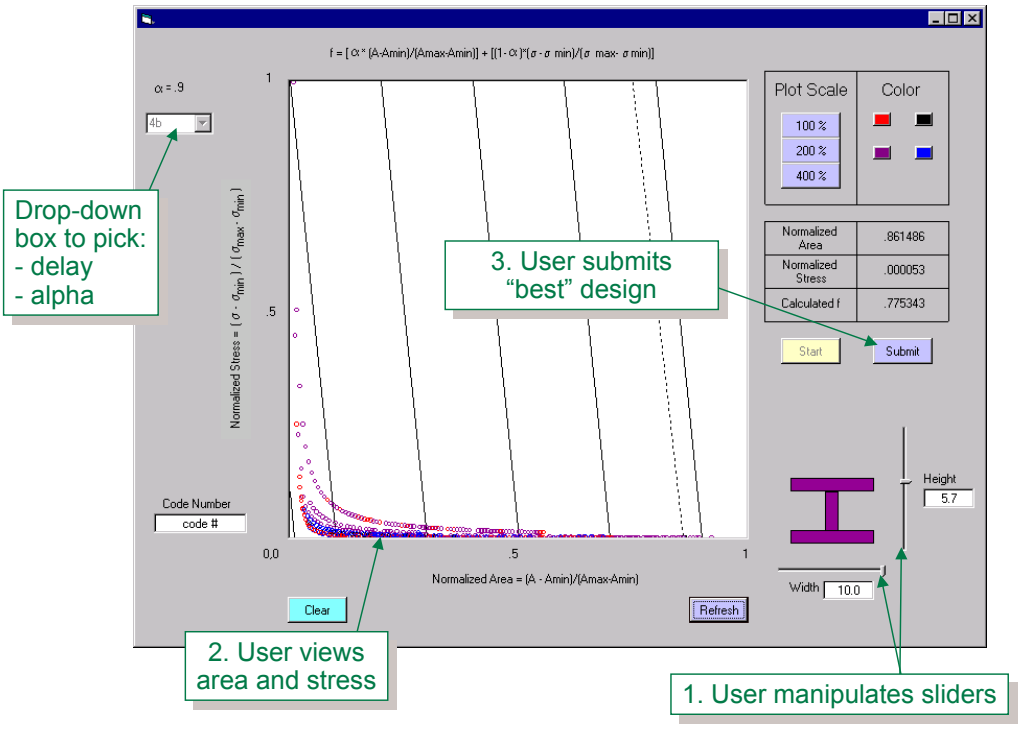


Figure 3-1: Graphical User Interface for I-Beam Design

3.2 Experimental Protocol

Eighteen graduate students designed I-beam cross sections using the graphical design environment shown in Figure 3-1. Users were asked to manipulate slider bars to adjust design parameters, h and w, to resolve tradeoffs between two competing objectives, σ and A, and then identify the "best" design. During the weighted sum optimization, the delay before the graphical window was updated after each design change was controlled. The experimental setup is pictured in Figure 3-2. Users were videotaped during the experiment, and the software recorded the following data for each user:

1. the final designs submitted for each exercise (i.e., design variables, stress, and area),
2. the time to obtain each design, and
3. the time spent on each slider bar.

Figure 3-2: Experimental Setup

Each user required approximately 15-30 minutes to complete the design exercises. The experiment began with an overview of the experiment, an informed consent form, and a pre-test questionnaire to determine the user's familiarity with multi-criteria optimization and computer literacy. Users then completed a simple training example to become familiar with the software. The first task was the free-form design exercise. Users were videotaped during this portion of the experiment and asked to speak aloud and articulate their thinking while designing. Users then completed a mid-test questionnaire regarding ease of use of the software, satisfaction with selected design and anxiety level. Next, users were asked to identify three designs based on three weighted-sum combinations of the two competing objectives (α = 0.1, 0.5, 0.9). The order in

which each user received the three α values and the magnitude of the time delay in the software were varied for each user. The users were also videotaped during this portion of the experiment and asked to speak aloud and articulate their thinking while finding the best design for each α value. The experiment concluded with a post-test questionnaire, where users were asked to rate the ease of use of the software, their designs and their design process, their learning between tasks, the impact of response delay on the design process, the impact of being videotaped. Suggestions for improving the experiment and/or the software were also recorded.

3.3 Results and Analysis

The preliminary analysis of user performance presented in this section follows the segments of the experimental protocol: pre-test questionnaire, free-form design exercise, mid-test questionnaire, design exercises with different weighting (α) values, and post-test questionnaire.

3.3.1 Pre-Test Questionnaire

Table 3-1 summarizes the mean and standard deviation of the user responses to the pre-test questions, where most questions were rated on a scale of 1 to 5. Users felt that they had a fairly extensive knowledge of computers and used them for 10 to 40 hours each week. They were generally familiar with multi-criteria optimization, since this topic

was covered in both graduate courses, and one third had previously developed graphical user interfaces for computer programs.

Table 3-1: Pre-Test Questionnaire Responses

Question	Mean	Std. Dev.
Computer knowledge	3.7	0.8
Weekly computer usage (hours)	26.7	14.0
Video games	2.8	1.1
Understanding of computers	3.7	0.6
Familiarity with multi-objective optimization	3.3	0.7
Previously develop a Graphical Design Environment	5 Yes	13 No

Based on these responses, we expected that the short training period in this test would be sufficient to prepare these users for the design exercises; however, the results presented in the following sections suggest that users continued to learn how to use the graphical design environment during the first two design exercises.

3.3.2 Free-Form Design Exercise

For the first exercise, users were asked to find a design to simultaneously minimize both stress and area. Figure 3-3 shows the normalized stress and area values for the designs created in the first exercise, which are compared with designs created earlier in the semester without the real-time graphical user interface. All of the designs fall close to the Pareto frontier, but designs created using the graphical interface show less variation and are generally better (i.e., closer to the utopia point). The graphical design environment helped the users resolve tradeoffs between competing design

objectives, and the questionnaire responses support this interpretation of the results (see Table 3-2).

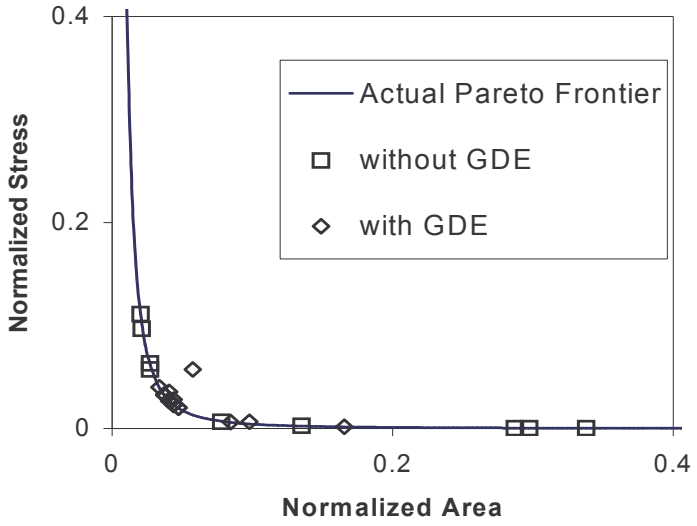


Figure 3-3: Free-Form Normalized Area vs. Stress for Graphical and Non Graphical Exercises

3.3.3 Mid-Test Questionnaire Responses

After completing the free-form design exercise, users thought that the graphical interface helped them find a good design, and they were confident that they had found a good design. They thought that the graphical design environment was easy to use. There were no questions relating to response delay time for the free-form design since no response delays were introduced during this design exercise.

Table 3-2: Mid-Test Questionnaire Responses

Question	Mean	Std. Dev.
Software helped make tradeoffs	4.0	0.6
Confidence in final design	3.8	0.6
Software easy to use	4.6	0.5
Frustration with software	0.4	0.6

3.3.4 Design Exercises with Three Different Preference Structures

For the next three design exercises, users were asked to minimize an aggregate objective function that was a weighted sum of normalized area (weight = α) and normalized stress (weight = 1 - α). The balanced Latin square experiment design prevented confounding of learning effects with effects due to the weight (α = 0.1, 0.5, 0.9) and delay (= 0.0, 0.1, 0.5). Figure 3-4 shows that the relative weights for the two objectives did not have a significant effect on the quality of the design. On the other hand, Figure 3-5 may indicate a learning effect: the largest percentage errors in the users' designs generally occurred during the first of these three exercises, regardless of the value of α. Meanwhile, Table 3-3 shows that the second and third trials tended to have lower errors than the first (3.61% vs. 7.10%); however, the p-value testing this hypothesis was not significant at the 5% level. Figure 3-5 also shows a general trend indicating that large errors occurred infrequently, but when they did, users who spent more time on to complete each design exercise tended to have lower errors. Figure 3-6 shows a related phenomenon: users who performed a larger number of controller actions tended to have

higher quality designs (smaller percentage error from the optimal objective function value). The effect of learning is also apparent in this figure, since the second and third trials often had lower error for a similar number of alternatives examined.

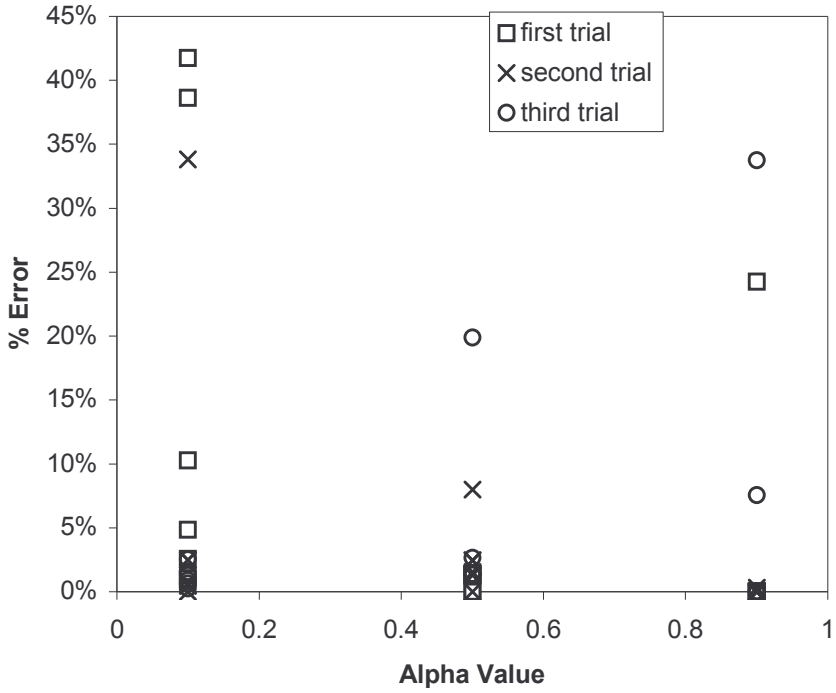


Figure 3-4: Percent Error in Objective Function vs. Weighting Factor

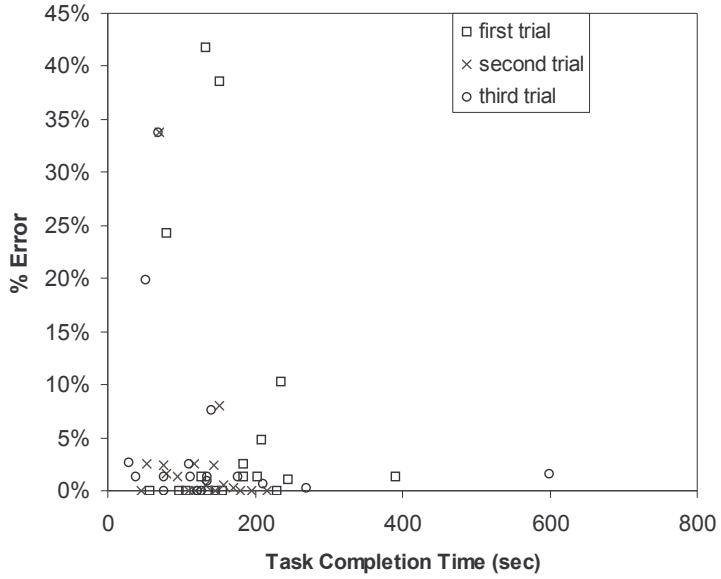


Figure 3-5: Percent Error in Objective Function vs. Time on Task

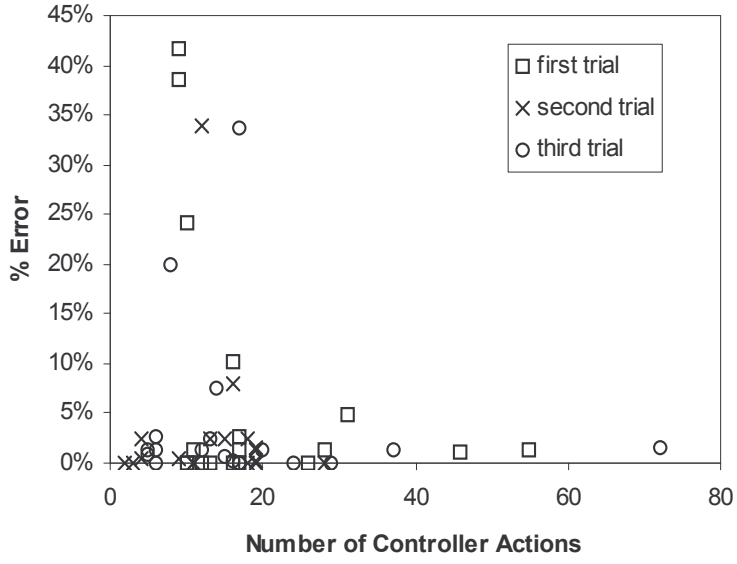
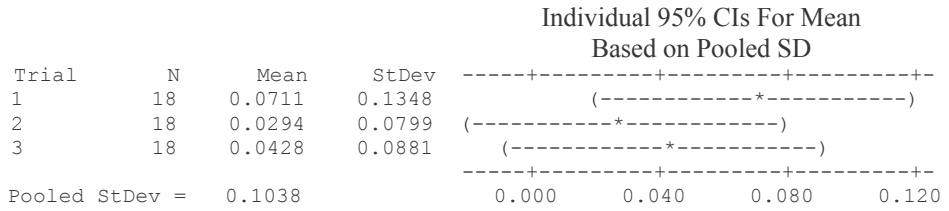


Figure 3-6: Error vs. Number of Controller Actions

Table 3-3: Statistical Data for the Learning Effect

Trial	N	Mean	SD	SE Mean
1	18	0.0710	0.1350	0.032
2,3	36	0.0361	0.0832	0.014

P = 0.32



A primary objective in this study is to determine whether or not small delays in system response affect the design process. Figure 3-7 shows that a delay as small as 0.1 second may cause deterioration in the quality of a design, in terms of percent increase over the optimal (weighted) objective function value. It appears that the advantage of a graphical design interface for improving design quality depends on the ability to produce near-instantaneous responses to design parameter changes. Shown in Table 3-4, the average percent error increased from 2.56% to 5.90% when delay was introduced. This difference was not statistically significant with a p-value of 0.17. On the other hand, Figure 3-8 shows that in these exercises, added response delay did not have a significant impact on the time to complete the design task.

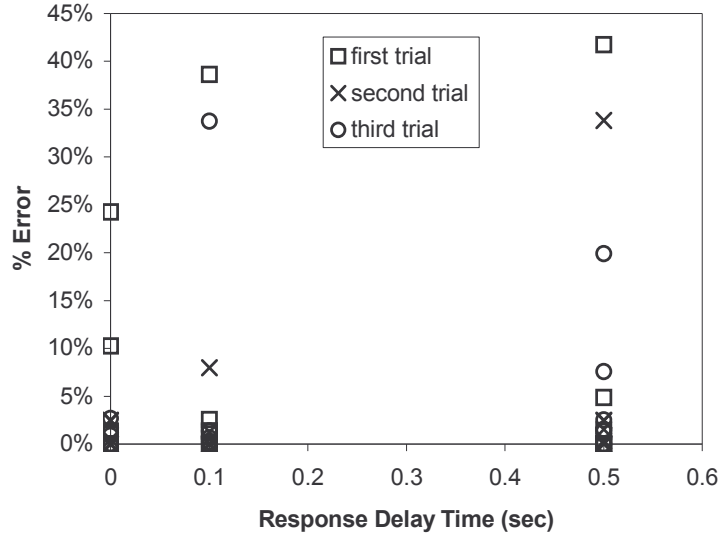


Figure 3-7: Percent Error vs. Response Delay

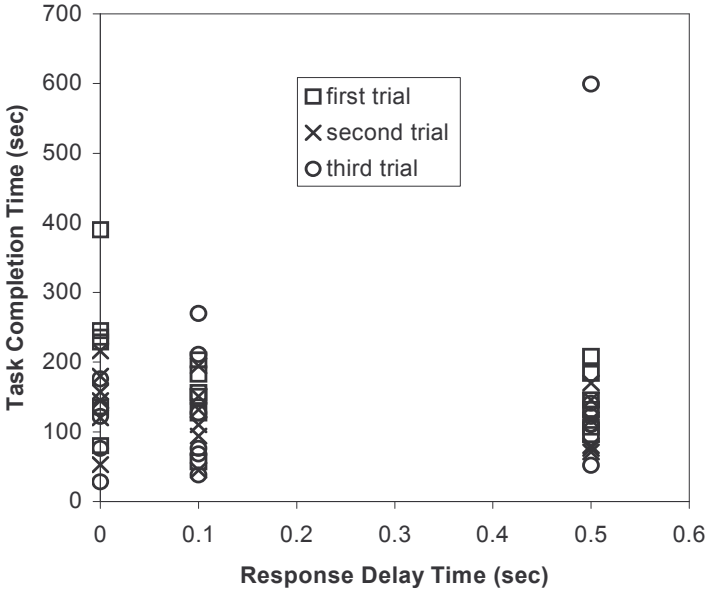
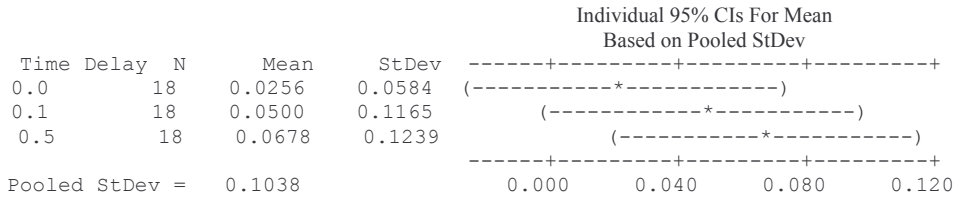


Figure 3-8: Time to Complete Task vs. Response Delay Time

Table 3-4: Statistical Data For Time Delay Effect

	N	Mean	SD	SE Mean
No Delay	18	0.0256	0.0584	0.014
Delay	36	0.0590	0.1190	0.020

P = 0.17



3.3.5 Post-Test Questionnaire Response

The average and standard deviation of responses to the post-test questions are listed in Table 3-5. The post-test questionnaire showed that the users thought that the software made it easy to resolve tradeoffs between the competing objectives of area and stress, although there was no uniform view of which design exercises were more difficult: the weighted-sum exercises or the free-form design. The weighted-sum exercises provided objective function contour lines to help users select a design, and most of them found this feature very useful. All but two of the users rated their confidence in their final designs as 4 or 5. All of the users rated the software environment highly, and none of them felt that the videotaping and questions disrupted their design process.

Table 3-5: Post-Test Questionnaire Response

Question	Mean	Std.Dev.
Software helped make tradeoffs	4.3	0.8
Confidence in final designs	4.0	0.8
Software easy to use	4.5	0.5
Frustration with software	0.8	1.1
Impact of response delay on choosing designs	2.4	1.3
How much did the contour lines help	4.3	1.0
Ease of use of slider bars and zoom	4.3	0.9
Ease of picking and submitting points	4.8	0.4
Did videotaping/talking interfere with your tasks	2.1	0.5
Satisfaction with part I (free-form) design	3.5	0.9
Which task (free-form or weighted sum) was easier	8F	10W
Rate your overall understanding of the problem	4.1	0.5

Users were not informed of the amount of response delay that would occur in each weighted-sum exercise, but they were asked to identify whether delays had any impact on the design process. The responses generally indicated either no effect or a small-to-medium effect.

3.4 Conclusions and Summary

The study has provided many insights based on a simple design exercise using a graphical design environment. First, users appreciated the graphical design interface. For the free-form design exercise, designs created with the graphical interface were more consistent and of higher quality than designs created without the interface Figure 3-3. Response time delay appeared to affect design quality, but it did not affect time to complete the design task; however, users considered fewer design alternatives as

response time delay increased. Some users needed more time to become familiar with the graphical interface as evidenced by improved second and third trial design quality and reduced time to complete the design task. Users appeared to have a better understanding of resolving tradeoffs during design after using the graphical design interface.

This experiment analyzes a two dimensional I-beam design problem with given *a priori* preference structures. The next chapter will discuss the Advanced Trade Space Visualizer (ATSV), that extends this work by allow a decision-maker to visualize more than two design parameters. In addition, the ATSV allows users to visualize different preference structures after viewing the trade space until a desired preference structure is formed, thereby classifying the process as an *a posteriori* articulation of preference.

Chapter 4

The ATSV Interface and Design Examples

The first section in this chapter illustrates the capabilities of the Advanced Trade Space Visualizer (ATSV). Then, a demonstration that extends the I-beam design problem from Chapter 3 is given, by visualizing more than two performance variables. A second demonstration, based on a conceptual model of satellite design, further illustrates the capabilities of the ATSV. A summary of the ATSV is given at the end of the chapter.

4.1 ATSV Interface

The following functionality was incorporated into the Advanced Trade Space Visualizer (ATSV):

1. Visualize complex datasets using multi-dimensional visualization techniques
2. Assign variables to glyph, histogram, and parallel coordinates plots
3. Specify upper and lower bounds of an n-dimensional design space
4. Implement dynamic brushing within glyph, parallel coordinates, and histogram plots to uncover relationships in the dataset (linked views)
5. Visualize different regions of interest, using preference shading and corresponding Pareto frontier identification
6. Create multiple views of glyph, histogram, and parallel coordinates plots of the same trade space
7. Select a design from the glyph plot to display quantitative information, 3D geometries, and other files such as images and documents
8. Use advanced visualization hardware to view graphs and 3D geometries in stereo mode

Specific capabilities of the ATSV interface are discussed in the following sections.

4.1.1 Glyph/Histogram/Parallel Coordinates

The ATSV displays multivariate information using glyph, histogram, and parallel coordinate plots. Illustrated in Figure 4-1, the ATSV displays up to seven dimensions using position, size, color, orientation, and transparency of cubed glyphs. The spatial position of each glyph cube represents three dimensions of an individual design. The size of the glyph displays an additional dimension, which allows for a qualitative analysis in the trade space [13]. A fifth dimension is represented by the glyph cube's color, in which blue cubes represent low values and red cubes represent high values, similar to a temperature scale. Glyph cube orientation displays an additional dimension, in which higher values are represented by cubes rotated about the x, y, and z axes. The final dimension is represented by transparency, where lower values are more transparent. The user can apply a constant value to any of these physical characteristics, thereby removing a dimension from the glyph plot display.

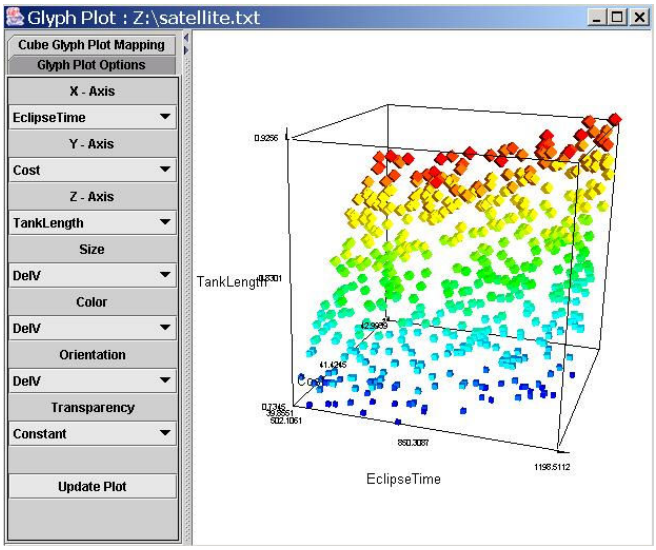


Figure 4-1: Glyph Plot

Using the ATSV, a decision-maker selects variables to be included in a histogram plots display, and a resulting window appears as shown in Figure 4-2. The overall variable distribution is displayed using black outlines, while the brushed distribution is displayed using red shaded areas.

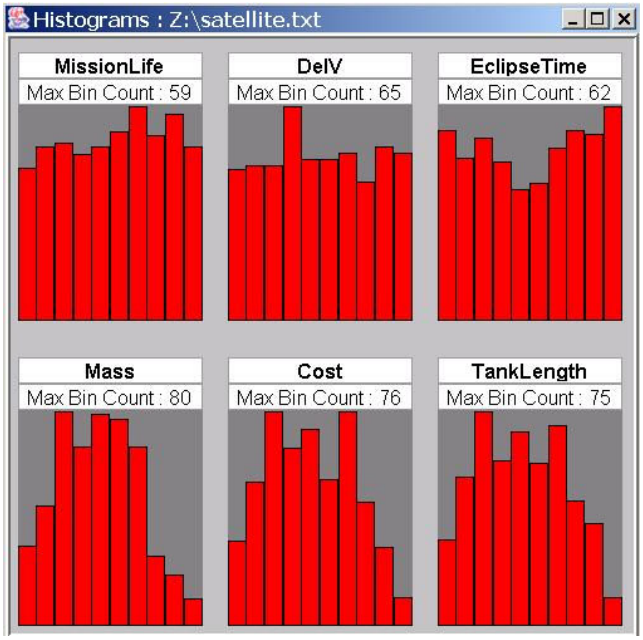


Figure 4-2: Histogram Plots

As with the histogram plot display, a decision-maker can select variables to be included in the parallel coordinates plot. The ATSV displays parallel coordinates that are orientated horizontally (shown in Figure 4-3), where low and high values correspond to polyline intersections on the left and right side of the axes, respectively.

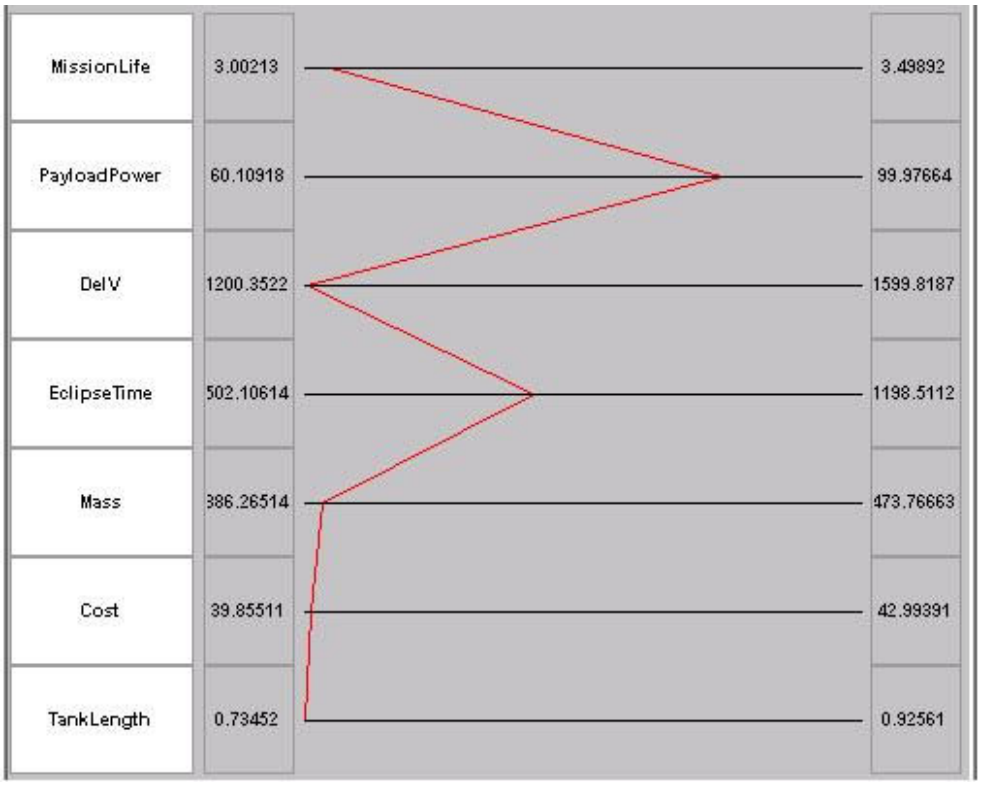


Figure 4-3: One Design in a Parallel Coordinates Plot

4.1.2 Brushing/Linked Views

An ATSV user can brush a trade space, using the range slider bars shown in Figure 4-4**.** These range slider bars set upper and lower limits for each variable's range, only displaying a subset of the trade space. Along with setting range limits, a user can dynamically brush across a variables range, by dragging the range slider bar between variable limits.

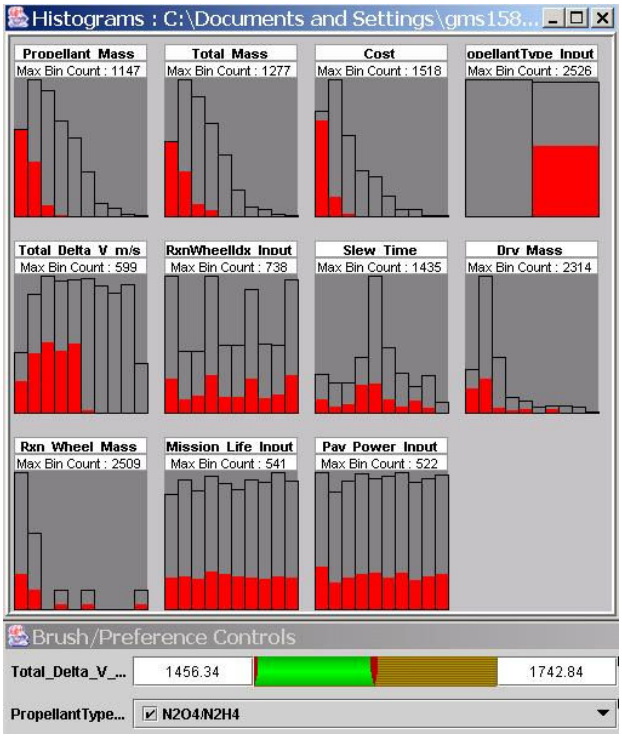


Figure 4-4: Histogram Plots/Brushing

4.1.3 Preference Shading/Pareto Frontier Display

Preference shading allows a user to experiment with different preference structures, observing dynamically how the designs order themselves in response. The designs are then sorted using a standard weighted-sum approach:

$$f = \sum_{i=1}^{n} w_i \, x_i \qquad 4.1$$

Eq. 4.1 is used to set the preference value of each design, with higher values being more preferred designs and lower values being less preferred designs. Figure 4-5 displays two different highlighted regions that correspond to preference structures indicated by the

arrow. Preference shading can be applied to any glyph plot feature, such as position; size; color; orientation; or transparency, which is shown in Figure 4-5.

The number of highlighted designs is controlled by a range slider bar, in which the user can set upper and lower limits on preference shading. A user has the ability to view a subset of preferred designs, whether the designs are most preferred, least preferred, or fall within a middle range of preferred designs. In addition, the user can dynamically brush through different levels of preferred designs.

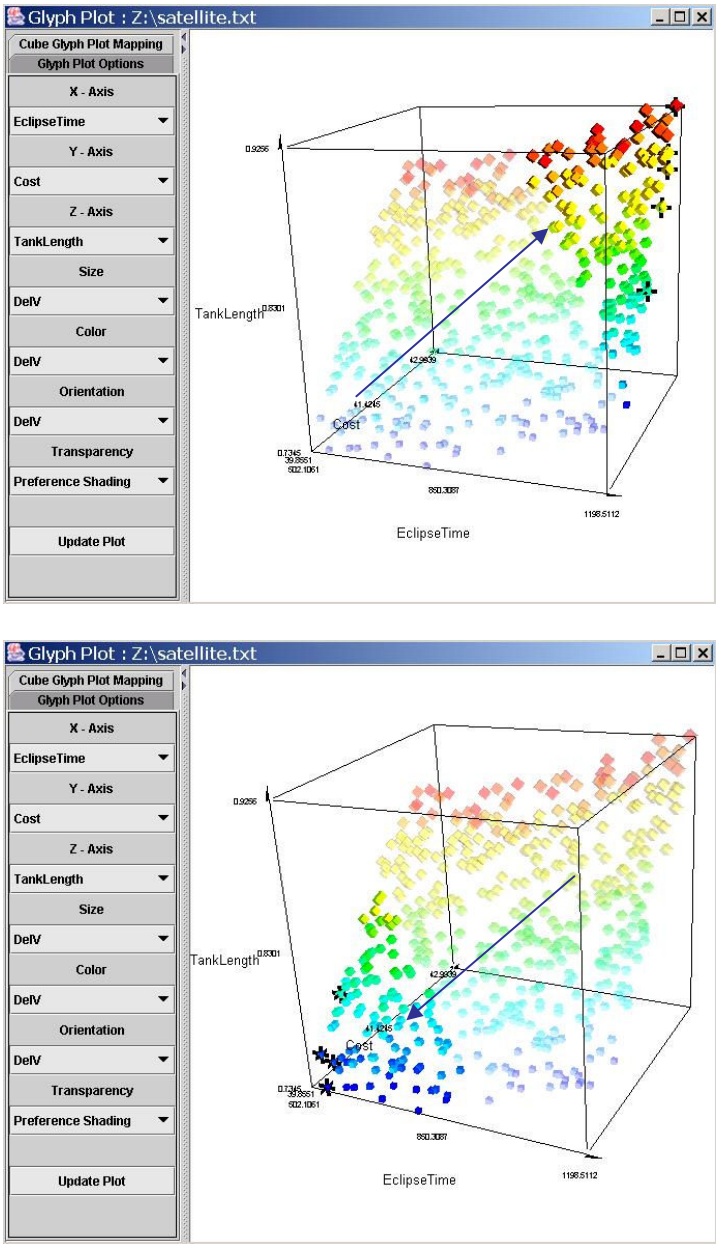


Figure 4-5: Preference Shading/Pareto Frontier Display in the ATSV

While preference shading uses slider bars to smoothly vary a weighting vector, the only aspect used for determining the Pareto frontier is whether more or less of the attribute is preferred, or if the attribute is to be ignored (i.e., weight equal to zero). Using

the same preferences as displayed in Figure 4-5, designs that fall on the Pareto frontier are distinguished using black markings.

4.1.4 Design Selection

A user can select an individual glyph (double mouse click) to display additional information of a satellite design. A new frame, shown in Figure 4-6, appears and displays all associated information, such as quantitative information, 3D geometries, images, and documents of the selected satellite design.

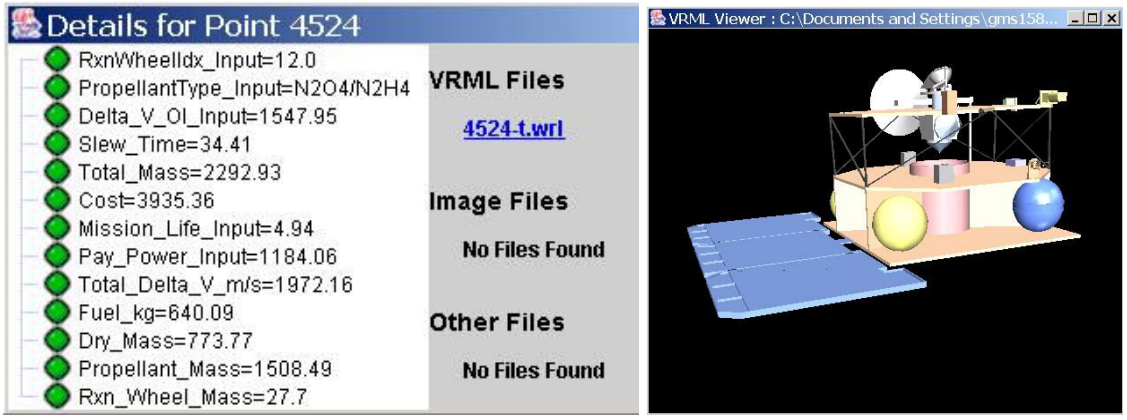


Figure 4-6: Quantitative Information and 3D Geometry of a Selected Point Design

4.1.5 Program Architecture

The ATSV front-end is displayed in Figure 4-7. The ATSV program architecture includes the Visualization Toolkit (VTK) [44], JFC/Swing, and Java3D (see Figure 4-8). VTK includes a Java wrapper that accesses VTK core classes, thereby allowing one to use JFC/Swing coupled with VTK. JFC/Swing is platform-independent language, which

is used to develop graphical user interfaces that have features such as scroll bars, combo boxes, internal frames, and menus. Additionally, 3D geometry files with a VRML extension are displayed using a Java3D VRML loader.

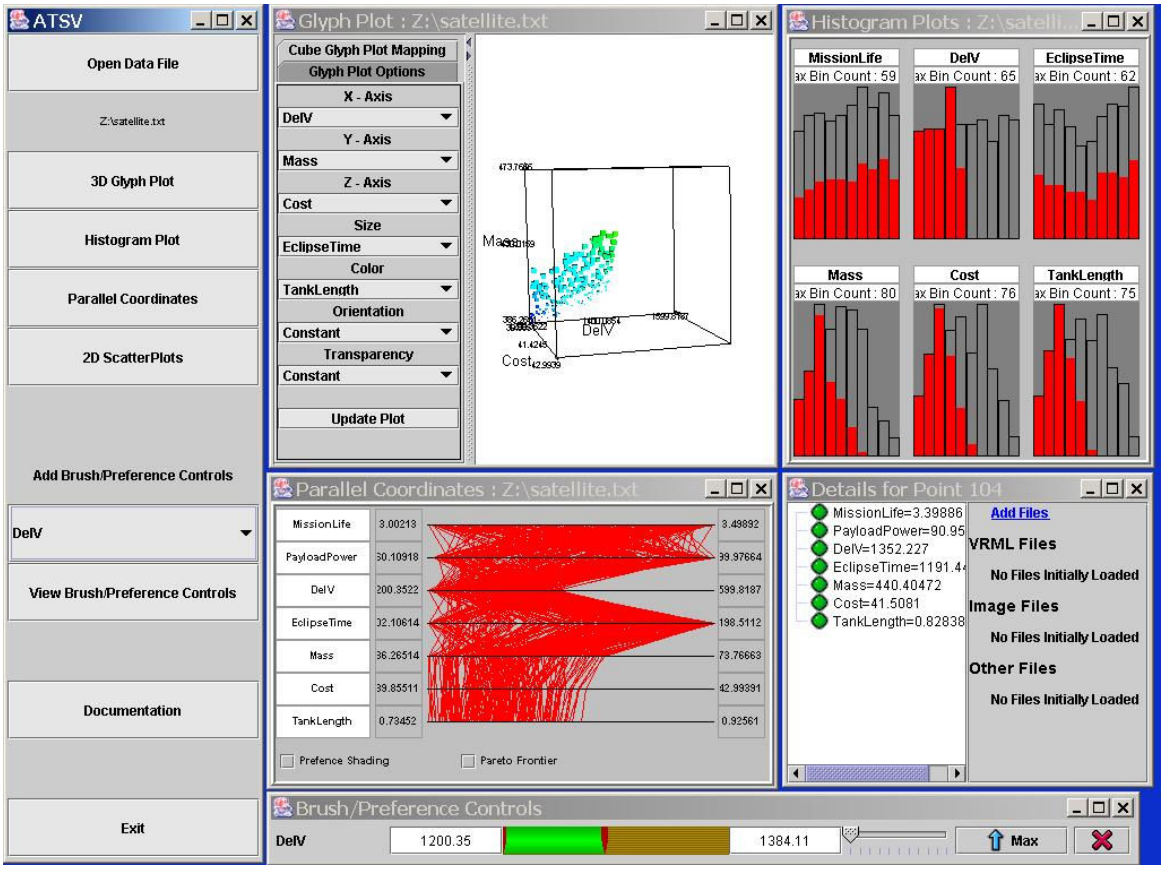


Figure 4-7: The ATSV Graphical User Interface

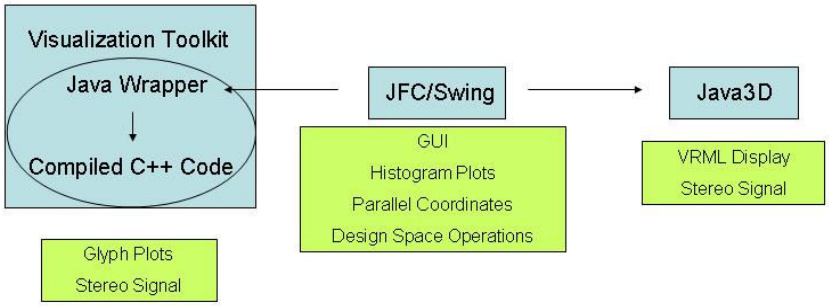


Figure 4-8: Program Architecture for the ATSV

The program architecture has been structured to display multiple windows simultaneously, allowing a user to view several different representations of the same data. Actions within individual graphical displays are independent from other plots. A user can toggle stereo mode in individual glyph plots and 3D geometry windows. In contrast, all glyph, histogram, and parallel coordinate plots are linked together, by only displaying designs that lie within brush limits; additionally, a user-specified preference structure and corresponding Pareto optimal designs are displayed in all plots.

4.1.6 Virtual Reality

VTK, along with a stereo compatible graphics card, outputs a signal in frame sequential stereo signal format; this signal can be used by desktop computers, stereo projectors, and immersive virtual reality environments to visualize stereoscopic images. The ATSV interface utilizes the advanced visualization setups, shown in Figure 2-11 and Figure 2-13.

4.2 I-Beam Design Example

The conceptual model for an I-beam design problem was developed using the following commercially available software:

- Microsoft Excel 2002
- Mathematica 5.0
- MathematicaLink For Excel
- Crystal Ball

The Microsoft Excel sheet shown in Figure 4-9 displays the input, calculated, and output variables of the I-beam design problem, and the design rules used within the I-Beam conceptual models are displayed in Eqs. 4.2-4.10. The input variables to the design problem are the width, height, thickness, material, and length of the I-beam. A calculated variable, moment of inertia, is used to compute the maximum bending moments within the I-beam. The other calculated variables, modulus of elasticity, density, yield strength, and cost factor are retrieved from a lookup table. The output variables are bending stress, shear stress, and safety factor, which are calculated at five different locations within the I-beam; the remaining output variables are cross-sectional area and cost.

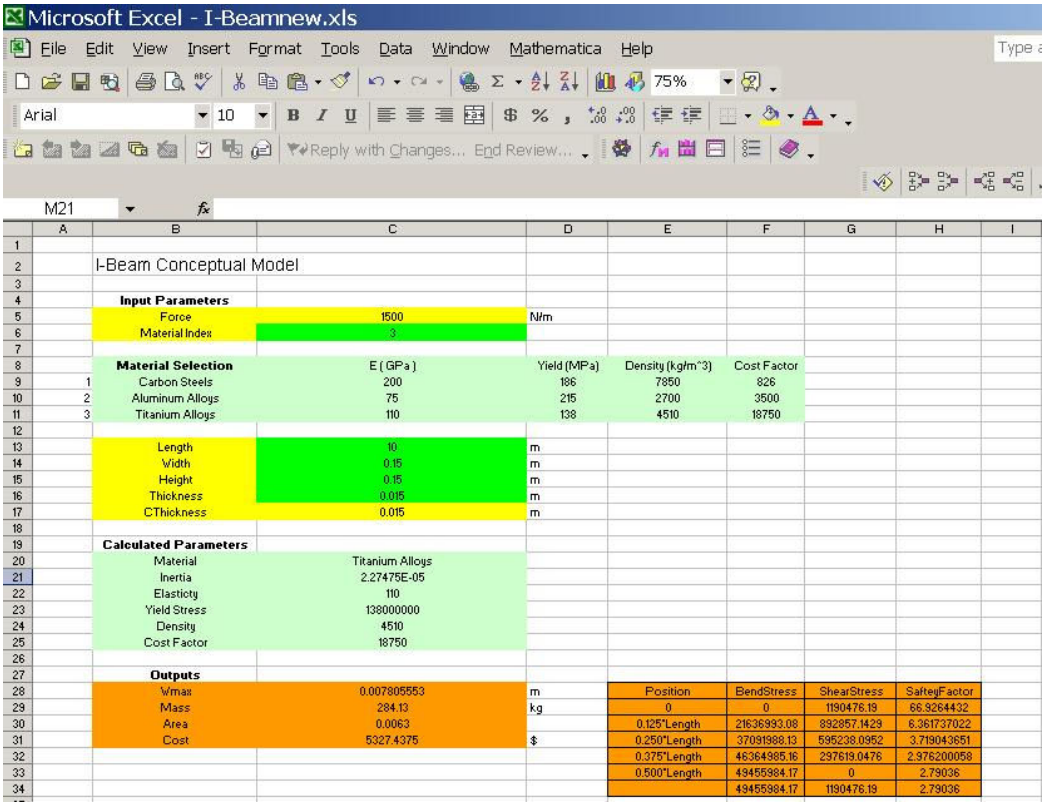


Figure 4-9: I-Beam Conceptual Design Model

$$I = \frac{wh^3 - (w - ct)(h - 2t)^3}{12} \qquad 4.2$$

$$A = 2wt + ct(h - 2t) \qquad 4.3$$

$$M(x) = \frac{F\,L\,x}{2} - \frac{Fx^2}{2} \qquad 4.4$$

$$F_s(x) = \frac{\frac{F\,L}{2} - Fx}{A} \tag{4.5}$$

$$\sigma = \frac{Mc}{I} \tag{4.6}$$

$$V = \frac{F_s}{A} \tag{4.7}$$

$$m = AL\rho \tag{4.8}$$

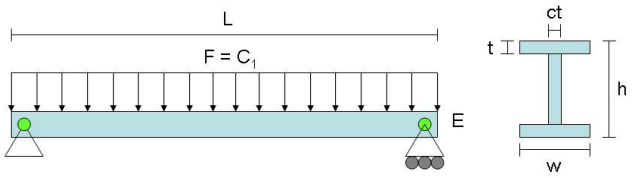
$$Cost = \beta m \tag{4.9}$$

$$SF = \frac{\sigma_{Yield}}{\sigma_{vonmises}} \tag{4.10}$$

w : width
h : height
t : thickness
I : Moment of Inertia
c: Distance from the Neutral Axis
A : Area
L : Length
F : Distributed Force

M(x) : Moment as a function of position
$F_s(x)$: Shear Force as a function of position
σ: Bending Stress
V : Shear Stress
m : Mass
β : Cost factor
SF : Safety Factor

Underlying calculations are placed into a Mathematica notebook, since Mathematica has good documentation and formatting capabilities. Using MathlinkForExcel, the Microsoft Excel front-end and Mathematica code can exchange information.



Maximum Displacement

```
wmax[force_, length_, elast_, inertia_] := 5 * force * length^3 / (384 * elast * inertia * 10^9)
```

Maximium Bending Stress

```
sigmax[force_, length_, height_, inertia_, thickness_] =
 Table[((force * length * xpos / 2 - force * xpos^2 / 2) * (height/2 - thickness) / inertia,
  {xpos, 0, length/2, length/8}]
```

Maximium Shear Stress

```
Off[NumberForm::sigz];
vmax[force_, length_, area_] :=
 Table[(force*length/2 - force * xpos) / area, {xpos, 0, length/2, length/8}]
```

Moment of Inertia

```
inertia[thickness_, width_, height_, cThickness_] =
 (width height^3 - (width - cThickness) * (height - 2 * thickness)^3) / 12
```

Mass

```
mass[thickness_, width_, height_, cThickness_, length_, density_] :=
 (2 * thickness * width + cThickness * (height - 2 * thickness)) *
  length * density
```

Stress Safety Factor

Use Mohr's Circle to find principle stresses
Calculate Von Mises Stress
Compare Yield Stress with Von Mises Stress

```
sf[sigMax_, shearMax_, yieldStress_] :=
  Module[{mohravg, mohrrad, sa, sb, vonmises, sf},
   mohravg = sigMax/2; mohrrad = Sqrt[mohravg^2 + shearMax^2];
   sa = mohravg + mohrrad; sb = mohravg - mohrrad;
   vonmises = Sqrt[sa^2 - sa*sb + sb^2]; sf = yieldStress/vonmises; sf];
```

Area

```
area[thickness_, width_, height_, cThickness_, length_] =
  Module[{area},
   area = 2 * thickness * width + cThickness * (height - 2 * thickness);
   area];
```

Cost

Material Factor is $ / Metric Ton
1 Metric Ton = 1000 kg

```
cost[thickness_, width_, height_, cThickness_, length_, density_,
   materialfactor_] =
  Module[{area},
   area = 2 * thickness * width + cThickness * (height - 2 * thickness);
   ibeamcost = materialfactor * area * length * density / 1000];

EndPackage[];
```

Figure 4-10: I-Beam Overview and Design Rules in Mathematica

4.2.1 I-Beam Model Development and Sampling Time Measures

The total time taken to create the I-beam conceptual model, consisting of the Microsoft Excel Worksheet and underlying Mathematica code was 2 hours and 30 minutes. The conceptual model sampled the input space at 5,000 unique locations using Crystal Ball. The conceptual model can calculate 5000 I-beam designs in 5 minutes and 30 seconds. As a result, a rapid mapping of the trade space is performed. No constraints are applied to the model.

	A	B	C	D	E	F	G
1	Dim::Length	Dim::Width	Dim::Height	Dim::Thickness	Dim::Cthickness	DisplacementMax	MaxBending
2	6.75	0.14	0.17	0.01	0.00	0.00	30,591,678.85
3	9.19	0.16	0.16	0.00	0.00	0.00	92,787,337.67
4	10.98	0.14	0.12	0.01	0.01	0.00	169,377,586.49
5	10.16	0.13	0.18	0.01	0.01	0.00	118,986,519.68
6	8.95	0.13	0.13	0.01	0.01	0.00	127,092,006.18
7	7.37	0.17	0.15	0.01	0.01	0.00	36,767,318.84
8	11.27	0.16	0.15	0.01	0.00	0.00	107,984,278.84
9	8.37	0.15	0.17	0.00	0.00	0.00	564,646,972.55
10	12.84	0.18	0.20	0.00	0.00	0.00	148,864,849.17
11	9.31	0.17	0.15	0.00	0.00	0.00	110,600,714.02
12	10.81	0.15	0.16	0.00	0.00	0.00	115,273,154.43
13	11.88	0.14	0.15	0.00	0.01	0.00	226,504,463.21
14	9.51	0.14	0.14	0.00	0.00	0.00	149,356,934.87
15	11.05	0.16	0.15	0.00	0.00	0.00	165,934,604.98
16	7.43	0.16	0.17	0.01	0.01	0.00	42,086,220.10
17	8.88	0.15	0.16	0.00	0.00	0.00	100,732,551.97
18	12.07	0.12	0.15	0.00	0.01	0.00	252,733,657.80
19	10.31	0.15	0.16	0.00	0.01	0.00	421,280,016.42
20	7.08	0.15	0.15	0.01	0.00	0.00	44,886,839.97
21	7.23	0.14	0.12	0.01	0.00	0.00	80,691,887.70
22	10.92	0.15	0.15	0.00	0.01	0.00	115,185,921.87
23	12.66	0.13	0.13	0.01	0.00	0.00	177,282,434.78
24	12.46	0.15	0.16	0.01	0.01	0.00	139,558,998.66
25	9.38	0.16	0.15	0.01	0.00	0.00	68,916,154.40
26	8.74	0.16	0.11	0.01	0.00	0.00	80,093,268.23
27	9.82	0.16	0.15	0.01	0.01	0.00	95,223,348.97
28	11.47	0.17	0.12	0.00	0.00	0.00	124,515,888.07
29	10.08	0.12	0.15	0.00	0.00	0.00	165,710,975.56
30	9.52	0.17	0.16	0.00	0.00	0.00	86,519,123.92
31	9.50	0.16	0.15	0.00	0.00	0.00	93,822,654.18
32	9.25	0.17	0.15	0.00	0.00	0.00	164,321,015.79
33	13.00	0.19	0.17	0.00	0.01	0.00	130,131,152.33
34	9.61	0.15	0.18	0.00	0.00	0.00	75,080,023.20
35	7.64	0.14	0.13	0.00	0.00	0.00	126,007,928.93
36	9.72	0.15	0.12	0.00	0.01	0.00	117,571,522.67
37	9.78	0.14	0.16	0.00	0.00	0.00	155,886,454.24
38	12.97	0.14	0.15	0.01	0.01	0.00	134,206,354.01
39	10.71	0.16	0.15	0.01	0.01	0.00	86,839,347.12

Figure 4-11: I-Beam Dataset

4.2.2 2D Analysis + Color to Represent an Additional Variable of the I-Beam Trade Space

Figure 4-12 displays a 2D scatter plot using colored cubes, in which each cube represents one I-beam design. The area and maximum bending stress of an I-beam design

is represented by a cube's position along the x-axis and y-axis, respectively. The color of each glyph represents a design's safety factor, a combined measure of bending stress and shear stress with respect to the yield stress. Designs shaded dark blue exhibit low safety factors, which subject them to a greater probability of failure. The Pareto frontier within these figures indicates a preference on minimizing both area and bending stress; as a result, a tradeoff curve is distinguished using black markings.

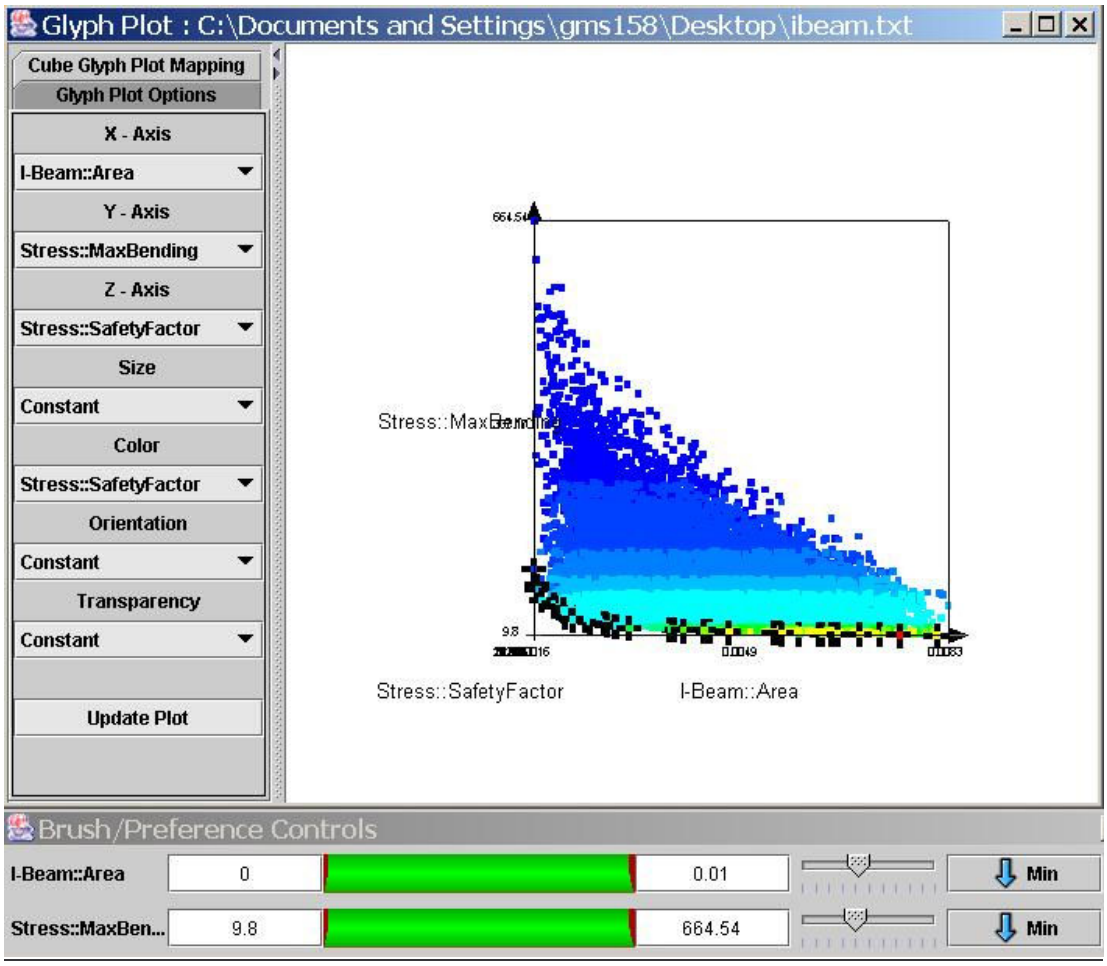


Figure 4-12: I-Beam Trade Space and Pareto frontier Display

Since no constraints are added to the conceptual model, constraints can be applied using brushing. Figure 4-13 displays I-beam designs with a safety index greater than 1, which applies a failure constraint.

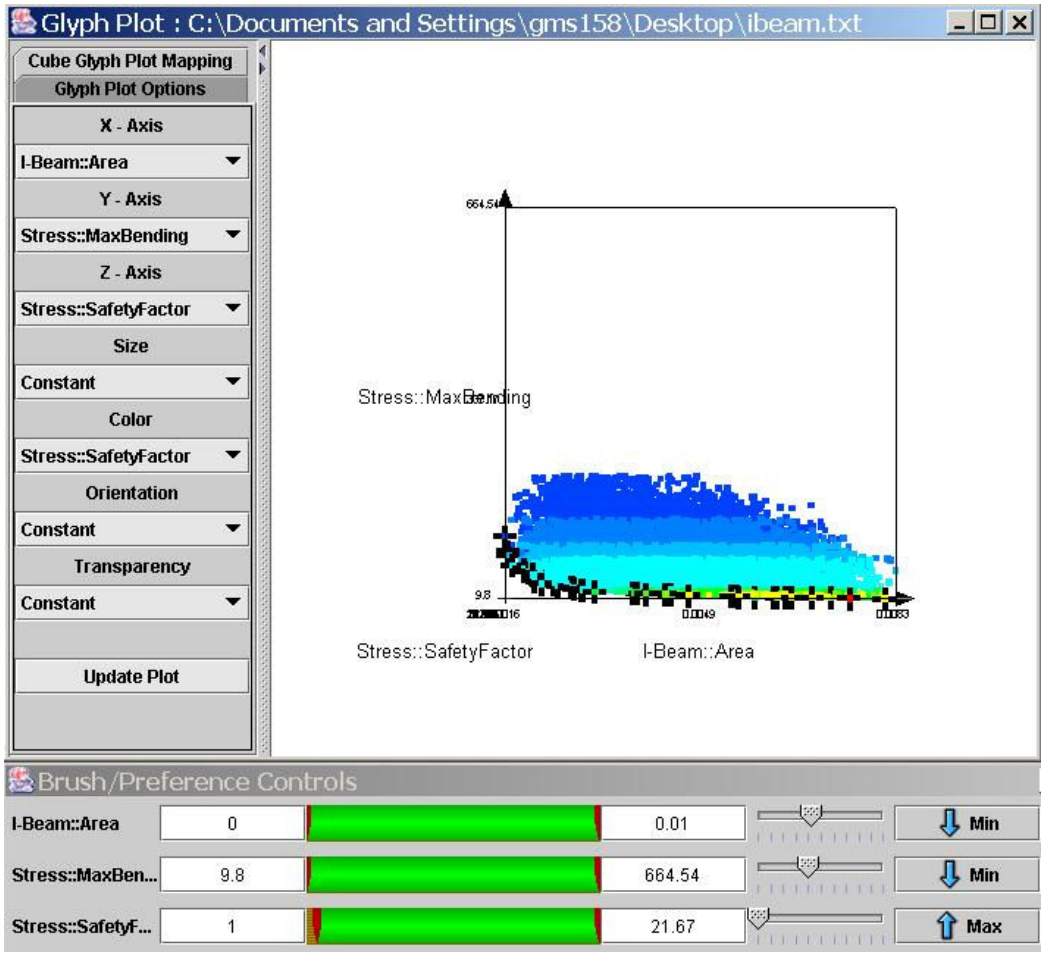


Figure 4-13: I-Beam Trade Space with a Constraint

Figure 4-14 displays preference shading by highlighting I-beam designs that rank highest according the relative weightings specified in the Brush/Preference Controls Toolbox. Preference structure direction remains the same in both figures (minimizing area and bending stress); the relative weightings between these two variables differ in the

two figures. The left glyph plot in Figure 4-14 illustrates a greater importance on minimizing bending stress, where the left glyph plot places a greater importance on minimizing area. Preference shading highlights regions in the glyph plot, allowing a decision-maker to visualize unique optimal designs that correspond to different preference structures.

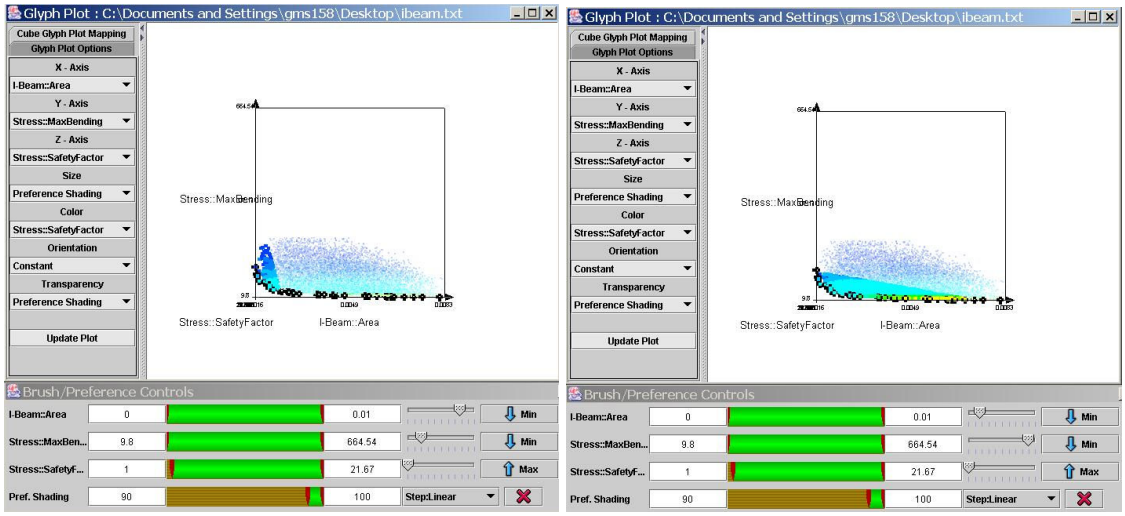


Figure 4-14: I-Beam Trade Space with Two Preference Structure

4.2.3 Multi-Dimensional Visualization of the I-Beam Trade Space

Figure 4-15 displays a 5-dimensional glyph plot where the cubes x, y, and z position represent an I-beam's length, maximum bending stress, and area, respectively. The size of the glyph represents an I-beam's shear stress, which is highly correlated to length; the color of each glyph represents a design's mass, a measure correlated with cross-sectional area, length, and material selection. Additionally, a constraint is applied to

the safety factor of each design, by only displaying I-beam's with a safety factor of 1 and greater.

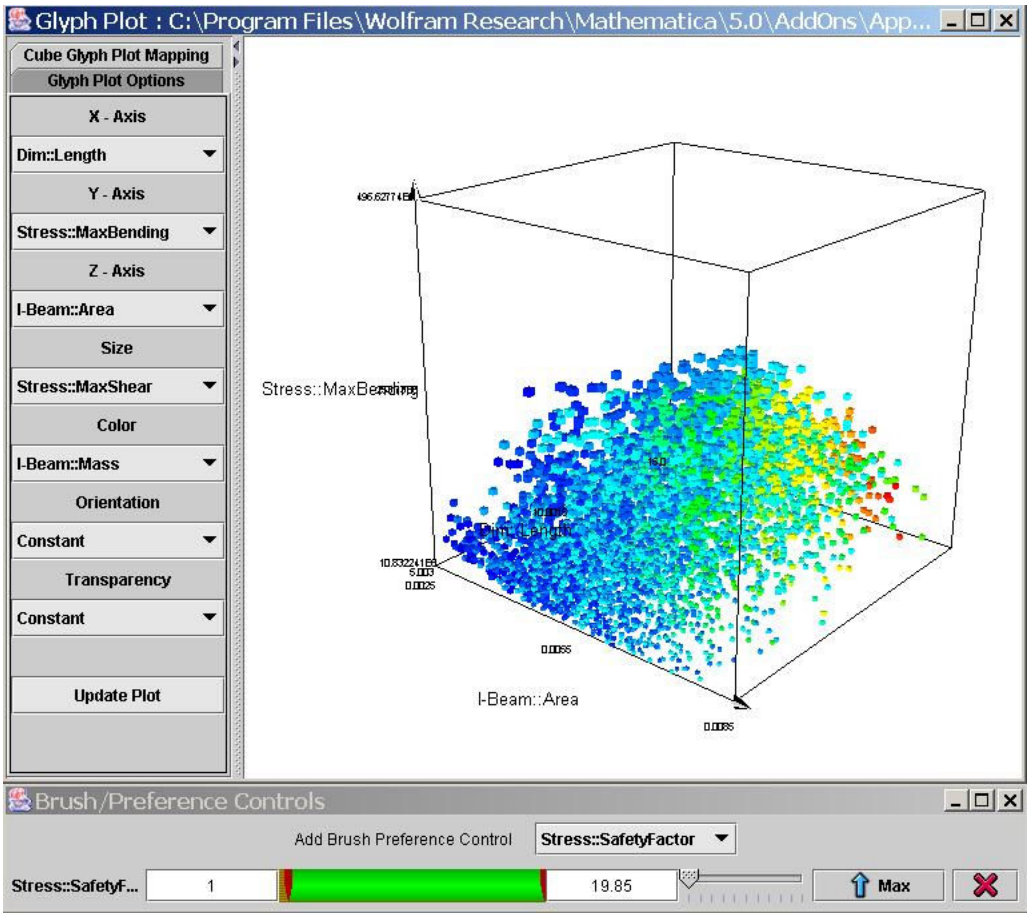


Figure 4-15: I-Beam Trade Space Using Glyph to Display 5 Dimensions

The glyph plot in Figure 4-16 displays the resulting Pareto frontier for the following preference structure:

- Maximize Length
- Minimize Maximum Bending Stress
- Minimize Cross-sectional area

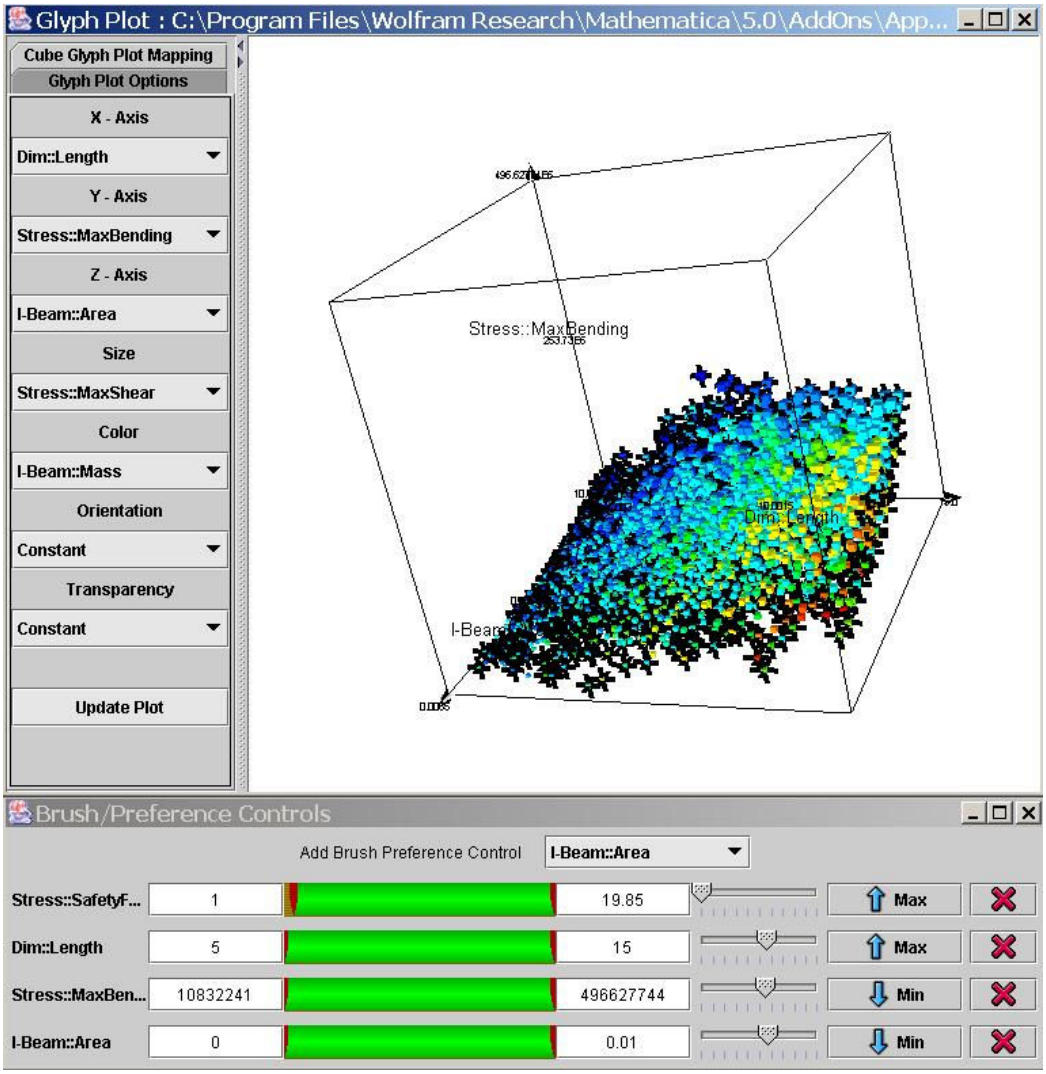


Figure 4-16: Type Caption Here

A feature within the ATSV allows a decision-maker to view only designs that lie on the Pareto frontier, and the resulting glyph plots are displayed in Figure 4-17. Preference shading is used to highlight different regions on the Pareto frontier, which highlights Pareto optimal designs that rank well with respect to different preference structures.

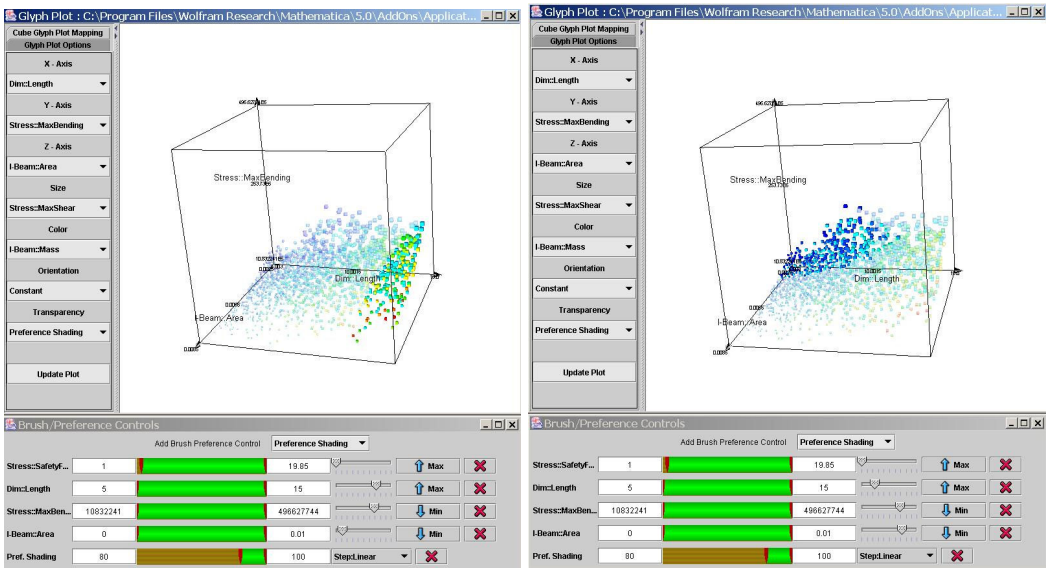


Figure 4-17: Preference Shading and Pareto Frontier Display in the I-Beam Trade Space

Two different preference structures are displayed in Figure 4-17. The first preference structure places a greater importance on maximizing length and minimizing bending stress, while the second preference structure places a greater importance of minimizing bending stress and minimizing area. Two different regions within the trade space are highlighted, illustrating that different designs rank well with different preference structures. Quantitative information windows compare different I-beam designs that are selected within these areas of interest, as shown in Figure 4-18.

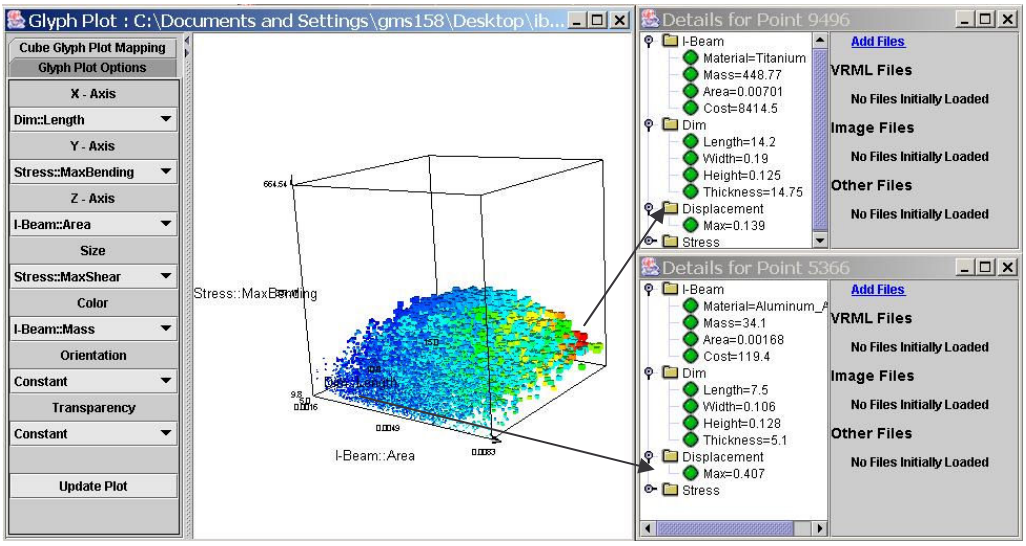


Figure 4-18: Design Selection

4.2.4 Enumerations within the I-Beam Trade Space

Figure 4-19 displays the I-beam's material, a choice of carbon steel, aluminum alloy, or titanium alloy, on the z-axis of the glyph plot. The glyph plot mapping is as follows:

- X-Axis : Maximum Bending Stress
- Y-Axis : Cost
- Z-Axis : Material Selection
- Color : Mass

Key trades are displayed within this trade space between material selection, resulting mass, and cost of an I-beam. Titanium I-beams have a high cost with respect to aluminum and carbon steel I-beam; however, the carbon steel I-beams have a very high weight with respect to the other material selections.

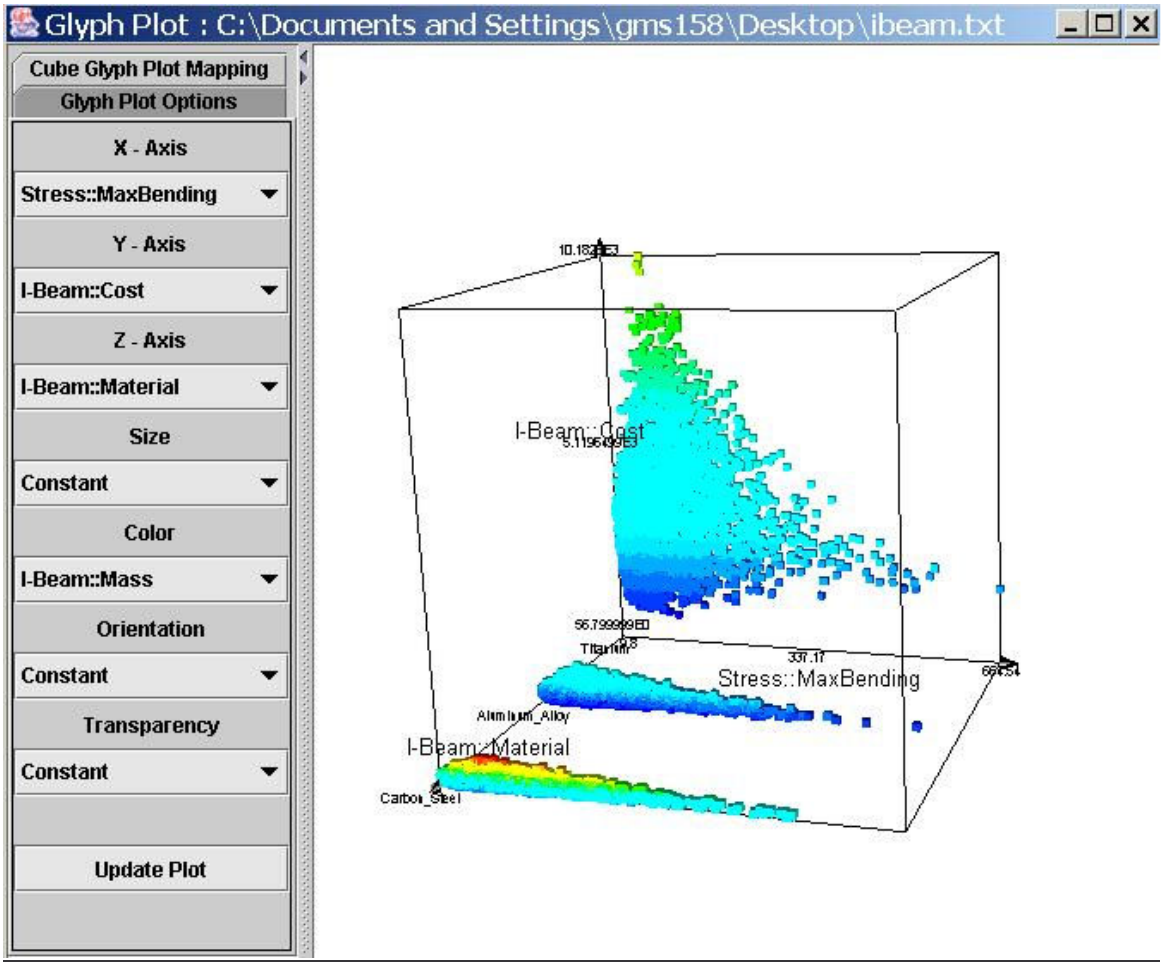


Figure 4-19: Enumerations within the I-Beam Trade Space

4.2.5 Histogram Plots of the I-Beam Trade Space

Figure 4-20 displays a histogram plot of the overall I-beam trade space. Individual histograms of input variables (material, length, width, height, thickness, and thickness), display a uniform distribution, since the input space was randomly sampled. Several

output variables (displacement, bending stress, safety factor, mass, and cost) display highly skewed distributions.

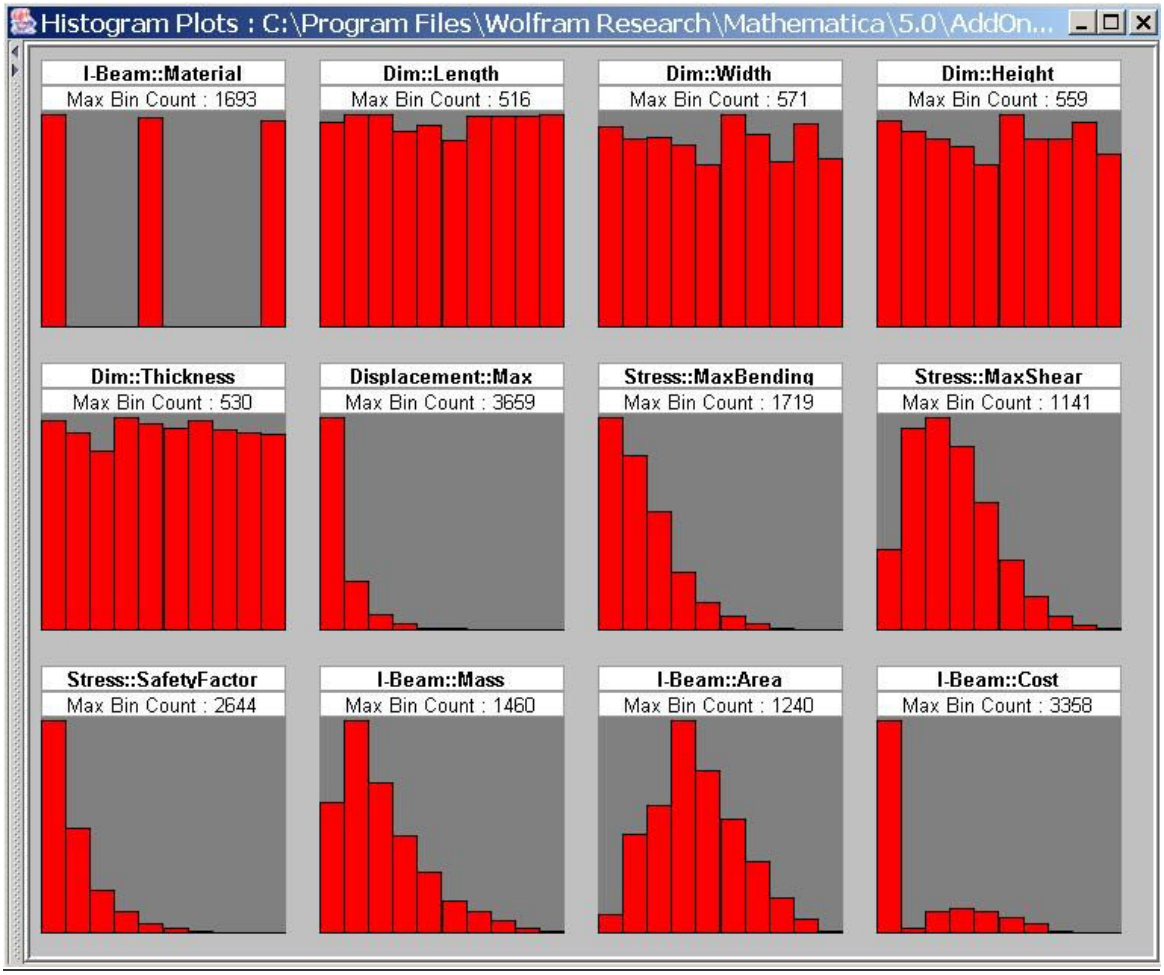


Figure 4-20: Histogram Plot of I-Beam Trade Space

Brushing of histogram plots aids the decision-maker in visualizing correlations and conditional probabilities in the design space. The overall distribution is displayed using black outline rectangles, while the brushed distribution is represented by solid red

rectangles. Figure 4-21 displays a histogram plot only displaying designs with bending stress less than 130.93 MPa.

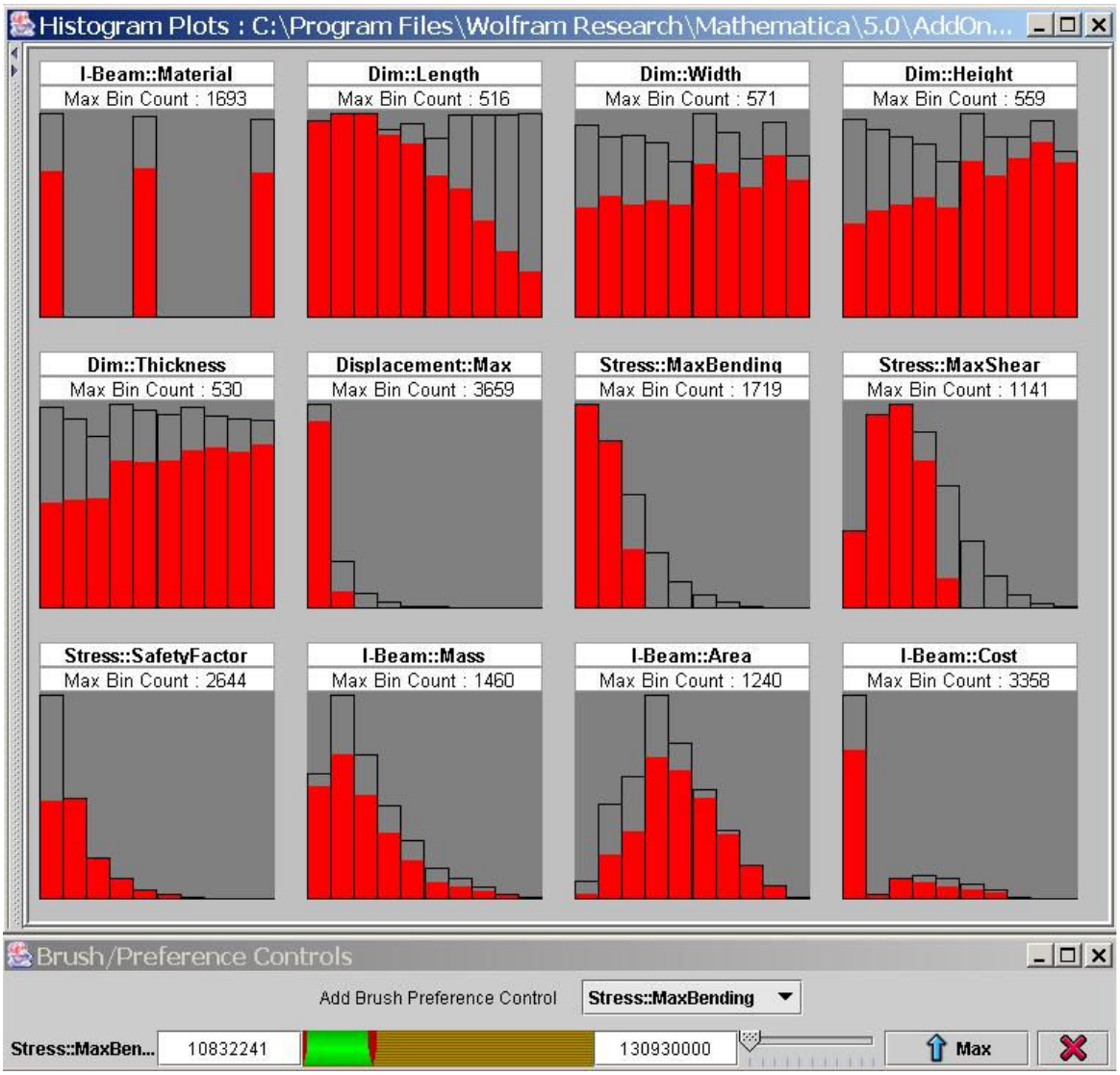


Figure 4-21: Histogram Plots with a Brushed Imposed on Maximum Bending Stress

The brushed distributions for length, maximum displacement, maximum bending stress, and shear stress are shifted towards the lower bounds of their ranges. The three

input variables—width, height, and thickness—display a non-uniform distribution, which indicates increasing these variables decreases the bending stress.

Conditional probabilities are displayed in histogram plots by visualizing the resulting brushed distributions. For example, the first bin in the length histogram plot is solid red. Given that the length of the I-beam falls within the range indicated by the first rectangle (design near the lower bound), the probability is 100% that bending stress is less than 130.93 MPa. In the length histogram plot, this probably decreases with each successive bin to the right, since each bin's solid red area decreases. Other input variables—width, height, and thickness—display the opposite trend, in that the conditional probabilities ($\sigma < 130.93$ MPa | Design lies in bin) increases as one moves toward the upper bound. This trend makes sense, since additional material will decrease the bending stress in the I-beam.

4.3 Satellite Design Example

4.3.1 Mars Orbiter Satellite

The data used to illustrate the ATSV was generated using a conceptual model, with rules based on satellite design textbooks and Mars Odyssey mission characteristics available through the open literature [45-50]. The 2001 Mars Odyssey satellite (shown in Figure 4-22) [51] is currently orbiting Mars, and its primary functions are to measure the elements and minerals on the surface of Mars, to search for the presence of water, and to measure radiation levels that would be experienced by manned-missions to Mars.

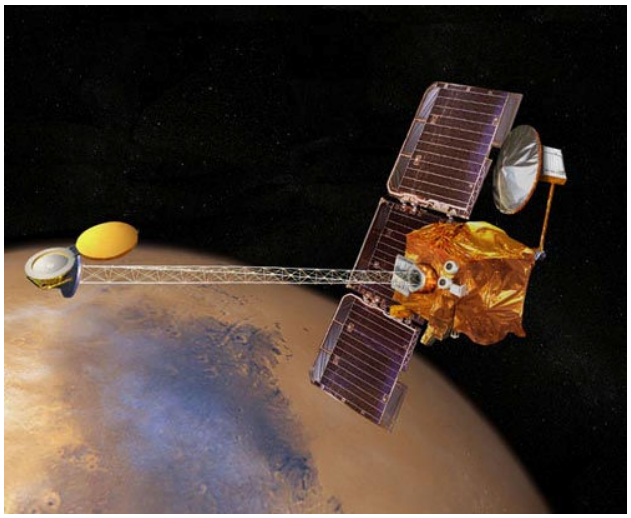


Figure 4-22: Mars Odyssey Satellite [51]

Conceptual models are used to populate the satellite trade space. Each design is calculated by randomly sampling a point in the input space; then, the conceptual model calculates a feasible satellite design starting from the initial sampled point. This process is repeated until an extensive mapping of the trade space is performed. The dataset for this example includes 6500 designs, each design having 14 variables, forming a 14-dimensional satellite trade space populated by 6500 unique satellite design configurations. The total elapsed time to generate 6500 designs was 12 hours. An example dataset is shown in Figure 4-23, where each row represents a design and each column represents a variable.

	A	B	C	D	E	F	G
1	RxnWheelIdx_Input	pellantType_In	Delta_V_OI_Input	Slew_Time	Total_Mass	Cost	Mission_Life_Input
2	6	N2O4/MMH	1337.89	100.11	2184.3	3619.74	2.45
3	6	N2O4/N2H4	1431.08	101.08	1901.37	2805.44	4.42
4	5	N2O4/N2H4	1231.7	115.04	1580.84	2203.26	3.84
5	13	N2O4/N2H4	1525.82	52.18	2463.23	4427.46	4.35
6	8	N2O4/N2H4	1308.18	121.43	1747.02	2502.43	4.84
7	13	N2O4/MMH	1568.74	56.94	3362.19	7239.67	4.28
8	1	N2O4/N2H4	1228.35	148.33	1621.95	2276.06	4.69
9	1	N2O4/MMH	1258.26	147.08	1960.26	3121.14	2.97
10	5	N2O4/N2H4	1204.53	123.27	1647.65	2285.03	4.27
11	12	N2O4/MMH	1107.46	33.1	2188.34	3411.6	4.23
12	12	N2O4/MMH	1522.56	33.54	2725.84	5401.61	2.82
13	7	N2O4/MMH	1300.09	109.24	2144.27	3526.75	2.67
14	3	N2O4/N2H4	1216.17	96.47	1573.69	2152.97	2.47
15	11	N2O4/N2H4	1253.53	107.68	1718.3	2416.08	4.48
16	3	N2O4/MMH	1282.06	105.02	2110.45	3445.5	2.45
17	10	N2O4/N2H4	1409.64	71.65	1812.05	2629.02	2.99
18	1	N2O4/N2H4	1430.07	143.53	1695.74	2483.71	3.02
19	3	N2O4/N2H4	1377.19	103.83	1808.94	2622.68	4.03
20	14	N2O4/N2H4	1135.46	104.59	2195.38	2860.59	3.93
21	1	N2O4/MMH	1193.93	157.96	2046.08	3275.72	4.64
22	13	N2O4/N2H4	1261.49	48.54	2063.7	2860.97	4.77
23	9	N2O4/N2H4	1371.16	172.21	1734.28	2495.78	3.14
24	13	N2O4/N2H4	1542.57	52.33	2498.76	4608.05	4.68
25	6	N2O4/MMH	1437.13	102.38	2359.83	4147.46	3.18
26	9	N2O4/N2H4	1532.99	176.21	1865.81	2779.12	2.04

Figure 4-23: Satellite Dataset Example

4.3.2 Trade Space Exploration of a Satellite Data Set

The primary design variables for this example are as follows:

- RxnWheelIdx: The index into the catalog of reaction wheels, of which there are 14 choices
- Propellant Type: Type of bipropellant, of which there are 7 choices (LOX/N_2H_4, LOX/RP1, LOX/UDMH, H_2O_2/N_2H_4, N_2O_4/N_2H_4, N_2O_4/50%UDMH50%N_2H_4, N_2O_4/MMH)
- ΔV: The change in velocity available over the mission
- Slew Time: The time for the vehicle to traverse a designated slew angle
- Mass: The mass of the vehicle with propellant
- Cost: The cost of the vehicle
- Mission Life: The overall duration of the mission
- Fuel : The fuel mass
- Prop_Mass: the propellant mass
- Dry_Mass: The mass of the vehicle without propellant
- RW_Mass : The reaction wheel mass

The key trade in this satellite design example is to see how the choice of reaction wheel and propellant type affects the slew time, cost, and overall mass characteristics of

a satellite. The glyph plot, shown in Figure 4-24, displays a satellite trade space of interest in the preliminary design stage. The glyph plot mapping is as follows:

- X – axis : Propellant Type
- Y – axis : Cost
- Z – axis : Reaction Wheel Index (sorted according to the Reaction Wheel Mass)
- Glyph size : Dry Mass
- Glyph color : ΔV

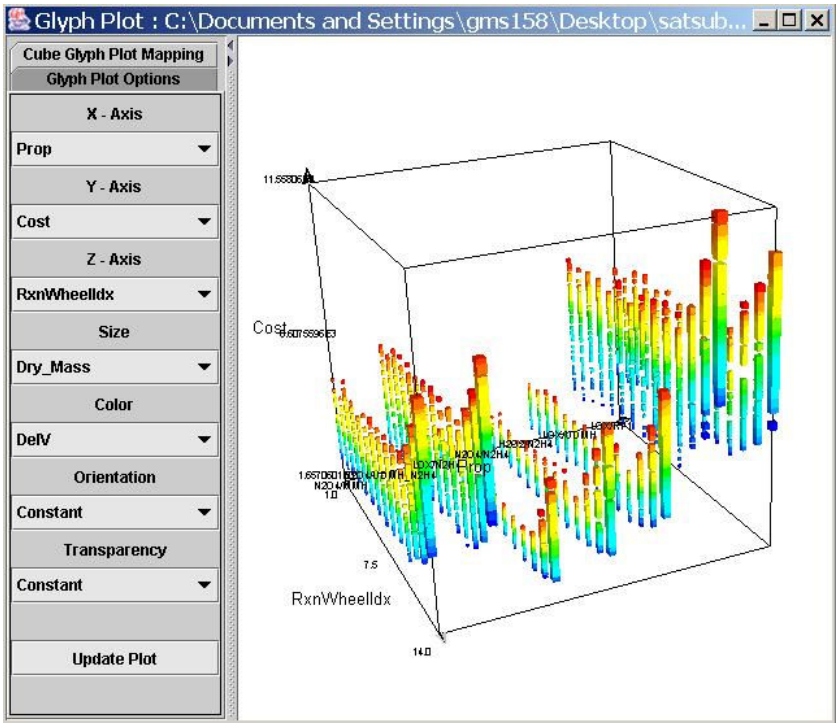


Figure 4-24: Glyph Plot Displaying Satellite Trade Space

This glyph plot illustrates that the choice of both propellant and reaction wheel choice affects the cost, ΔV, and dry mass of the satellite. Satellite designs having N_2O_4 as the fuel choice (middle 3 values along the X-axis) perform well with respect to cost, since these satellites have a lower cost compared to the other four propellant choices. The

glyph plot shows that, for a given cost, satellites with N_2O_4 as the fuel choice achieve a higher ΔV (color) with respect to other propellant choices. Additionally, the dry mass (size) of the satellite increases as both ΔV and RW Index increase. Dry mass increases with ΔV, since additional propellant is needed; the volume of the propellant tanks increases to accommodate the additional fuel requirements, which drives the weight of the propellant tanks. Also, RW Index Mass increases from index 1 to 14, respectively, which causes the dry mass to increase as RW Index increases.

Figure 4-25 displays the range of slew times that individual reaction wheels can achieve. Lower Slew Times are more preferred, since less time is needed to slew between different pointing orientations. Figure 4-25 displays that reaction wheel index 12 performs the best with respect to minimizing Slew Time; additionally, RW Indices 10, 11, 13, and 14 also perform well with respect to Slew Time.

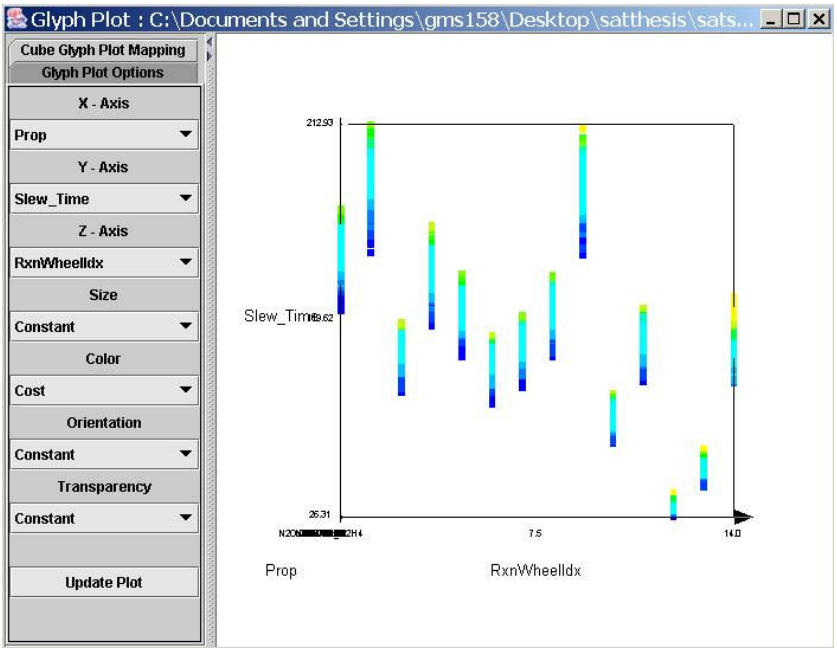


Figure 4-25: Slew Time vs. Reaction Wheel Index

A preference structure in this satellite design problem is as follows:

- Minimize Cost
- Minimize Slew Time
- Maximize ΔV

Figure 4-26 illustrates how the ATSV is used to shop for the most preferred design. Tradeoffs exist between Cost, Slew Time, and ΔV, and different relative weightings between these variable will lead to different preferred designs.

Only showing designs with N_2O_4 as the fuel choice, Figure 4-26 displays the following four preference structures:

1. Greater Importance on minimizing Cost
2. Greater Importance on minimizing Slew Time
3. Greater Importance on maximizing ΔV
4. Equal Importance Placed on All Variables

More preferred designs are red, and less preferred designs are blue and black in glyph and parallel coordinate plots, respectively.

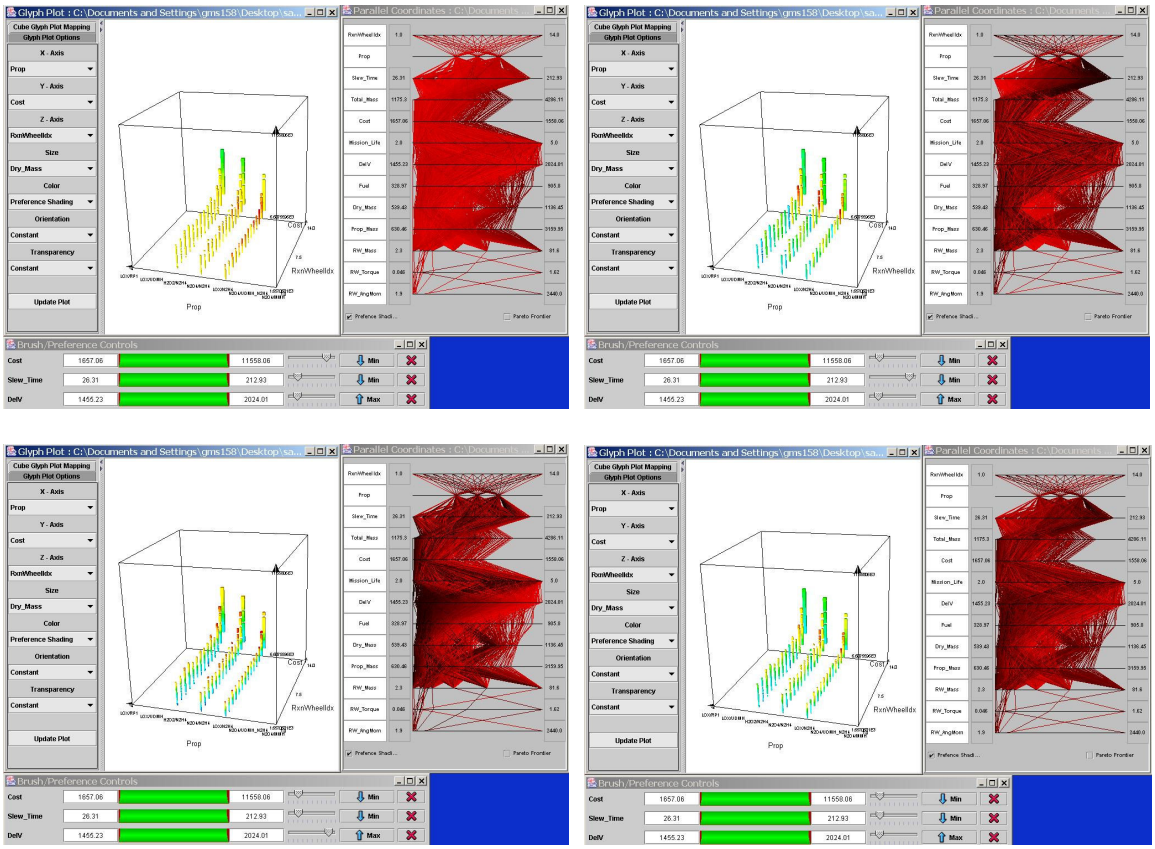


Figure 4-26: Different Relative Weighting between Cost, Slew Time, and ΔV

Figure 4-26 illustrates that different preference structures highlight different designs in the trade space. The first plot highlights a large majority of satellites, since all designs, except ones with higher reaction wheel indices, perform very well with respect to this preference structure. The second plot highlights designs within a middle range of Dry Mass and Reaction Wheel Mass; additionally, satellites with RW Index 12 are highlighted, which corresponds to data displayed in Figure 4-25. Better performance in slew time will increase the Dry Mass of the satellite, since the reaction wheels needed for low Slew Time have a large mass. Additionally, the Cost of the satellite will increase

with respect to higher RW Index choices. The third figure places a greater preference on maximizing ΔV, which highlights designs that have high Cost and more Fuel Mass. All RW Index and propellant type combinations have satellites capable of high ΔV; however, Cost differs between these choices of combinations. The final figure places an equal importance on all variables in the preference structure, and designs with RW Index 12 and high ΔV value are highlighted. Also, these highlighted designs have a low Cost with respect to all designs in the trade space. The parallel coordinates plot for this preference structure highlights designs with low Slew Time, high ΔV, high Fuel Mass, a middle range of Dry Mass, and a middle range of Reaction Wheel Mass.

Illustrated in Figure 4-27, decision-makers can display additional information by selecting a design in the glyph plot. New windows with quantitative information and links to 3D geometries, images, and other files associated with designs are displayed to the decision-maker.

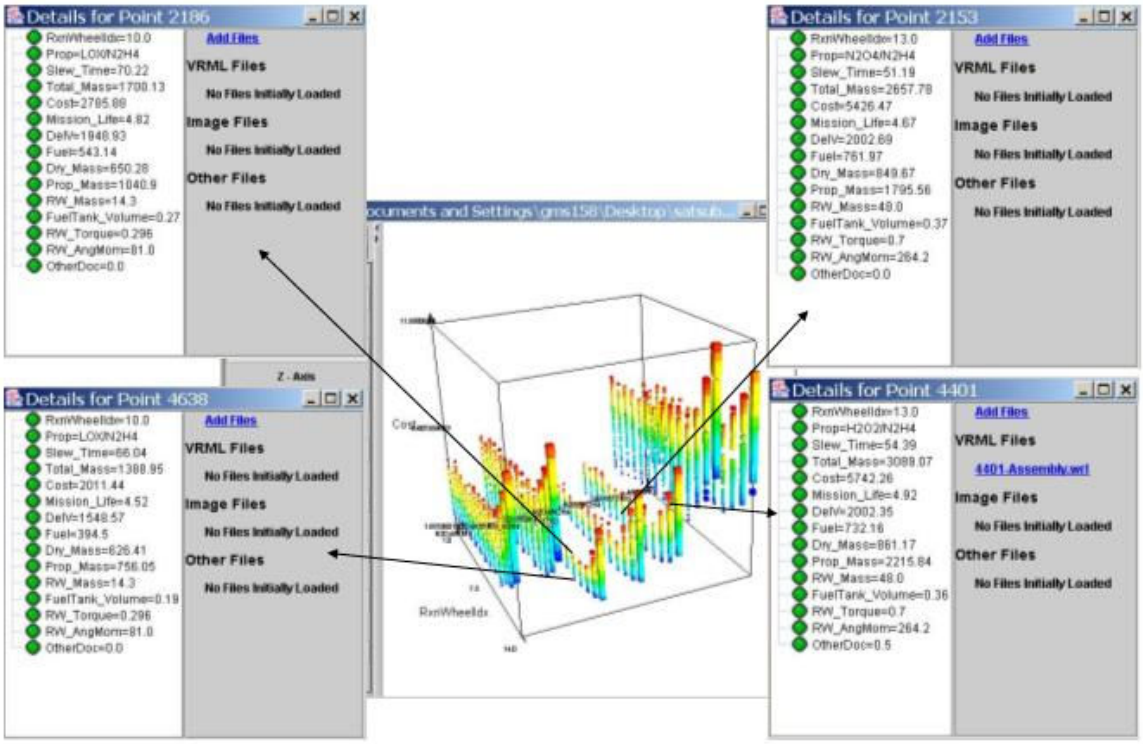


Figure 4-27: Quantitative Information Display of Selected Satellites

The 3D geometry display illustrates addition information to the decision-maker, such as position, size, and orientations of components within a virtual prototype. For example, two solid models are displayed in Figure 4-28, where the platform locations of the main satellite structure are driven by the individual components of the satellite. The vertical position of the middle platform is driven by the volume needed to accommodate the fuel and pressurant tanks, yellow and blue tanks, respectively; propellant choices with high densities will place the middle platform close to the lower platform, since less volume is needed. Additionally, the location of the upper platform is driven by the size of the reaction wheels and oxidizer tank. The reaction wheels are located underneath the

third platform (shaded in grey), and the oxidizer tank (also shaded in grey) is located in the center of the satellite structure, which is only visible in the first 3D geometry in Figure 4-28. The second satellite has larger reaction wheels compared to the first satellite, which indicates better Slew Time performance; as discussed in the aforementioned section, designs with good Slew Time performance will have high Cost, Dry Mass, and Reaction Wheel Mass.

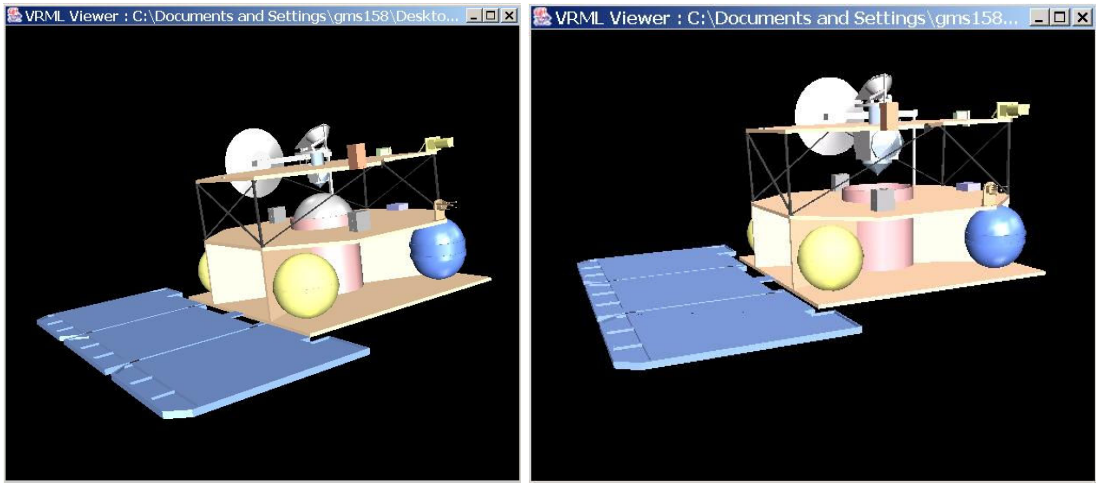


Figure 4-28: 3D Geometry File Comparison

The ATSV allows decision-makers to visualize different preference structures by highlighting different regions within the trade space. Preference shading is used to aid decision-makers in understanding tradeoffs between variables, and Pareto optimality will draw attention to specific satellites within these highlighted regions. Different relative weighting between design variables highlight regions within the trade space, aiding decision-makers in selecting the most preferred design. The shopping process is

summarized in Figure 4-29, where different preference structures will lead to different preferred designs.

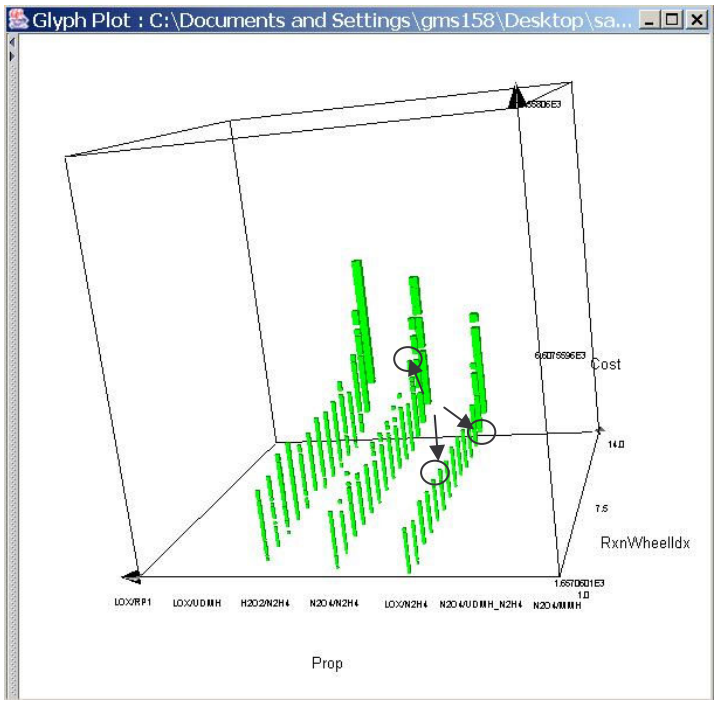


Figure 4-29: Design Selections Based on Formed Preference Structures

4.4 ATSV - General Procedure

The ATSV uses several different visualization techniques to search trade spaces. With many options available to the decision-maker, a general procedure to use this software is proposed. Design sessions using the ATSV could vary from case to case, but general guidelines for its use are as follows:

1. Use parallel coordinates plot, histogram plots, and brushing to find variables of interest within the trade space.

2. Display these variables of interest using one or more glyph plots to display different mappings of these variables.
3. Reduce the dataset by removing designs that do not satisfy a decision-maker's requirements using brushing.
4. Visualize different preference structures and Pareto frontiers using preference shading and Pareto frontier display.
5. Select a preferred design.

The first step in the procedure is to find variables of interest in the trade space by brushing parallel coordinate and histogram plots. These two visualization techniques can display a large number of dimensions to the decision-maker and are effective ways to visualize general trends in the trade space. By brushing variables, the resulting histogram and parallel coordinate plots will display which variables are possibly correlated to the brushed variable. This process allows a decision-maker to start searching for trends within the dataset.

The second step is to plot these variables of interest within one or more glyph plots. This multi-dimensional visualization technique seems to be most identifiable with most decision-makers since it is similar to a 3D scatter plot with icons that change their physical characteristics. Different mappings of the trade space result in different views of the data, and the decision-maker can explore which mappings are most beneficial.

The third step is to reduce the dataset in the display by removing designs that do not satisfy requirements or perform as well as other designs. Using brushing, a decision-maker can specify limits to the trade space, which will reduce the display of the dataset

to only designs that satisfy all requirements. This step reduces a large dataset to a subset of designs that are of interest to the decision-maker.

The fourth and fifth steps are to use preference shading and Pareto frontier display to visualize different preference structures and select a preferred design. Changing the weighting vector between performance variables will highlight different designs in the trade space, allowing the decision maker to explore tradeoff and sensitivity information between different preference structures.

4.5 Summary

The capabilities of the ATSV have been introduced and demonstrated using two design examples. The I-beam design example varied dimensions and material selection of an I-beam, and resulting deformation and stress values are calculated based on the randomly sampled point in the design space. Then, the ATSV illustrated a satellite data set which consisted of 6500 designs, each having 14 performance variables. Trade studies between the mass, cost, and slew properties of a satellite are visualized. The next chapter summarizes the contributions of the I-beam experiment, discussed in Chapter 3, and the ATSV.

Chapter 5

Conclusions and Future Work

Two different graphical user interfaces, the ATSV (Advanced Trade Space Visualizer) and an I-beam design environment, have been developed to aid decision-makers in multi-criteria design optimization. An I-beam design experiment analyzed the effectiveness, efficiency, and satisfaction of using graphical user interfaces (GUIs) in the decision-making process. This interface supplied an *a priori* specification of preference to the decision-makers, and the decision-maker varied I-beam dimensions (width and height), to find the optimal design in the trade space. This work is extended to the multi-dimensional case using the Advanced Trade Space Visualizer (ATSV). The ATSV was developed to support trade studies in the preliminary design stage, where overall trade studies between design variables are important. When faced with a multi-criteria optimization problem, decision-makers can select from a set of non-dominated or Pareto optimal designs. However, different relative weightings between variables in the trade space can lead to different preferred designs on the resulting Pareto frontier.

The development of the ATSV has offered a new procedure for searching and finding preferred designs within a trade space. A general procedure has been proposed that involves first finding variables of interest, reducing the dataset to a subset of interesting designs, and visualizing different preference structures. All visualization techniques within the ATSV contribute to the overall method of selecting a preferred design.

A limitation of this work includes no assessment of the effectiveness, efficiency, and satisfaction of the decision-making process using the ATSV. A graphical user interface has been proposed, but no heuristics support its measures of effectiveness, efficiency, and satisfaction in finding the most preferred design.

Future work may focus on testing the effectiveness, efficiency, and satisfaction of the decision-making process using the ATSV. Additional experiments that compare and contrast different multi-dimensional visualization techniques may offer insight into the advantages and disadvantages of different visualization techniques. Since the ATSV can utilize advanced virtual environments, future experiments can compare the effectiveness, efficiency, and satisfaction of the decision-making process when using desktop stereoscopic visualization as compared to stereoscopic visualization on projection screens. One can test the ATSV in the following three environments: desktop monitor, projection screens, and an immersive four-sided wall environment. Additional work on algorithms will improve the efficiency and modularity of the ATSV program architecture; and program development will incorporate additional features into the interface. Also, the ATSV can effectively display datasets of up to 50 variables, however, as the trade space dimensionality increases, new tools and algorithms to find interesting variables within the trade space can be incorporated into the ATSV.

Bibliography

1 Hwang, C.-L. and Masud, A. S., 1979, *Multiple Objective Decision Making - Methods and Applications*, Lecture Notes in Economics and Mathematical Systems, Springer-Verlag, New York.

2 Balling, R., 1999, "Design by Shopping: A New Paradigm," *Proceedings of the Third World Congress of Structural and Multidisciplinary Optimization (WCMSO-3)*, Buffalo, NY. pp. 295-297.

3 From IMTR Modeling & Simulation "Fanout Review" Draft, 11 Dec 98

4 Ng, W. Y., 1991, "Generalized Computer-Aided Design System: A Multiobjective Approach," *Computer-Aided Design,* Vol. 23, No. 8, pp. 548-553.

5 Jones, C. V., 1994, "Visualization and Optimization," *ORSA Journal of Computing,* Vol. 6, No. 3, pp. 221-257.

6 Eddy, W. F. and Mockus, A., 1995, "Dynamic Visualization in Modeling and Optimization of Ill-Defined Problems: Case Studies and Generalizations," *Technical Report,* Department of Statistics, Carnegie Mellon University, Pittsburgh, PA.

7 Head, A. J., 1999, *Design Wise: A Guide for Evaluating the Interface design of Information Resources,* Information Today, pp. 63–64.

8 Benyon, D. and Palanque, P., 1996, *Critical Issues in User Interface Systems Engineering*, Springer, New York.

9 Kraus, M. and Ertl, T., 2001, "Interactive Data Exploration with Customized Glyphs," Visualization and Interactive Systems Group, University of Stuttgart, pp. 20-23.

10 Cleveland, W. S. and McGill R., 1984, "Graphical Perception: Theory, Experimentation, and Application to the Development of Graphical Methods", *Journal of the American Statistical Association*, Vol. 79, No. 387, pp. 531-554.

11 Cleveland, W. S. and McGill, R., 1985, "Graphical Perception and Graphical Methods for Analyzing Scientific Data", *Science*, New Series, Vol. 229, No. 4716, pp. 828-833.

12 Nowell L., Schulman R., and Hix D., 2002, "Graphical Encoding for Information Visualization: An Empirical Study", *IEEE Symposium on Information Visualization,* IEEE Computer Society, October 29-30.

13 Spence, R., 2001, *Information Visualization*, Addison-Wesley ACM Press, New York, p. 69

14 Inselberg, A. and Dimsdale, B., 1990, "Parallel Coordinates: A Tool For Visualizing Multi-Dimensional Geometry", *Proceedings of IEEE Visualization '90*, IEEE, October, pp. 361-378.

15 Reed D. A., Shields K. A., Scullin W. H., Tavare L. F., Elford C. L., 1994, "Experimental Analysis of Parallel Systems: Techniques and Open Problems," *Proceedings of the 7th International Conference on Modeling Techniques and Tools for Computer Performance Evaluation*, (*invited paper*), Vienna, Austria, pp. 25-51.

16 http://www-pablo.cs.uiuc.edu/images/Projects/VR/Avatar/exterior.gif

17 Becker, R. A. and Cleveland, W. S., 1987, “Brushing scatterplots,” *Technometrics*, Vol. 29, pp. 127-142, reprinted in *Dynamic Graphics for Statistics*, W. S. Cleveland and M. E. McGill, Eds. Chapman and Hall, New York

18 Buja, A., McDonald, J. A., Michalak, J., and Stuetzle,. W., 1991, “Interactive Data Visualization Using Focusing and Linking”, *Proceedings of IEEE Visualization ’91*, IEEE Computer Society Press, October, pp. 156-163.

19 LeBlanc, J., Ward, M.O., and Wittels, N., 1990,”Exploring N-dimensional databases”, *Proceedings of the First IEEE Conference on Visualization '90*, pp. 230-237.

20 Feiner, S. and Beshers, C., 1990, “Worlds Within Worlds”, *ACM Symposium on User Interface Software and Technology*, Snowbird, UT, pp. 76–83.

21 http://www.cs.columbia.edu/graphics/projects/AutoVisual/AutoVisual.html

22 Friedman, J. H and Tukey, J. W., 1974, “A Projection Pursuit Algorithm for Exploratory Data Analysis”, *IEEE Transaction on Computing C*, Vol. 23, pp. 881-889.

23 Friedman, J.H., 1987, “Exploratory Projection Pursuit” *Journal of American Statistical Association*, Vol. 82, pp. 249-266.

24 Cook, D.R, Buja, A., Cabtea, J., and Hurley, H., 1995, "Grand Tour and Projection Pursuit", *Journal of Computational and Graphical Statistics*, Vol. 23, pp. 225-250.

25 Becker R. A. and Cleveland W. S., 1996, *Trellis Graphics User’s Manual*, MathSoft, Inc.

26 Nagel, H. R., Granum, E., and Musaeus, P., 2001, "Methods for Visual Mining of Data in Virtual Reality", *Proceedings of the International Workshop on Visual Data Mining in conjunction with ECML/PKDD2001*, 2nd European Conference on Machine Learning and 5th European Conference on Principles and Practice of Knowledge Discovery in Databases, Freiburg, Germany, September

27 Lewis, K., 2003, "Visual Design Steering as a Decision Support Aid in Design and Rapid Virtual Prototyping," *NSF Design, Service and Manufacturing Grantees and Research Conference Proceedings*, Birmingham, AL, pp. 246-264.

28 Ribarsky, W., Ayers, E., Eble, J., and Mukherjea S., 1994, "Glyphmaker : Creating Customized Visualizations of Complex Data," *IEEE Computer*, Vol. 27, No. 7, pp. 57–64.

29 Tweedie, L., Spence, R., Dawkes, H., and Su, H., 1995, "The Influence Explorer", In *Proceedings of ACM CHI 95 Conference on Human Factors in Computing Systems*, ACM Press, Denver, CO, pp. 129–130.

30 http://miner3d.com/

31 http://stats.math.uni-augsburg.de:16080/mondrian/

32 http://www.partek.com/

33 *Spotfire*, Brochure from Spotfire, Sommerville, MA, 2001

34 http://www.spotfire.com/products/decision.asp

35 Telylingen, R. V., Ribarsky, W., and Mast, C. V. D., 1997, "Virtual Data Visualizer*," IEEE Transactions on Visualization and Computer Graphics*, Vol. 3, No. 1, pp. 65-74.

36 Keim, D. and Kriegel H.-P., 1994, “VisDB: Database Exploration Using Multidimensional Visualization*,” IEEE Computer Graphics and Applications*, Vol. 14, No. 5, pp. 40-49.

37 http://213.92.76.50/

38 Ward., M., 1994, “Xmdvtool: Integrating Multiple Methods for Visualizing Multivariate Data,” *In IEEE Proc. of Visualization*, Washington, D.C., pp. 326-33

39 Swayne, D. F., Cook, D., and Buja, A., 1998, “XGobi: Interactive Dynamic Data Visualization in the X Window System*,” Journal of Computational and Graphical Statistics*, Vol. 7, No. 1, pp. 113-130.

40 http://www.arl.psu.edu/capabilities/ins_sealab.html

41 Haftka, R. and Gürdal, Z., 1992, *Elements of Structural Optimization*, 3rd Revised and Expanded Edition, Kluwer Academic Publishers.

42 Simpson, T. W., Peplinski, J., Koch, P. N. and Allen, J. K., "Metamodels for Computer-Based Engineering Design: Survey and Recommendations," *Engineering with Computers,* 2001, Vol. 17, No. 2, pp. 129-150.

43 Barton, R. R., 1998, "Simulation Metamodels," *Proceedings of the 1998 Winter Simulation Conference (WSC'98)* (Medeiros, D. J., et al., eds.), Washington, DC, IEEE, December 13-16, pp. 167-174.

44 Schroeder, W. J., Ed., 2001, *The Visualization Toolkit User's Guide*, Kitware, Inc.

45 Brown, C. D., 1998*, Spacecraft Mission Design,* Second Edition, American Institute of Aeronautics and Astronautics, Reston, VA.

46 Brown, C. D., 2000, *Elements of Spacecraft Design*, American Institute of Aeronautics and Astronautics, Reston, VA.

47 Larson, W. J. and Wertz, J. R., 1992, *Space Mission Analysis and Design*, W. J. Larson and Microcosm, Inc., Torrance, CA.

48 Sutton, George P. and Biblarz, O, 2001, *Rocket Propulsion Elements,* 7th Edition, John Wiley & Sons, New York.

49 Wertz, J. R. and Larson, W. J., 1996, *Reducing Space Mission Cost*, W. J. Larson and Microcosm, Inc., Torrance, CA.

50 http://content.honeywell.com/dses/products/ pointing/wheel.htm

51 http://mars.jpl.nasa.gov/odyssey/overview/mars-odyssey-image.html