Whitespace Exploration: The Next Step in Searching the Design Space

Andy Ko\textsuperscript{1}, Jason Daniel\textsuperscript{2}, William Keel\textsuperscript{2}, Alan Baines\textsuperscript{2}

\textit{Phoenix Integration, Blacksburg, VA, 24060}

Multi-objective optimization techniques that result in the generation of a Pareto frontier have allowed decision makers to investigate optimal tradeoffs between a set of designs. Ideally, the designs present in the Pareto frontier are acceptable candidates in the design process. However, what if none of the designs in the Pareto frontier meets the design requirements? Or, what if the desire is to find designs that are beyond the Pareto frontier? This unevaluated or undiscovered area beyond the Pareto frontier can be described as \textit{whitespace}. The process of whitespace exploration is the systematic process of exploring and investigating areas of interest previously not evaluated. A methodology for whitespace exploration is presented, which involves revisiting the assumptions and factors implicit in the generation of a Pareto frontier. The methodology presented is tested against two design problems to show efficacy. Additionally, a prototype whitespace exploration tool was developed to aid in the testing, development and demonstration of the process.

I. Introduction

Exploration of a system’s design space is a critical facet of engineering design that carries with it the possibility to yield great benefits. One such technique for design space exploration is multi-objective optimization, a process that seeks to find the optimal set of designs such that improvement in one objective can only come at the cost of others objectives. This set of designs is known as the Pareto frontier which the decision maker can gain better insight into the trade-offs between competing objectives for a given model. However, why is the Pareto frontier the limit to the model? What if none of the designs present in the generated Pareto frontier meet system requirements? What are the next steps in the design process that should be taken in these situations? This uncharted region beyond the Pareto frontier is known as the whitespace. In fact, any portion of the objective space that has yet to be explored can be considered whitespace. The methodologies built around the concept of exploring and understanding these regions of the design space can be classified as whitespace exploration.

The existence of the whitespace is due to assumptions made in the creation of the model and corresponding datasets. These assumptions range from general modeling decisions such as model fidelity, and the Pareto frontier search algorithm used, to specific numerical choices, such as variable bounds, design space constraints and constants fixed within the model. The focus of whitespace exploration here will consider those numerical choices, and not delve into the more general modeling decisions.

It should be noted that while it is possible to develop automated processes to explore the whitespace, the practical application of it requires a strong reliance on human knowledge and intuition, particularly when the model assumptions are challenged. Some assumptions are rightfully more rigid than others; for example, modifying the value for the speed of light is much more difficult to justify than to assume an efficiency assigned to a subsystem. It takes a human decision to distinguish the difference between the two, and decide when one can be changed while the other not. It is therefore our strong opinion that whitespace exploration should be a human interactive exercise, with the scientist or engineer making critical decisions at each step.

In this paper, we propose a methodology for whitespace exploration, specifically in its application to progressing a Pareto frontier. This methodology will systematically guide the scientist or engineer to consider the numerical assumptions made in the generation of the Pareto frontier, and suggests informed modifications to those assumptions that has the potential to move the Pareto frontier in the desired direction in the design space. This methodology will

\textsuperscript{1} Engineering Services Manager, Phoenix Integration, 1715 Pratt Dr., Suite 2000, Blacksburg, VA 24060, AIAA Senior Member.
\textsuperscript{2} Applications Engineer, Phoenix Integration, 1715 Pratt Dr., Suite 2000, Blacksburg, VA 24060.
be demonstrated on two different engineering test models to show efficacy. Finally, the prototype software tool that was developed will be presented.

II. The Whitespace Exploration Process

The whitespace exploration methodology presented in this paper makes several assumptions in order to properly scope its design and function. These assumptions should not however be applied to whitespace exploration in general and simply serve as a starting point to develop this particular approach. The assumptions made in the development of our approach are as follows

**Assumption 1** The model or analysis being considered is a black box

- We shall consider the model or analysis that is being performed to be a “black box” in the sense that no adjustments or introspection into the workings of it can be made. However, there will be inputs to the model or analysis that can be adjusted. Inputs can take the form of design variables, variable bounds, adjustment factors, and analysis options. Outputs from the black box can also be gathered and would take the form of performance parameters, constraints, or the results of intermediate calculations. Any introspection or changes as to how the internal parameters are calculated, set, transferred or adjusted cannot be made.

**Assumption 2** The methodology will only look at the progression of the Pareto frontier into the whitespace

- While whitespace exploration encompasses any activity that investigates a design space area where no previous data exists, the methodology presented in this paper will focus on a subset of that issue which is the progression of a Pareto frontier into the whitespace.

**Assumption 3** All input variables are continuous

- To simplify the development of the whitespace exploration algorithm discussed in this paper, it will be assumed that all input variables are continuous. The question of discrete variables such as integers or enumerated options in the context of whitespace exploration is certainly valid and an area deserving of further research.

The generation of a Pareto frontier starts with the determination of the design variables to be modified, the identification of the objectives to be considered, and any necessary constraints that might limit the consideration of a feasible result. These parameters are then defined and entered into a search algorithm that is tasked with interrogating the analysis model to find the set of designs that produce the best objective values. Interaction from the scientist or engineer will be utilized at the following three key points in the whitespace exploration process,

1. The value or extent of the side constraints of the design variables. This is a limitation that is imposed on the model inputs.
2. The value or extent of the constraints. This is a limitation imposed on the analysis or model responses.
3. The input values that were not considered as design variables. These are the fixed values in the model.

The flowchart in Fig. 1 describes the proposed whitespace exploration process. Each of the steps are then described in further detail
Figure 1. The whitespace exploration process Flowchart showing the inputs, outputs, and sub-processes that make up the proposed whitespace exploration methodology.
1) Generate an initial Pareto frontier. This starts from an initial design baseline that is comprised of the selection of design variables (and their associated box bounds), selected objectives, specified constraints and value limits, and fixed variable values.

2) Select a point in the whitespace that we wish to move the Pareto frontier towards. We call this the desired point.
   a. This point will be used to define the direction that we wish to direct the Pareto frontier towards
   b. This point is typically generated by system requirements or a utopia point depending on user goals in the objective space.

3) Modify the response variable constraints
   a. If possible, relax any of the active constraints to see if it progresses the Pareto frontier in the desired direction. There are various constraint relaxation methods that can be applied to do this. A recalculation of the Pareto frontier might be required especially if the constraints are relaxed to a condition where the Pareto frontier boundary is no longer sufficiently defined.
   b. If sufficient progression of the Pareto frontier is achieved, the process can be halted at this step.

4) Define a representative point on the Pareto frontier boundary that will be used for performing the next steps. We call this the selected point.
   a. The selected point will be used to help determine what variables, and to what extent they will have to be modified, in order to progress the Pareto frontier.
   b. We suggest that the selected point should be should be either the closest point to the desired point, or a point that is orthogonal to the desired point.
   c. There might be design variables with active side constraints. These are design variables whose values are at their bounds. The bounds of these variables will be candidates for consideration in the next steps.

5) Identify any input variables that were not design variables that can be modified. A technology factor that was fixed during the Pareto frontier search would be an example. These selected variables would be added to the list of candidates for consideration.

6) Perform a sensitivity study on the list of candidates for consideration identified in Steps 4c and 5. This study should look at the impact of the candidate list variables on the objectives and constraints about the selected point.

7) Based on the sensitivity study results in Step 6, determine a subset of variables with the greatest effect on the objectives, in consideration to its effects on the constraints that should be modified.

8) Determine the optimal combination of values from the identified subset of variables to move the Pareto frontier. While the main effects are generally considered in the sensitivity study in Step 6, the cumulative effect of changing the identified subset of variables to be modified (Step 7) will be taken into account here. This step is referred to as the direction search.
   a. An optimization is performed on the identified subset of variables by modifying their values about the selected point.
   b. The objective of this optimization is to align the unit vector movement of the selected point with the desired point unit vector pointing in the direction of the whitespace. In optimization terms, the objective is to maximize the dot product of these two unit vectors.

9) Determine an appropriate step size.
   a. Now that a combinations of variable values has been determined to progress the selected point in the direction of the desired point in the whitespace, we will need to determine an appropriate step size for those variables.
   b. In the current proposed methodology, this action is performed by the user in a trial-and-error fashion. A step size is picked and applied to the variables. Then a single point analysis is run from this point to determine the resultant output variables in the objective space. The process is repeated until sufficient movement into the whitespace is achieved.

10) Apply the change in variable values based on the optimization results in Step 9 and the step size in Step 10. Expand the bounds for the applicable design variables, and modify the fixed values for the other input variables. This will be considered the new design baseline for the next Pareto frontier.

11) Generate a new Pareto frontier based on the new design baseline.

12) Based on the progression of the Pareto frontier, repeat steps 3 through 11 until the Pareto frontier has been sufficiently progressed into the whitespace as determined by the user. We refer to each iteration of this process as a round.
Philosophically, one could consider the proposed methodology as a “directed” optimization of the Pareto frontier towards a goal. Its structure resembles a bi-layer optimization scheme with an outer layer determining a series of design baselines, while the inner layer resembling a gradient-based optimization scheme. While many of the steps of the inner layer can be automated, it is important that the decisions of the outer layer involve knowledgeable human interaction to make decisions on both what variables can be modified and the extent of the values it can be modified to. Neglecting this key aspect of the methodology could lead to designs that may make progress into the whitespace, but break critical assumptions resulting in nonsensical designs.

III. Case Studies

Now that the whitespace exploration process has been defined, we apply this process to two different case studies. These case studies use simplified engineering analysis models as representatives for real world design problems. The first case study is a satellite micrometeorite and orbital debris (MMOD) protection system design problem. It is also frequently referred to as the Whipple Shield, named after its inventor, Fred Whipple (Ref. 1). The design problem seeks to maximize the system reliability while minimizing the ratio of the mass to volume of the system. This produces a two dimensional Pareto frontier of possible designs. The second case study looks at a modified version of the sphere packing problem (Ref. 2) where the objective is to minimize the principal moment of inertia along each axis in three dimensional space with a constraint placed on the interference between spheres.

A. The Whipple Shield Case Study

In this case we examine the design of a simple Whipple shield which is a passive device for protecting spacecraft from micrometeorite and orbital debris (Ref. 3). The design goals of this problem are to maximize the impact reliability of the system while minimizing the density of the system, that is, the ratio of the mass to volume. The Whipple shield consist of three pieces, the bumper which the projectile first comes in contact with, the back wall, and the posts that attach the bumper to the wall. The design variables are focused on sizing the geometry of the

<table>
<thead>
<tr>
<th>Model Input Variables</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bumper Thickness cm</td>
<td>The thickness of the shield bumper material</td>
</tr>
<tr>
<td>Wall Thickness cm</td>
<td>The thickness of the shield wall material</td>
</tr>
<tr>
<td>Post Radius cm</td>
<td>The radius of each of the 4 support posts for the shield</td>
</tr>
<tr>
<td>Spacing cm</td>
<td>The spacing between the bumper and the wall of the shield</td>
</tr>
<tr>
<td>Shield Radius cm</td>
<td>The radius of the shield</td>
</tr>
<tr>
<td>Bumper Density gm/cm$^3$</td>
<td>The density of the shield bumper material</td>
</tr>
<tr>
<td>Wall Density gm/cm$^3$</td>
<td>The density of the shield wall material</td>
</tr>
<tr>
<td>Projectile Density gm/cm$^3$</td>
<td>The density of the projectile material</td>
</tr>
<tr>
<td>Impact Velocity km/s</td>
<td>The relative velocity between the shield and the projectile</td>
</tr>
<tr>
<td>Wall Yield Strength MPa</td>
<td>The yield strength of the shield wall material</td>
</tr>
<tr>
<td>Post Yield Strength MPa</td>
<td>The yield strength of the shield post material</td>
</tr>
<tr>
<td>Post Density gm/cm$^3$</td>
<td>The density of the shield post material</td>
</tr>
<tr>
<td>Max Launch Acceleration g</td>
<td>The maximum expected acceleration during launch</td>
</tr>
<tr>
<td>Max Launch Acceleration Standard Deviation g</td>
<td>The standard deviation on the maximum expected launch acceleration assuming a normal distribution</td>
</tr>
<tr>
<td>Mission Time Years</td>
<td>The total mission duration</td>
</tr>
<tr>
<td>Mass Multiplier</td>
<td>Multiplier applied to the mass of the shield to account for uncertainty, and unaccounted for hardware required to assemble and attach the shield to the spacecraft</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model Output Variables</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shield Impact Reliability</td>
<td>The reliability of the shield to protect from impacts over the duration of the mission</td>
</tr>
<tr>
<td>Shield Density gm/cm$^3$</td>
<td>The ratio of the shield mass to the shield volume</td>
</tr>
<tr>
<td>Post Reliability</td>
<td>The reliability of the shield post material</td>
</tr>
<tr>
<td>Shield Mass kg</td>
<td>The mass of the shield shield material</td>
</tr>
<tr>
<td>Shield Volume m$^3$</td>
<td>The volume of the shield shield material</td>
</tr>
</tbody>
</table>

Table 1. Description of Model Variables for the Whipple Shield Case Study.
Whipple shield and include the bumper thickness, the wall thickness, and the length and diameter of the support posts. The impact reliability for the shield is obtained by first calculating the critical diameter of the shield using the performance equations developed by Christiansen as shown in Ref. 3. This is the largest diameter of projectile that the shield can protect against. Then, a simplified probability density function for the MMOD flux environment is used to find the chance that an object with a size larger than the critical diameter will impact the shield area over the mission duration. The reliability therefore will be the chance that this event will not occur. From the shield dimensions and material properties we then calculate the mass and volume to obtain the shield density. Constraints are placed on the total shield mass and volume as well as a constraint placed on the reliability of the shield support posts to withstand loads due to the launch environment using the techniques developed by Elishakoff. A description of each of the variables available in the model is given in Table 1. The non-design input variables include material property parameters, mission requirements, and parameters that define the MMOD environment. In this case, we leave the parameters related to the MMOD environment fixed, leaving the others to be modified in the whitespace exploration process if it is determined to have a significant enough effect on the objectives.

The initial Pareto frontier is generated using the setup specified by the baseline design Table 2 and the NSGA-II multi-objective optimization algorithm (Ref. 5). Figure 2 shows all of the designs evaluated during the Pareto frontier search. In the figure we observe that the feasible portion of the design space is relatively small and bounds the Pareto frontier. No constraints were relaxed in the first round, but was in subsequent rounds. An impact reliability of 1.0 and a shield density of 0.0 was specified as the desired point. While we do not expect to reach such an impact reliability and shield density in our search, it provides us with a direction or goal to try to progress the Pareto frontier towards.

Sensitivity analysis identified that Shield Radius, Mission Time, Wall Yield Strength, Wall Density, and Mass Multiplier, all have a significant effect on the objective values and are so, are selected as the design variables for the direction search optimization. The extent of the step into the whitespace from the selected point is chosen arbitrarily. This process is then repeated for three more rounds resulting in the Pareto frontier progression shown in Figure 3. These results show that through the whitespace exploration process the Pareto frontier can be progressed into the whitespace through a systematic procedure.

### Table 2: Baseline design of the Whipple Shield Case Study.

<table>
<thead>
<tr>
<th>Objective Variables</th>
<th>Variable</th>
<th>Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Shield Impact Reliability</td>
<td>MAXIMIZE</td>
</tr>
<tr>
<td></td>
<td>Shield Density</td>
<td>MINIMIZE</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Constraint Variables</th>
<th>Variable</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Post Reliability</td>
<td>0.8500</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>Shield Mass</td>
<td>N/A</td>
<td>65.0000</td>
</tr>
<tr>
<td></td>
<td>Shield Volume</td>
<td>N/A</td>
<td>0.1700</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Variables</th>
<th>Variable</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Bumper Thickness</td>
<td>0.0010</td>
<td>0.0100</td>
</tr>
<tr>
<td></td>
<td>Wall Thickness</td>
<td>0.1000</td>
<td>0.6000</td>
</tr>
<tr>
<td></td>
<td>Post Radius</td>
<td>0.1000</td>
<td>0.3000</td>
</tr>
<tr>
<td></td>
<td>Spacing</td>
<td>5.0000</td>
<td>12.0000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Non-Design Inputs</th>
<th>Variable</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Shield Radius</td>
<td>100.0000</td>
<td>cm</td>
</tr>
<tr>
<td></td>
<td>Bumper Density</td>
<td>2.7130</td>
<td>gm/cm²</td>
</tr>
<tr>
<td></td>
<td>Wall Density</td>
<td>2.7130</td>
<td>gm/cm²</td>
</tr>
<tr>
<td></td>
<td>Projectile Density</td>
<td>4.5000</td>
<td>gm/cm²</td>
</tr>
<tr>
<td></td>
<td>Projectile Velocity</td>
<td>10.0000</td>
<td>km/s</td>
</tr>
<tr>
<td></td>
<td>Wall Yield Strength</td>
<td>250.0000</td>
<td>MPa</td>
</tr>
<tr>
<td></td>
<td>Post Yield Strength</td>
<td>100.0000</td>
<td>MPa</td>
</tr>
<tr>
<td></td>
<td>Post Density</td>
<td>3.0000</td>
<td>gm/cm²</td>
</tr>
<tr>
<td></td>
<td>Max Launch Acceleration</td>
<td>60.0000</td>
<td>g</td>
</tr>
<tr>
<td></td>
<td>Max Launch Acceleration Standard Deviation</td>
<td>12.0000</td>
<td>g</td>
</tr>
<tr>
<td></td>
<td>Mission Time</td>
<td>10.0000</td>
<td>years</td>
</tr>
<tr>
<td></td>
<td>Mass Multiplier</td>
<td>2.2500</td>
<td></td>
</tr>
</tbody>
</table>
B. The Sphere Packing Case Study

The sphere packing case study is a version of the problem described by Slone in Ref. 2. The problem statement is that given a collection of five solid spheres of different sizes and material densities, what is the optimal packing configuration such that the principle moments of inertia are minimized? Intuitively, the optimal solution would be to ‘pack’ the spheres as tightly as possible, with the ones of greatest densities being closer to the centroid of the cluster. The design variables considered for this optimization problem would be the individual locations of each of the spheres in three-dimensional space. In Cartesian coordinates, this would be represented by the X, Y and Z locations of each sphere, resulting in a total of 15 design variables. Constraints enforcing the interference between the spheres is required to ensure that there is no impingement between objects. These are summarized in Table 3.

This deceptively simple problem is an exceptionally difficult optimization challenge. This is due to the fact that as the spheres get closer together, the feasible design space becomes much smaller and isolated. Imagine a cluster of spheres packed together. To move one of the spheres (let us call this “Sphere A”) from one location to another, say to the opposite side of another sphere beside it (called “Sphere B”), Sphere A will have to move away from the centroid and around Sphere B to get there. The process of moving away will likely increase the principle moments of inertia making this an undesirable movement, even if the eventual result would be a more optimal solution.

Additionally, one would expect that the Pareto frontier for this particular problem would be symmetrical about the diagonal of the Cartesian coordinate space. This is because for a given cluster or sphere arrangement, rotating the entire configuration about the coordinate space diagonal would not change the principle moments of inertia, but merely redistributing its components in the three different coordinate directions.

As a mathematical exercise to demonstrate the efficacy of the whitespace exploration methodology presented here, the diameter and material density of each of the spheres were allowed to be considered as candidate variables for modification during the different rounds of the exploration. If no limits were imposed on the values, the expected result would be for the diameter and material density of each of the spheres to be driven to as close to zero as possible, with the locations of each of the spheres being as coincident as possible. It would also be expected that the principle

| Table 3. NSGA-II optimization setup for the Sphere Packing Case Study. |
|-----------------|------------------|------------------|
| **Objective Variables** |                     |                   |
| Variable | Goal |                     |
| Ixx      | MINIMIZE |                   |
| Iyy      | MINIMIZE |                   |
| Izz      | MINIMIZE |                   |
| **Constraint Variables** |                     |                   |
| Variable | Lower Bound | Upper Bound |
| Interference | N/A | 0.0000 |
| **Design Variables** |                     |                   |
| Variable | Lower Bound | Upper Bound |
| X Position [1-5] | -5.0000 | 5.0000 |
| Y Position [1-5] | -5.0000 | 5.0000 |
| Z Position [1-5] | -5.0000 | 5.0000 |
moments of inertias usually be more sensitive to the diameter of the spheres compared to its material density since principle moments of inertia is a function to the third power of diameter but is only linearly a function of density.

Figure 4 shows the progression of the Pareto frontier between rounds, with the desired point being zero principle moments of inertia in all three Cartesian directions. The exploration steps identified the diameters of the largest spheres as prime candidates to be modified, in line with expectations.

![Figure 4. Pareto front progression of the Sphere Packing test case](image)

IV. The Whitespace Exploration Prototype Tool

In order to fully realize the concept of whitespace exploration, a prototype tool was developed herein referred to as Whitespace Explorer. The tool was developed with two goals in mind, first to better understand the whitespace exploration algorithm and provide a platform for its further development, and second, to explore the aspects of human interaction present in the methodology. Whitespace Explorer guides users through each step of the whitespace exploration process in an integrated framework which automatically passes relevant information from one step to the next.

The tool is built to leverage features in a Phoenix Integration product called ModelCenter® Cloud. ModelCenter® Cloud allows users to develop ModelCenter® workflows and publish it to a central server. Other users can then be provided access to run those published workflows through a custom designed web interface. The executions of these workflows can also be distributed to connected execution nodes to parallelize its evaluation, and scale the use and execution of those workflows to many more users. The workflows can be distributed to a high performance computing infrastructure, without burdening the user with configuration and setup details, Whitespace Explorer connects to this scalable infrastructure, which enables fast optimizations and whitespace explorations. Users will be able to develop and validate ModelCenter® workflows, then publish it to ModelCenter® Cloud. Once published, Whitespace Explorer will have access to those workflows, allowing for whitespace exploration to be performed on those workflows.

The Whitespace Explorer user interface is web based, and therefore is accessible via any modern web browser. All data generated in the exploration is saved on the server, making it easily accessible. The following figures shows how the user interface implements each of the steps previously outlined in the whitespace exploration process.
### Description

**Whitespace Explorer Login.**
Allows each user to have their own account. Manager level accounts are able to view explorations performed by other users.

**Continue existing exploration or start a new one.**
*Users can continue an existing exploration, or start a new one. All exploration data is saved on the server.*

**Create new exploration.**
The user can supply metadata to the new exploration and select from models available on the server they wish to use for the exploration.

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Workflow URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>310</td>
<td>Satellite Exploration</td>
<td>workflows/JPLSatelliteModel/JPLSatelliteModel.pacx</td>
</tr>
<tr>
<td>155</td>
<td>Satellite Model</td>
<td>workflows/Brake/Brake.pacx</td>
</tr>
<tr>
<td>506</td>
<td>Example</td>
<td>workflows/Brake/BrakeWrapper.pacx</td>
</tr>
<tr>
<td>296</td>
<td>New Example</td>
<td>workflows/Whipple/Whipple.pacx</td>
</tr>
<tr>
<td>0</td>
<td>Example</td>
<td>workflows/JPLSatelliteModel/JPLSatelliteModel.pacx</td>
</tr>
</tbody>
</table>

---

**User Interface**

**Whitespace Explorer**

- **Username:**
- **Password:**

**Continue existing exploration or start a new one.**

- *Users can continue an existing exploration, or start a new one. All exploration data is saved on the server.*

**Create new exploration.**

- *The user can supply metadata to the new exploration and select from models available on the server they wish to use for the exploration.*
Workflow Review. The workflow review page is used to define the baseline design. The model can be executed from this view to ensure that the baseline is a valid design.

Pareto Frontier Optimization. The view allows to the user to drag-and-drop variables into the Design Variables, Constraints, and Objectives tables and supply values to define the Pareto frontier search setup.

Visualization. While the Pareto frontier search is in progress, evaluated points will stream live into the plot. The user can hover over points to get the design point values. This view can also be used to specify the selected and desired points.
Visualization plot options.
Both 2D and 3D Scatter plots are supported and the user can specify which variables to display on the axes. The user can also specify whether or not to display invalid designs.

Selected and Desired Point.
The user specifies the selected point by clicking on a point on the Pareto frontier in the plot, the values of that plot are displayed in the table here. The desired point is specified by manually entering the desired value for each objective.
Constraints.
Here the user can experiment with different constraint values and visualize their effect on the design space using the plot. If constraints are adjusted such that there is insufficient resolution in the Pareto frontier, the Pareto frontier search can be resumed using the new constraint values and run until sufficient resolution is achieved.

Round Comparison.
This tab can be used to select rounds other than the one the user currently has active which will display the Pareto frontier from that round on the plot.

Sensitivity Analysis.
Here the user can setup and execute a sensitivity analysis to determine which variables should be included in the direction search.
Direction search. This view contains the results of the sensitivity analysis which the user can use to select which variables should be included as inputs to the direction search. The lower half of the interface lets the user specify the lower and upper bound for each of the input variables to the direction search.

Recommendations. The recommendations page displays the results of the direction search.

Sidebar. In the tool, a navigation panel is available on an expandable side bar which allow the user to return to any previous step in the exploration.

V. Potential Application to the Scientific and Engineering Community

One of the goals during the design process is to explore as many design alternative within given time and resource constraints. Similar to other design space exploration techniques, whitespace exploration represents a new area of study for improving the design process. The whitespace exploration process presented in this paper provides a methodology for the designer to systematically investigate the effects of the assumptions that were made in the initial design process. Through the relaxation of these assumptions, previously unexplored portions of the design space can now be considered in the design process. The results of the whitespace exploration process can also serve as a useful tool when pushing back on system requirements or identifying key areas where additional understanding and technological advancement can best improve system objectives. For example, in the case of satellite design, the whitespace exploration process may discover that battery capacity needs to be improved in order to maintain mass
constraints and improve performance metrics. This information can be used by decision makers to either push back on system constraints, or dedicate resources to improve the battery capacity. The process of discovery enabled by the methodology also could result in the finding that the combined improvement in several technology changes would result in the same effect as a single much larger technology change in a particular area. This trade-off information is valuable to decision makers to make more effective decisions in terms of resource allocations in a project.

Additionally, in performing the presented whitespace exploration process, human interaction with the method invariably will result in a better understanding of the design space. This is because the engineer or scientist is actively making informed decisions at every step. Unlike many “fire and forget” algorithms such as optimization, the methodology was intentionally designed to involve human interaction. This interaction results in better understanding.

Ultimately, the intent is to help in the discovery of the design space as efficiently as possible by challenging assumptions and opening up possibilities that were previously not considered. The focus is to help answer the question of why a particular design is limited in its performance. While the area of whitespace exploration is still in its infancy, the progress made through the development of this methodology is a very good step in maturing this new area of research.

VI. Future Areas of Research and Development

Given that the work presented here represents only a baseline, future work on the prototype tool could include a variety of algorithmic and interface features to improve the whitespace exploration process. Some possibilities include the development of various metrics to quantify the exploration process providing useful feedback to users on how the exploration is progressing, and allow for easier comparison between different explorations of the same model. These metrics can also be useful for comparing future modifications of the whitespace exploration algorithm itself to the baseline. Other improvements can come through integrating modern techniques for optimization and sensitivity analysis which has the potential to provide equivalent, or better, results with fewer computational resources, thus reducing the time required to perform the exploration. The concept of branching within an exploration is another area of interest that we feel could greatly improve the usability of the system by allowing users to ‘branch off’ from any baseline design and explore different combinations of variables and setups within a round.

Acknowledgments

The authors thanks Dr. Steven Wall and Dr. Steven Cornford at the Jet Propulsion Laboratory in Pasadena, CA, for their support of this effort. The concept of whitespace exploration originated from them and was developed with their guidance. This work was funded through NASA contract NNX14CP04C.

References