44th AIAA Aerospace Sciences Meeting and Exhibit
9 - 12 January 2006, Reno, Nevada

AIAA 2006-728

Development of a Collaborative Capability-Based Tradeoff Environment for Complex System Architectures

Patrick T. Biltgen*, Tommer Ender*, and Dimitri N. Mavris†,

Aerospace Systems Design Laboratory (ASDL)
School of Aerospace Engineering
Georgia Institute of Technology, Atlanta, GA 30332-0150

Abstract

The design of complex systems in the presence of changing requirements, rapidly evolving technologies, and design uncertainty continues to be a challenge. Furthermore, the design of future platforms must take into account the interoperability of a variety of heterogeneous systems and their role in a larger "system-of-systems." To date, methodologies to address the complex interactions and optimize the system at the macro-level have lacked a clear direction and structure and have largely been conducted in an ad-hoc fashion. Traditional optimization has centered around individual vehicles with little regard for the impact on the overall system. A key enabler for reduced cost and cycle time is the ability to rapidly analyze technologies and perform trade studies using a capability-based approach. While many entities have expressed a desire to perform capability-based design, the need for a structured discipline exists. This research will examine how collaboration for the design of such systems-of-systems can be enabled through the use of surrogate models and will demonstrate a top-down analysis methodology for the evaluation of systems and technologies with respect to desired capabilities. A technique for inverse design where any variable can be treated as an independent variable is made routine through the structured use of surrogate models and probability theory. For the testbed demonstration, a depoliticized, notional scenario was postulated to develop a testbed environment in which humanitarian aid and supplies must be delivered to forward-deployed troops for dispersal in a host country under fire.

Nomenclature

AFRL	=	Air Force Research Laboratory
ASDL	=	Aerospace Systems Design Laboratory
FANNGS	=	Function Approximating Neural Network Generation System
JCIDS	=	Joint Capabilities Integration and Development System
L/D	=	Lift to Drag Ratio
MOE	=	Measure of Effectiveness
MOP	=	Measure of Performance
SAM	=	Surface to Air Missile
TSFC	=	Thrust Specific Fuel Consumption
UTE	=	Unified Tradeoff Environment

* Graduate Research Assistant, School of Aerospace Engineering, Student Member, AIAA.
† Professor, School of Aerospace Engineering, Associate Fellow AIAA.

Copyright © 2006 by Patrick Biltgen. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission.

I. Introduction

Since the end of the Cold War, the military has been in the midst of a transformational process to address the need for agile, responsive forces that can respond to a wide range of threats under a variety of operating conditions. This need has been accelerated by the events of September 11, 2001 and the realization that military forces have been optimized for several point solutions and do not operate efficiently at off-design conditions. To eliminate redundancy and enable agile response across a spectrum of military power, the Department of Defense has instituted the Joint Capabilities Integration and Development System (JCIDS) to design "born joint" capabilities from the early phases of the design process. These joint capabilities rely heavily on the interoperation of heterogeneous elements in the large-scale system architecture that constitutes the U.S. military and are dominated not by the attributes of individual systems, but rather, the complex interactions of multiple systems that combine to provide a capability. The design of "systems-of-systems" requires collaboration between multiple entities that are responsible for the development of systems and subsystems. This research identifies how such collaboration can be efficiently performed and using a top-down analysis process to identify system solutions that meet top-level capability-based goals.

II. Problem Definition

To demonstrate a top-down, capability-focused design methodology, a relevant problem of interest must first be identified. To avoid security issues and proprietary concerns, a depoliticized, notional scenario was postulated to develop a testbed environment. In the scenario, humanitarian aid and supplies must be delivered to forward-deployed troops for dispersal in a host country; however, the country of interest is protected by surface to air missiles (SAM's) and local insurgents threaten ground troops. As a result, friendly forces cannot stay in one area for an extended period of time: the density of the local insurgency controls the *loiter time* of friendly forces around the receiving point. Delivery aircraft must quickly deliver supplies to forward deployed friendly forces while avoiding attacking SAM sites themselves. The desired capability is "support forward deployed forces through strategic airlift." Instead of tracking system-level metrics such as turn rate and gross takeoff weight, metrics of interest (or measures of effectiveness) of a proposed system solution include *total aid delivered*, *number of friendly aircraft lost*, and *SAM sites neutralized*. The objective in the scenario is to maximize aid delivered and minimize friendly losses. Depending on the parameters of the scenario, neutralizing SAM sites may or may not contribute to the overall effectiveness of the proposed solution.

III. Technical Approach

In this paper, a process will be identified for the selection of system architectures and technologies that provide the aforementioned capability to the highest degree of effectiveness. The simplest method for performing this analysis is a committee approach where various solutions are proposed and evaluated qualitatively. This process can be ad-hoc and subject to personal biases; therefore, a quantitative, physics-based approach that uses modeling and simulation to assess effectiveness is desired. This approach raises several technical challenges. First, since multiple systems and subsystems must be modeled to assess the effectiveness of a proposed "system-of-systems," collaboration between multiple entities is required since different entities can be considered experts in individual components. Unfortunately, cooperation between industry partners that are otherwise competitors is rife with intellectual property and security concerns. A collaboration technique that addresses these serious issues must be devised. Secondly, while software exists to promote the integration of physics-based computer codes at multiple geographically distributed locations, large-scale integrated simulation projects have historically been extremely costly and complicated. A proposed approach that mitigates security concerns and information technology obstacles is to use parametric **surrogate models** of identified physics-based design tools, integrated at a central location, to perform the hierarchical system-of-systems analysis. Surrogate models are fast-running approximations of the actual engineering codes that, when properly constructed, exhibit no loss in fidelity when compared to the engineering tool. Furthermore, since they are built for a specific problem they can shield intellectual property because they cannot be reverse engineered. Surrogate models are platform and operating system independent since they are only equations, easing the IT burden on the respective collaborative partners. In this manner, surrogate models can be traded as a "currency of communication" between individuals in a collaborative activity. This paradigm is similar to a breadboard for prototyping electrical systems: surrogate models act as the elements of a circuit and have well-defined and standardized interfaces to each other.

It is important to note that surrogate models can be used as the components of an integrated, geographically distributed design environment, collaboration can also be performed by integrating surrogate models at a central location. This approach, where the surrogate models are generated by the respective disciplinary experts, was chosen to minimize the use of available resources and maximize the probability of success in the time allotted for the study.

The surrogate modeling approach also answers another major technical challenge involved in large-scale design and simulation of systems-of-systems: the long run times for constructive simulations are not conducive to real-time parametric design space exploration. Since surrogate models are fast-running equations, they can be evaluated nearly instantaneously and enable probabilistic analysis over a multidimensional space (although advanced visualization techniques will be needed to understand the results). Surrogate models are of great interest to this research due to their ability to solve these two major technical challenges; however, the approximation of constructive simulations is not trivial. Due to the complex interplay of different systems in such simulations, there are multiple discontinuous behaviors that cannot be modeled using traditional polynomial response surface equation surrogate models. A technique is needed to capture the non-linearity of the problem while still maintaining the speed and accuracy advantages of surrogate models. Instead of using polynomial approximations, an approach was proposed that uses neural network surrogate models due to their ability to capture the highly multimodal and discontinuous behaviors of the proposed constructive simulations. A three-layer, feed-forward neural network was used to model the complete military simulation hierarchy. Unfortunately, due to the large number of responses that must be tracked to calculate system-of-systems level capabilities, the training process for the neural network could be a time consuming, error-prone, manual process that relies more on serendipity than structure. To overcome this limitation, an automated neural network regression tool, the Function Approximating Neural Network Generation System (FANNGS), developed by Johnson was used to optimize the training process [1]. The graphical user interface facilitates computer-controlled exploration of the ideal number of hidden nodes and the identification of model coefficients within user specified bounds to optimize the neural network topology. Using FANNGS, the man hours required to generate neural network surrogate models were reduced by over 90% and the accuracy of the equations created improved dramatically.

Using neural network surrogate models, the hierarchical modeling and simulation environment consisting of vehicle propulsion system models, aircraft (platform) synthesis and sizing codes, and the top-level military campaign analysis were approximated and integrated into a multi-level Unified Tradeoff Environment (UTE) [2] that allows simultaneous trades between design variables, requirements, and technologies at each hierarchical level as shown in Figure 1.

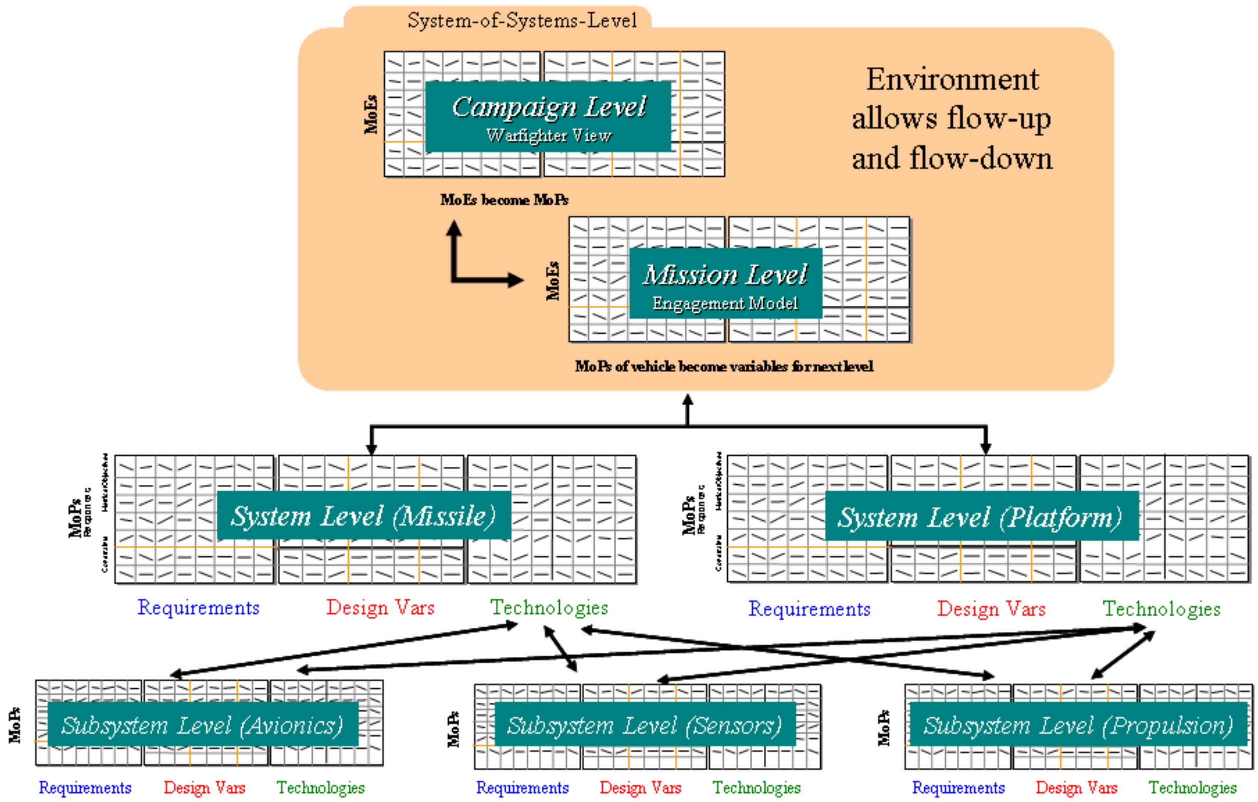

Figure 1: Hierarchical, Surrogate Modeling Environment for Systems-of-Systems Analysis.

While this integrated modeling and simulation environment uses a suite of surrogate models, the surrogates must be created from parametric, physics-based design tools. Another major challenge was the identification of design tools with the appropriate fidelity, degrees of freedom, and availability for this research. Instead of acquiring the proprietary tools of collaborative entities from industry, the approach used due to the short time for the study was the use public-domain tools to create surrogate models of the respective systems and subsystems with validation from industry partners. A wide variety of models were required to effectively perform the necessary campaign simulation to assess effectiveness against capabilities. First, a parametric aircraft model was developed using the energy-based sizing formulation advocated by Mattingly [3] and the Breguet range equation for vehicle sizing. The energy-based equation for thrust-to-weight ratio as a function of aircraft design parameters is of the form:

$$\frac{T_{SL}}{W_{TO}} = \frac{\beta}{\alpha}\left\{\frac{qS}{\beta W_{TO}}\left[K_1\left(\frac{n\beta}{q}\frac{W_{TO}}{S}\right)^2 + K_2\left(\frac{n\beta}{q}\frac{W_{TO}}{S}\right) + C_{D_o} + \frac{R}{qS}\right] + \frac{1}{V}\frac{d}{dt}\left(h + \frac{V^2}{2g_o}\right)\right\}$$

The above parameters are defined in detail in Reference 3. Due to the prevalence of enemy SAM sites on the battlefield, a parametric weapon model was developed using the methods in

Reference 4 to analyze the necessary weapon attributes to handle various threats. Next, a parametric propulsion system model for a turbofan, turbojet, and ramjet engine was creating using GasTurb 10 [5]. Results of a parametric exercise of the generated surrogate models for the propulsion system are shown in one form in Figure 2. This three-dimensional Pareto Frontier shows that for each of the three engine types, there is a well-defined boundary between specific thrust and TSFC that can be obtained through variation of the engine design parameters across reasonable ranges. Also visible are clear boundaries over the range of Mach number where different engines are applicable. Using JMP statistical analysis software [6], a multivariate profiler can be used to understand these trends over all possible dimensions by generating all possible "slices" of all possible three-dimensional surfaces as shown in Figure 3. The multivariate profiler is useful in analyzing capabilities: desired thresholds on the measures of effectiveness can be highlighted in the upper left corner and the corresponding system solutions can be identified. This technique will later be utilized across the multi-level tradeoff environment shown in Figure 1.

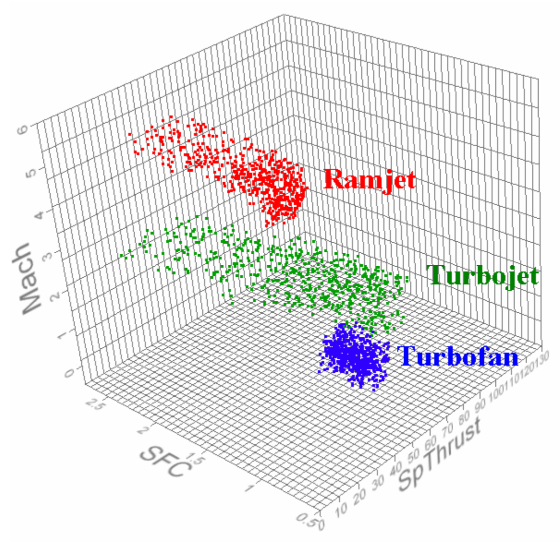

Figure 2: Three-Dimensional Pareto Frontier of SFC and Specific Thrust as a Function of Cruise Mach Number.

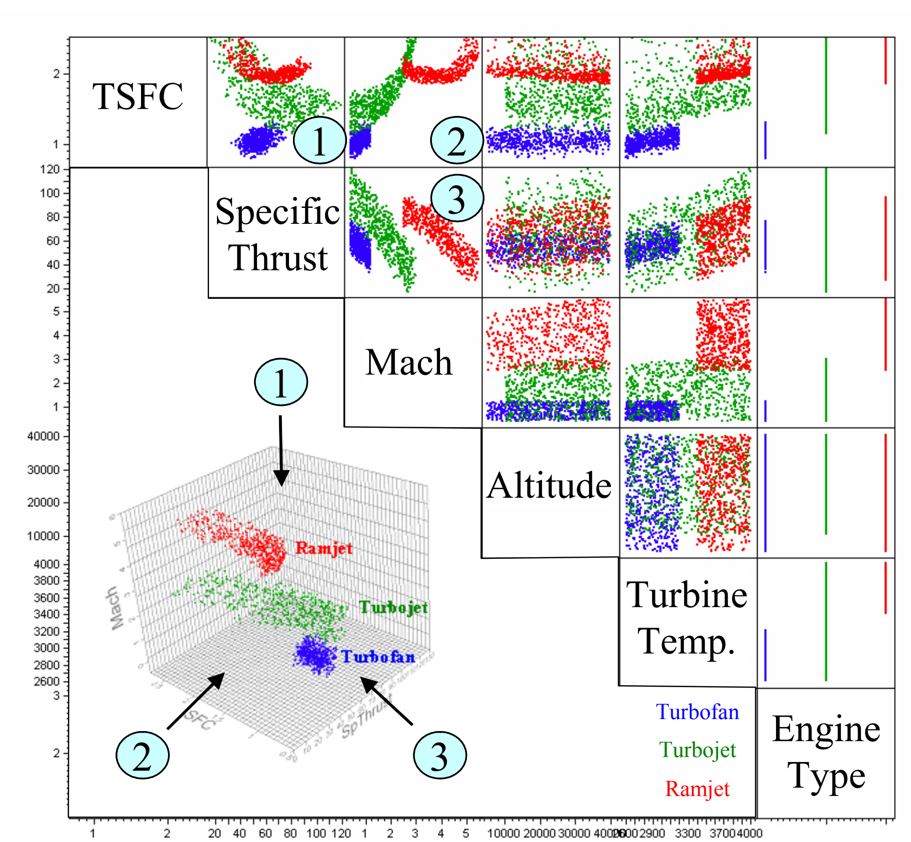

Figure 3: Multivariate Profiler for a Parametric Propulsion System Model.

Finally, the major code development effort in this research task was the creation of a campaign/engagement model for the chosen scenario. A simulation was written in MATLAB using basic equations from Driels [7], Ball [8], and Przemieniecki [9]. Much of the logic in the campaign code was developed through consultation with Dr. Martin Ulehla from the Raytheon Corporation. In this simulation, there are two types of objectives. The first objective is a SAM site, which is a hostile target that shoots back at friendly assets. The second objective is a time-critical concentration of friendly troops to which aid must be delivered. Due to the nature of the local insurgency, friendly troops cannot remain visible indefinitely. Their *loiter time* is a variable in the scenario. Friendly aircraft consist of two types: fighters, which attack the SAM sites with weapons and cargo aircraft, which drop aid to the friendly forces on the ground. Fighters are parametrically variable with a speed between Mach 0.8 and 4.0 while cargo aircraft are variable between Mach 0.8 and 2.0. Both types of aircraft have variable stealth characteristics. Each fighter can engage one SAM site at a time and each cargo aircraft can carry ten small aid pallets. These parameters were chosen to develop a notional, unclassified scenario and could be easily changed to represent a different situation.

After creating a linked, hierarchical modeling and simulation environment using the aforementioned tools, it is possible to perform physics-based analysis throughout the military simulation hierarchy. Neural network surrogate models provide a means for quickly assessing the system-of-systems level impact of lower level design decisions by approximating the physics-based codes throughout the hierarchical modeling and simulation environment shown in Figure 1; however, these solutions are still point designs. The ability to rapidly generate point designs enables the use of probability theory to analyze a large number of designs for statistically significant trends at the system-of-systems level. Using Monte Carlo Simulation on the input parameters, a large number of cases can be quickly executed, resulting in "clouds" of solutions at the system-of-systems level. These clouds represent different levels of effectiveness at providing one or more capabilities. With this information, it is possible to highlight desired thresholds on the capability metrics at the system-of-systems level and identify system solutions that meet these limits. The multivariate profiler can again be used, as shown in Figure 4, to discover these solutions using this top-down analysis technique. In the environment, the points from the Monte Carlo Simulation are linked across multiple hierarchical levels. A desired solution at the top level is linked directly to a specific threat. The corresponding friendly system attributes that provide the required effectiveness against that threat are visible at the aircraft, weapon, and propulsion system level. Using the JMP scripting language, files can be hyperlinked to each point such as engine flowpaths, weapon cross sections, flight paths, and aircraft 3-D geometry models. Clicking on any point can highlight a "vector of attributes" for the identified system as well as link directly to any of these supporting files that may contain information that is difficult to capture with numerical metrics.

In this manner, surrogate models and probabilistics enable "inverse design," where *any variable can be treated as an independent variable.* Using this technique, thresholds can be placed on vehicle performance, system-of-systems level campaign outputs, cost, schedule, risk, and any parameter that can be calculated analytically using the results of a physics-based tool anywhere in the simulation hierarchy. Previously, a top-down, inverse design process was not realizable. Using surrogate models and probability theory, inverse design can be readily performed on a wide range of problems in the modeling and simulation community.

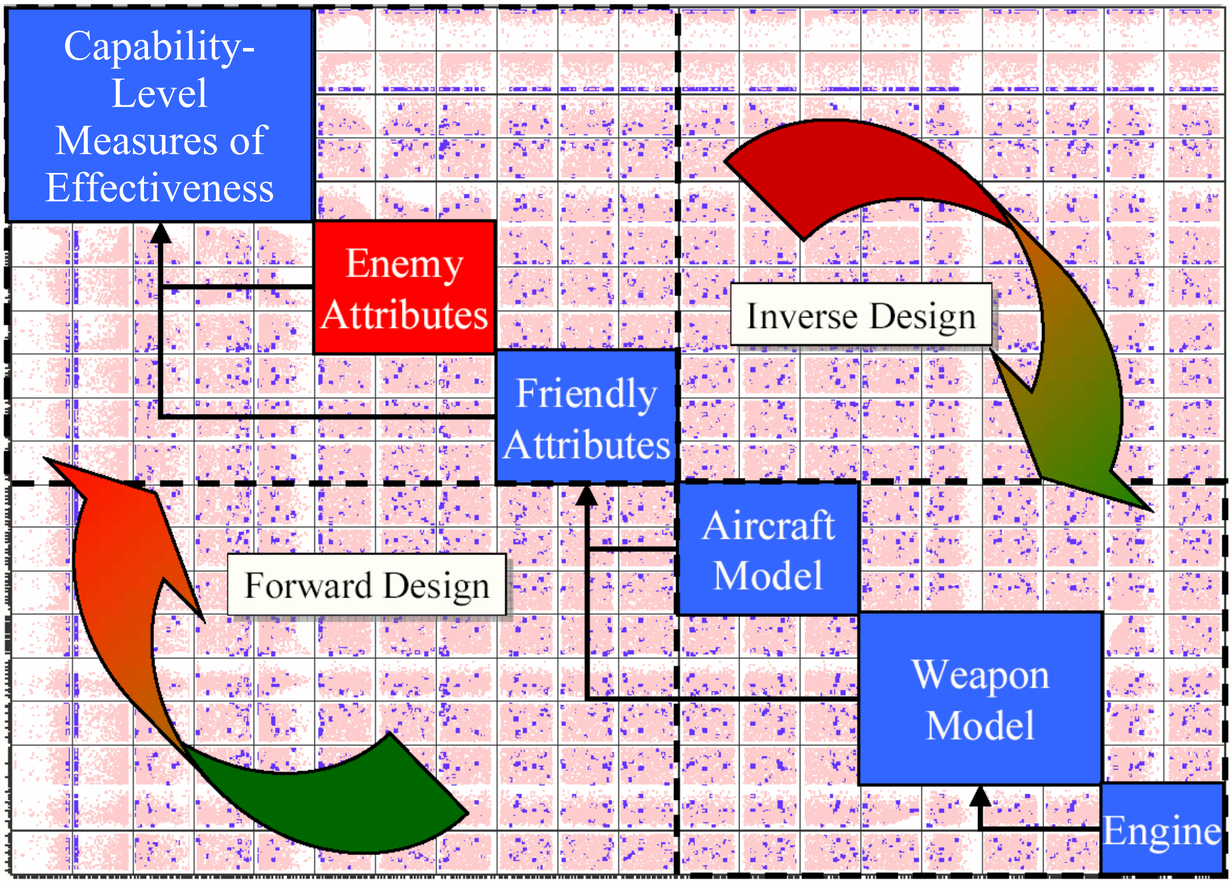

Figure 4: Multivariate View of the Top-Down Decision Making Process for Inverse Design.

The large amount of information generated by the Monte Carlo analysis requires advanced visualization and statistical techniques to analyze and understand. For a large constructive simulation, when the data is analyzed using the multivariate profiler it becomes obvious that solutions meeting top-level criteria are non-unique: there are many different ways to accomplish a mission with the same level of effectiveness. Each of these solutions may differ in cost, risk, technology readiness, or use of existing assets; however, tracking all these degrees of freedom simultaneously quickly becomes overwhelming for the analyst. A technique is needed to reduce the dimensionality of the problem to a manageable set. This creates a dilemma: on one hand, the freedom to make decisions in multiple dimensions is desired, but this freedom makes it difficult to actually make decisions. The proposed approach uses the largest number of degrees of freedom that are reasonable for a given problem, and selectively locks these dimensions down after the surrogate models are created. This Rubik's Cube®-like approach to design allows an analyst to examine certain design trades with all other degrees of freedom held constant, and then lock other degrees of freedom and make other trades. The locking procedure can be quickly conducted with surrogate models by simply changing the distribution on the input parameters to a single value, whereas such exploration without surrogate models would require re-executing cases ad nauseam. Lastly, the final answer can be confirmed by re-opening all degrees of freedom.

The concept of locking degrees of freedom is analogous to the mathematical concept of total derivatives versus partial derivatives. A partial derivative is a derivative with respect to one variable with all other variables held constant and can be related to each two-dimensional slice in Figure 3. A total derivative is the sum of all partial derivatives with respect to all variables and is indicative of the entire multivariate profiler with variation over all degrees of freedom. Comparatively, partial derivatives (2-D slices) are much easier to understand than an *n*-dimensional hyperspace.

To demonstrate the analysis of technologies across the domain of total and partial derivatives, an example was created using k-factor scale parameters on TSFC, L/D, and aircraft drag as shown in Figure 5. These parameters impact the system level attributes of aircraft range and speed. When the aircraft is placed in the military simulation, the system-of-systems level parameter of aircraft lost can be calculated. This metric is also a function of the friendly force loiter time and the capability of hostile SAM's. In Figure 5, the gray dots show how aircraft range, aircraft speed, and aircraft lost change when k-TSFC, k-L/D, and k-Drag are varied uniformly over the entire range of variability (total derivative). The green solid lines show how these points then collapse to a single trend line when k-TSFC and k-L/D are held constant and only k-Drag is varied uniformly (partial derivative). As a result, the green line in the topmost row of the multivariate profiler shows how each of the output parameters vary as a function of k-Drag only (Item A in Figure 5).

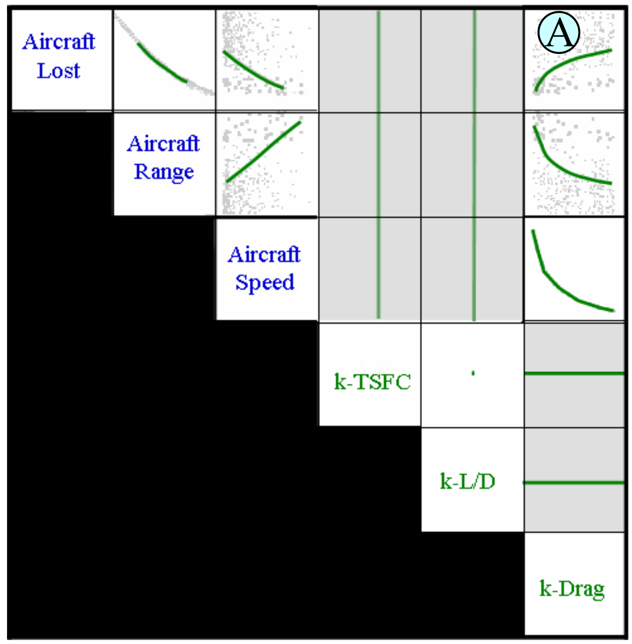

Figure 5: Multivariate Plot for Technology Exploration Showing Partial and Total Derivatives.

Using the inverse design technique, it can be shown that aircraft lost is minimized when drag is lower, when fuel consumption is lower, or some combination of both. Allowing all parameters to vary reveals clear Pareto Frontiers in the allowable settings of these variables, identifying potential technology tradeoffs between aerodynamics and propulsion. When lower-level parameters are analyzed, it is also possible to identify propulsion system characteristics and specific aerodynamic features that contribute to lowering drag or fuel consumption. This is an enabling technique to allow propulsion companies and subsystem manufacturers to better quantitatively assess how subsystem design decisions provide value to the ultimate customer in terms of capabilities provided, avoiding expensive and potentially risky technology development programs with unknown results.

The parametric technique can also be used to calculate necessary platform characteristics in the presence of changing enemy capability. For example, since the time that friendly forces can be available for receiving aid is inversely proportional to the density of insurgents, it is desired to assess how changing insurgent density impacts the overall mission effectiveness and the specification of aircraft requirements to achieve a certain mission effectiveness. Figure 6 shows the relationship of aid delivered against two platform characteristics for a high insurgency density. Only high speed and long range solutions can deliver maximum aid due to the short loiter time of friendly forces on the battlefield. On the other hand, Figure 7 shows that for a low density insurgency, the aircraft speed and aircraft range have no impact on mission success: if friendly forces can loiter on the battlefield indefinitely, a high speed, long range solution is not required. This could imply that higher capability solutions would be needed early in a conflict and lower capability solutions could be infused at some point during the aid delivery mission to maintain constant effectiveness while minimizing the risk to expensive platforms. It is also of interest to note that both Figure 6 and Figure 7 were creating using the same neural network equation and resulted from the change of a single input parameter. The neural network model is able to capture the two extreme cases with high accuracy and high speed. This case is indicative of only one example in the parametric modeling and simulation environment. Variables such as aircraft stealth, payload of aid per sortie, density of SAM sites, speed of SAM missiles, and other enemy and friendly attributes are all on slide bars in the collaborative design environment. Using advanced visualization, these parameters can be changed in real time and the graphics in Figure 6 and Figure 7 update automatically. In this manner, variable threats, theater constraints, and variable requirements at all hierarchical levels can be varied. In a collaborative venue, partners from various entities can play "what-if" games with other subject matter experts to aid the decision making process for the selection of future systems.

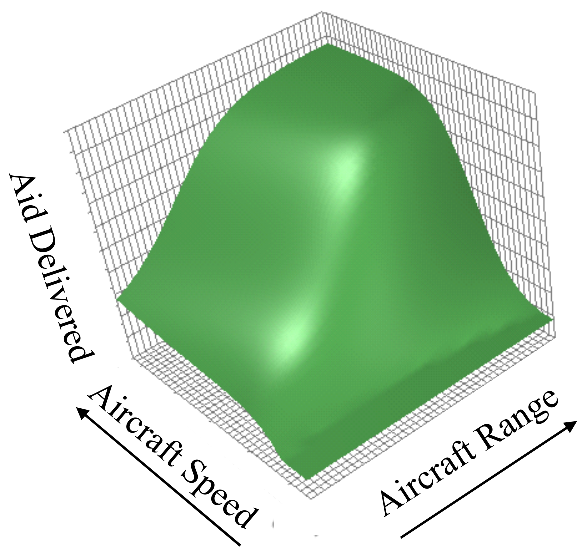

Figure 6: Aid Delivered Versus Platform Characteristics for High Insurgency Density (Low Friendly Loiter Time).

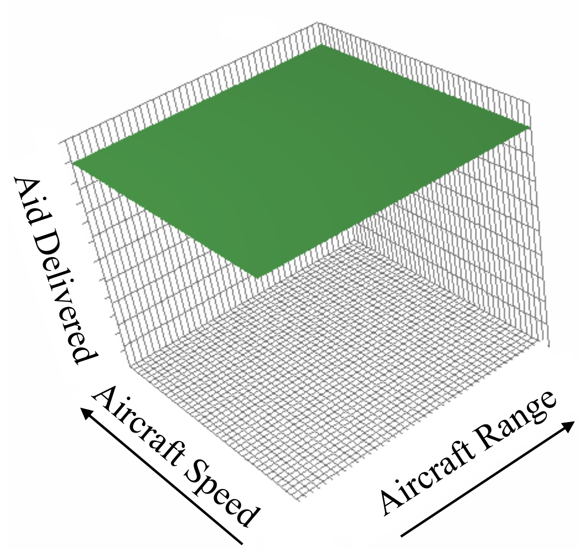

Figure 7: Aid Delivered Versus Platform Characteristics for Low Insurgency Density (High Friendly Loiter Time).

IV. Summary

The goal of this research was to demonstrate that collaborative design and decision making is possible for constructive simulation of complex systems. As a variation on the traditional, information technology-intensive integrated distributed design environment, a local design environment consisting of surrogate models was used to perform the collaborative activity. Surrogate models were generated around public-domain tools using expert input and verification to create a linked, hierarchical, modeling and simulation environment of a mission to deliver aid to forward deployed troops in a hostile environment. Using neural network surrogate models, the complex nonlinearities of the constructive simulation were accurately modeled over a range of input parameters that represented multiple degrees of freedom at the platform, propulsion system, enemy attribute, and campaign level. An inverse design approach where any variable can be treated as an independent variable was developed using surrogate models and probability theory. This technique was matured and standardized for use on other problems.

Analysis of the information generated in this study relied on visualization and statistical techniques as well as an understanding that designers are traditionally not used to the ability to control more than one or two factors at a time. A selective locking approach was demonstrated and graphical techniques were used extensively to assess the effectiveness of systems solutions at providing a required capability in the presence of changing requirements and enemy attributes. The methods outlined in this research serve not as a "final answer" on the selection and acquisition of complex system architectures, but rather as a tool to aid the decision maker in quantitatively making such decisions in the presence of multiple, conflicting requirements and unknown future threats.

V. Future Work

The testbed design environment utilized a single "thread" through a complex system architecture. Future work will examine multiple threads in an effort to determine the optimum set of systems and technologies to provide a desired capability with respect to required thresholds that may vary as a function of time. Variable threats and multiple missions may be included. Ultimately, a robust system-of-systems solution that provides maximum effectiveness over a range of capabilities is desired. The surrogate modeling approach proposed ultimately seeks to address this issue.

Finally, the example case shown in this paper demonstrates that a fast aircraft with long range is desired to deliver maximum aid to forward deployed troops. Although the technique quantifies such an intuitive result and identifies specific subsystem characteristics that provide capabilities, an interesting question arises: can a top-down, capability-focused design methodology identify *unconventional* or *nonintuitive* solutions that are more effective than traditional designs? What is gained through the additional expenditure of resources to create a hierarchical environment simply to validate the opinion of subject matter experts? Finally, if such a nonintuitive answer is discovered as a result of this technique, can it be shown that the answer is genuine and not simply the result of a computational error in one of many computational models? Future research will seek an existence proof of such solutions and attempt to validate the results with respect to the final research question.

Acknowledgements

The authors would like to recognize Mr. David Brown from the Air Force Research Laboratory Vehicles Directorate and the Wright Brothers Institute for their generous support of this research. We also wish to thank Mr. Robert Nelson and Dr. Martin Ulehla from the Raytheon Corporation and Mr. Carl Weiss from Pratt and Whitney for their advice and support throughout this study. Finally, the authors would like to recognize graduate students Bjorn Cole, David Culverhouse, William Engler, Matthew Gottfredson, Jared Harrington, Russell Joffrion, Carl Johnson, Joseph Roop, Mark Smith, Louis Wei, and Bryan Wood for their hard work on this research effort.

References

[1] Johnson, C., Function Approximating Neural Network Generation System (FANNGS) Documentation, Version 1.2. Georgia Institute of Technology, 225 North Avenue, Atlanta, GA 30332, June 14 2005.
[2] Baker, A. P., Mavris, D. N., "Assessing the Simultaneous Impact of Requirements, Vehicle Characteristics and Technologies During Aircraft Design," in Presented at the 39th Aerospace Sciences Meeting and Exhibit, Reno, NV, AIAA-01-0533, January 8-11, 2001.
[3] Mattingly, J. D., Heiser, W. H., and Pratt D. T., Aircraft Engine Design, Second Edition. AIAA Education Series, 2002.
[4] Fleeman, E. L., "Tactical Missile Design". Reston, VA: American Institute of Aeronautics and Astronautics, 2001.
[5] Kurzke, Joachim, Gasturb 10, Computer Program, 2005.
[6] SAS Institute Inc., Cary, NC, JMP, Computer Program and Users Manual, 1994.
[7] Driels, M. R., "Weaponeering: Conventional Weapon System Effectiveness". Reston, VA: AIAA Education Series, 2004.
[8] Ball, R. E., "The Fundamentals of Aircraft Combat Survivability Analysis and Design". AIAA, 2003.
[9] Przemieniecki, J., "Mathematical Methods in Defense Analysis". AIAA, 3rd ed., 2000.