FACT: An M&S Framework for Systems Engineering

Tommer R. Ender and Daniel C. Browne
Georgia Tech Research Institute
Atlanta, GA 30332-0840
tommer.ender@gtri.gatech.edu
daniel.browne@gtri.gatech.edu

William W. Yates and Michael O'Neal
Marine Corps Systems Command
Quantico, VA
william.yates@usmc.mil
michael.oneal1@usmc.mil

ABSTRACT

The ability to leverage models within a broader application of systems engineering has been limited in many cases due to lack of capability for distributed simulations to concurrently model multiple attributes of a system. Military performance models typically emphasize one area such as mobility or survivability, and are rarely connected to models for reliability, maintainability, and availability or procurement and lifecycle sustainment cost. Federations of training simulations to support the requirements of the training community are numerous and well-studied, but leveraged far less during the early phases of systems analysis. The Framework for Assessing Cost and Technology (FACT) is an open architecture web services based environment that enables the interconnecting of models to provide a rapid exploration of the design tradespace in support of systems engineering analysis. FACT is government owned, model agnostic, and capable of linking disparate models and simulations of both government and commercial origin through the application of community established data interoperability standards. This paper describes the utility of using FACT to achieve near real-time analysis for exploring the design parameter trades that affect the overall performance, reliability, and cost of a system design. FACT provides decision support tools to the acquisition program IPT to manage risks of cost, schedule, and performance through a rapid analysis of alternative technology and materiel using surrogate models, or equation regression representations of more complex M&S tools, as illustrated through several successful implementations discussed in this paper. FACT will ultimately reduce program development and life cycle costs, both of which are tenets of effective "should-cost" management.

ABOUT THE AUTHORS

Dr. Tommer R. Ender a Senior Research Engineer at the Georgia Tech Research Institute, and serves as head of the Systems Engineering Applications Branch. His primary area of research includes development of systems engineering tools and methods as applied to complex systems-of-systems, concerned with supporting decision making through a holistic treatment of various problems. Dr. Ender is an instructor in Georgia Tech's Professional Masters in Applied Systems Engineering. He earned a B.S., M.S., and Ph.D. in Aerospace Engineering from Georgia Tech.

Daniel C. Browne is a Research Engineer at the Georgia Tech Research Institute. His primary research area involves developing web-deployed systems engineering toolsets to provide unique, collaborative, and real-time analysis tools to decision makers. He earned a B.S. in Aerospace Engineering and an M.S. in Computational Science and Engineering from Georgia Tech. He is currently pursuing a Ph.D. in Computational Science and Engineering focusing on techniques to manage and optimize power and/or energy usage required to accomplish a constant amount of computational work.

LtCol William W. Yates is a graduate of Naval Postgraduate School with a M.S. degree in Modeling, Virtual Environments, and Simulations. His assignments in M&S include Officer in Charge of the MAGTF Training Command Battle Simulation Center, M&S officer for the Program Manager for Training Systems, Assistant Program Manager for Range Training Aids, Devices and Simulators, and Deputy M&S Lead for Marine Corps Systems Command. He is presently assigned as the project manager for FACT.

Mr. Michael O'Neal is the modeling and simulation lead at the Marine Corps Systems Command. He was commissioned an Ensign, US Navy in 1975 after earning a B.S. in Physics from Miami University. He joined the Naval Reserve as an Engineering Duty Officer retiring with the rank of Captain. He holds an M.S. in Engineering Administration from The George Washington University and a MPA from Indiana University. He has held various civil service assignments including Deputy PM for Directed Energy and Electric Weapon Systems.

FACT: An M&S Framework for Systems Engineering

Tommer R. Ender and Daniel C. Browne
Georgia Tech Research Institute
Atlanta, GA 30332-0840
tommer.ender@gtri.gatech.edu
daniel.browne@gtri.gatech.edu

William W. Yates and Michael O'Neal
Marine Corps Systems Command
Quantico, VA
william.yates@usmc.mil
michael.oneal1@usmc.mil

INTRODUCTION

Background

A shrinking defense budget and constant demands in the Department of Defense (DoD) for efficiency mandates new processes be developed to help our decision makers and designers. General Dempsey highlighted this need during his confirmation hearing for Chairman of the Joint Chiefs of Staff. Congress asked, "General Dempsey, what's the remedy for Admiral Mullen's belief DoD has 'lost the ability to prioritize, to make hard decisions, to do tough analysis, to make trades'?" The Framework for Assessing Cost and Technology (FACT) intends to help DoD perform the analysis, conduct the trades, and make those hard decisions.

FACT was initiated at a time when the Marine Corps had just terminated the EFV (Expeditionary Fighting Vehicle) Program and was initiating an AoA for the Amphibious Combat Vehicle (ACV). The Marine Corps Systems Command surveyed the tools available to make the AoA process more efficient. Out of this need FACT was born.

The Framework for Assessing Cost and Technology (FACT) emerged to answer the fundamental acquisition questions:

- How well will the system perform?
- How reliable will it be?
- How much will it cost? and
- When can we get it?

The FACT process achieves the capability to answer these questions concurrently rather than in a stove-piped independent fashion. FACT algorithms recognize the inter-dependence of design and maintenance and procurement philosophy on the tradespace. The options in the tradespace represent the inter-related impacts of cost, performance and reliability based on the multitude of design options available to the Program Manager. Based on the options selected, FACT calculates the procurement cost for the system and projects, the operational and support costs for the system versus a level of performance, and associated reliability metrics. Additionally, FACT is designed to support the Analysis of Alternatives (AoA) process, comparing viable options against weighted objectives.

The FACT framework focuses on interoperability and data sharing with the emphasis centered on metadata. Definition of metadata is the building block on which the FACT data sharing is founded. FACT was designed on a philosophy of open architecture to enable extensibility. To achieve this, and avoid the encumbrance of licensing fees limiting its use or tethering it to a single manufacturer over its lifetime, FACT was built using open source software and government-owned code. Architectural guidance for FACT mandates a government-owned toolset for accommodating either government or commercially developed software and models inside a loosely coupled federation. Additionally, FACT must be web-based and accessible from common computer workstations in the NMCI (Navy Marine Corps Intranet) environment. FACT complies with mandated DoD information assurance standards and is in the process of DIACAP accreditation.

In a 22 April 2012 Memo Ashton Carter, Undersecretary of Defense for Acquisition, Technology, and Logistics, outlined further guidance for the services to implement the "will-cost/should-cost" strategy. FACT addresses the challenge of how to determine will- and should-cost and then apply it in a concurrent tradespace analysis. FACT has developed cost estimation relationships between the design components and addresses the maintenance philosophy, repair vs. replace, obsolescence analysis, disposal strategy, and usage data developed from operational scenarios to project operations and support cost information. Similarly, procurement cost data has been gathered from contracts and purchase documents.

FACT also needs to address the question which doomed earlier modeling efforts of this type: "Are your models accredited?" FACT provides the "pedigree" of the models incorporated into its framework. Recognizing the need for quick response from user queries, FACT incorporates the use of surrogate models in its design, which are parametric regression equation representations of high fidelity M&S tools developed via Designs of Experiments sampling (Forrester, 2008). These surrogate models are developed offline from FACT, but are easily integrated into the framework.

Motivation for FACT

Addressing the broad challenges of modeling and simulation (M&S) support for the acquisition enterprise is a huge problem space and requires some upfront choices about where increments of benefit can be obtained quickly and with the greatest return on investment. Recognizing the choices made during the DoD 5000 pre-milestone A conceptual design of systems offers the greatest opportunity to influence the performance and cost of a system, the authors chose to initiate FACT to provide tradespace analysis during conceptual design. Other stages of the system lifecycle can benefit from the FACT process, but the conceptual design phase is where both good and bad decisions have the greatest impact on cost and performance. It was also necessary FACT provide a useful tool to specific program offices. While the Marine Corps performs systems engineering across the gamut of systems, the most immediate and largest opportunity to realize a benefit was in the domain of ground tactical vehicles.

Consequently, the first applications of FACT were to the Amphibious Combat Vehicle (ACV) and the recapitalization program from the HMMWV, building heavily on prior work in support of the Marine Personnel Carrier (MPC) program. The focus on these ground vehicles does not imply the FACT framework is not equally applicable to weapon systems or Command and Control (C2) systems. Exploratory efforts are underway to examine the applicability of FACT in support of naval surface vessel programs, weapon systems, and Unmanned Aerial System (UAS) programs with joint applicability.

Organizing Principles

M&S in support of defense acquisition often suffers from stove-piped processes, creation of boutique solutions and one-off tools without broad application beyond a specific program, and lack of authoritative data management to facilitate the reuse and update of models. To address these issues, FACT approaches the problems from the perspective of creating a process based on open architecture and open standards, focusing on the portability of data rather than direct communication between disparate models of vastly different pedigrees. FACT facilitates creation of a federation of models and access to databases by creating a common language for data interchange based on the Systems Modeling Language (SysML), described in more detail later in this paper.

Literature Review

Much work has been conducted, and breakthroughs documented, in the general area of collaborative design and decision making. The authors, however, have not been able to find any published literature proving comparable, complete functionality and capability to what is current and/or planned for FACT; however, many of the concepts introduced in this paper have been developed and applied in other areas.

Ender et al. (2009) describe an approach enabling decision makers to assess and identify those technologies which are critical to the success of a system yet are not matured. Classic systems engineering tools and processes such as Quality Function Deployment (QFD) are used to prioritize user needs for an armored personnel carrier, which in turn prioritize high impact elements of the vehicle's functional architecture. Technologies are then scored based on satisfaction of high impact function, maturity, and compatibility with other technologies. Luskin (2010) shows how advanced designed methods could be applied to explore design tradespaces for a Fuel Efficiency Demonstrator. Surrogate models capture high fidelity modeling and simulation and are integrated into an Excel based tool, enabling a (single) user to conduct trade studies. This integrated tool is then explored for designs that meet various requirement constraints via a top-down, filtered Monte Carlo methodology.

McDermott et al. (2011) discuss collaborative development of system architectures utilizing fundamental architecting principles, as documented by Maier (2009), and executed in SysML. Xiaoquing et al. (2007) introduce a web-based collaborative engineering framework that supports conflict resolution in addition to multi-user access to engineering tools. The authors show how large scale argumentation networks based on many stakeholders can be used to select favored design alternatives amongst designs from collaborative solid modeling software.

Novelty of Approach

Exploring design tradespace to determine in real-time whether a proposed system design is both technically feasible, while also being within budgetary constraints for procurement and sustainment, is essential for both program managers and systems engineers. Performance requirements that are beyond the reach of technology, and attempts to achieve those requirements with new untested technologies, is a major cause of expensive program failures such as cancellation or forcing a new baseline. FACT is an M&S framework that enables real-time exploration of design tradespace to gain insight into performance, cost, and reliability of a proposed system or for proposed improvements to existing systems.

FACT stands apart from previous tools for exploring design tradespace in that it is model agnostic, built on an open and extensible architecture, characterized entirely in the Systems Modeling Language (SysML) open standard, and enables a stakeholder team to collaborate in real-time through a zero-footprint web interface. Additionally, FACT is built up entirely from open source software (OSS), complying with the DoD's strategy to exploit OSS to "update its software-based capabilities faster than ever, to anticipate new threats and respond to continuously changing requirements" (DoD, 2009). FACT does, however, enable interaction with commercial or otherwise proprietary software; this may include specific web application visualizations such as Adobe Flash on the client side "dashboard", or integration with M&S tools for specific analysis.

By defining a system through the data inputs and outputs of various performance, cost, and reliability models FACT facilitates concurrent use of modeling tools as disparate as spreadsheets and compiled executable software. The flow of data between various models is mediated by FACT to provide immediate answers to the user. For example, the impact on mobility, procurement cost, and lifecycle maintenance costs resulting from changes made to the armor thickness of a tactical vehicle in pursuit of better survivability are calculated immediately. This process prevents an engineering decision made to attain the objective performance in survivability from being made without seeing the adverse impact that the decision has on the speed, acceleration, and fuel consumption of the vehicle.

BOTTOM –UP DESIGN, TOP-DOWN ANALYSIS

Engineering Design

To fully understand the implications of altering a design parameter in a system configuration, a tradespace exploration tool must provide instant feedback on how those changes impact all system metrics. Linking models from each of these domains together in a federation that communicates strictly by data inputs and outputs provides immediate visibility of both first and second order consequences of a design change.

For example, systems engineers working on the Survivability Key Performance Parameter (KPP) for a ground tactical vehicle understand there are second order effects to the design parameters they select for vehicle armor. However, in the conventional process those systems engineers do not see the tradespace in other KPPs impacted by design parameters assigned to meet the Survivability KPP, for example. Intuitively, adding armor thickness has consequences on a vehicle's interior and exterior dimensions, weight, acceleration, maximum climbing grade, righting moment, and fuel consumption. To see these effects, the design team needs a federation of engineering and cost models that communicate using data tagged with precise metadata definitions. FACT is such a framework.

Program Analysis

FACT facilitates top-down analysis of system designs through its ability to instantiate many system models (dozens, hundreds, thousands, etc…) by combining the various subsystems and components identified in the Work Breakdown Structure (WBS) of a system. Beginning with the system KPPs (generally threshold and objective performance, although not limited to) and budgetary constraints for procurement and sustainment, FACT compares a virtually unlimited number of potential system designs and then provides mechanisms to filters those instantiations based on user-specified criteria. Perhaps the most direct benefit of using FACT for concurrent exploration of the design tradespace is its ability to eliminate from consideration those system designs with little or no potential for success early in the conceptual design phase. This process applies a top-down filtered Monte Carlo methodology introduced by one of the authors in Ender (2006) to identify feasible options using surrogate models.

A Tool for Resource Constrained Acquisitions

FACT has applicability beyond bottom-up and top-down system design. A team can rapidly conduct an AoA using FACT by loading numerous proposed designs and comparing their cost and performance at the system and subsystem level. When industry submits several designs in response to a Request for Proposal (RFP), decision makers can use FACT to analyze those proposed designs and identify their strengths and weaknesses and the cost drivers for each system. When the offeror of a particular system design asserts superior performance and a price lower than expected, a FACT user can drill down through the system WBS to identify where the proposed design achieves performance and efficiencies beyond the expectations of previously demonstrated technology.

A forensic examination of numerous cancelled acquisition programs for cost overruns and schedule delays reveals system performance requirements and program budgets were not realistically linked by validated cost estimation relationships. The vetting of requirements against the maturity level of current and near-future technology is essential if decision makers are to avoid defining a system composed of "unobtainium."

FACT ARCHITECTURE FRAMEWORK

The FACT framework is based on a series of guiding principles documented, as previously discussed by the authors (O'Neal et al., 2011). This section is a summary of the relevant goals and considerations that shape the FACT architecture.

Architecture Goals

The enterprise data strategy discussed in this paper contains five primary Data Goals listed below:

1. Leverage DoD Net-Centric and M&S Architectural and Data standards to establish a FACT Architecture to allow for discoverable and sharable FACT data and services
2. Allow for a services oriented approach to allow for easy access and approach to new functionality
3. Allow for effective configuration management and promote visibility of FACT services and data via metadata standards
4. Data Sources and Services to provide visibility and pedigree according to VV&A best practices
5. Contain the necessary data/information that will enable users to get educated on capabilities/limitations and technical foundations related to FACT data and services

Architecture Considerations

Development of any web application requires consideration of the same architectural aspects. These nine architectural aspects guided the system design process for FACT:

- Scalability
  service increasing numbers of concurrent users and easily upgrade hardware
- Performance
  provide a near real-time experience to users
- Persistence
  store analyses for future retrieval and avoidance of duplicate computational effort
- Data homogenization
  standards-based approach for data centric communication
- Computational engine
  computational core utilizing data homogenization to ensure data integrity
- Collaboration
  provide interface for real-time collaboration of design team (e.g. Google Docs-like experience)
- Redundancy
  ensuring rapid access to data to handle large volume loads; avoidance of data loss through regular backup procedures
- Client footprint
  near-zero client install to support users with varying system restrictions

SYSTEM CHARACTERIZATION

SysML Backbone

SysML is a general purpose modeling language for systems engineering applications, and supports the specification, analysis, design, verification and validation of a broad range of systems and systems-of-systems. SysML is a profile (dialect) of the Unified Modeling Language™ (UML™), the industry standard for modeling software-intensive systems. This may include hardware, software, information, processes, personnel, and facilities. Its canonical specification provides a standard XML interface to transfer information between toolsets. The four pillars of SysML enable architecture development via rich, interrelated, collaborative system knowledge, as shown in Figure 1.

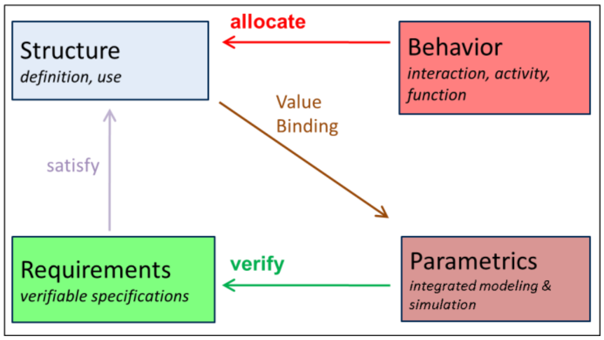

Figure 1. Pillars of SysML

Traditional vehicle analysis, as shown notionally in the left column of Figure 2, involves the identification of requirements and allocating those requirements to the specific Work Breakdown Structure (WBS) elements as defined for the vehicle. This WBS represents the functional and physical decomposition of the vehicle (e.g. engine provides propulsive power). For each decomposed WBS element, an engineer defines certain attributes of interest necessary for analysis. Cost modelers develop Cost Estimating Relationships (CERs) for WBS elements of interest including the aggregation of low level cost estimates to the highest WBS level (e.g. the vehicle itself) for total acquisition cost. The ability to predict total life cycle cost requires a model for operational and support (O&S) cost. By analyzing historical data of similar systems, an engineer derives sizing rules for the new vehicle. Similarly, an engineer derives performance rules based on performance results of similar systems. Finally, a trade study exercises the cost and performance rules/models to provide insights into new designs.

Integration of this traditional process within FACT involves developing individual SysML models of each, as shown in the right column of Figure 2. One SysML model captures requirements, providing traceability to verification. A Block Definition Diagram defines the WBS ensuring the relationship amongst elements of the complex hierarchy is maintained. Parametric diagrams capture the predictive models for cost and performance. These parametric block models capture the fundamental input and output relationships between the models. Note the parametric blocks can then calculate the required values by calling on an equation (such as a regression based surrogate model) or call on an external model. FACT parses the XML representation of the SysML exported by the SysML authoring tool to generate the data the framework requires.

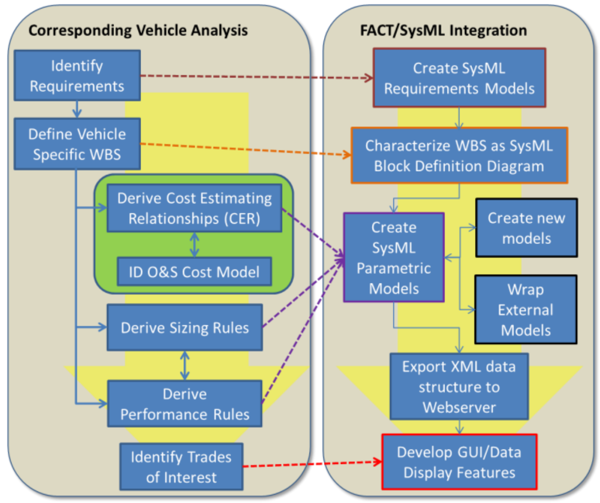

Figure 2. System Characterization in SysML Process Summary

Work Breakdown Structure

Figure 3 provides a SysML Block Definition Diagram (BDD) of a notional WBS excerpt for an armored, amphibious vehicle. Note the highest level element is the vehicle domain, which contains the vehicle (or system), environment, and other support systems with attributes required by the underlying models. Each of level may then be further decomposed, as illustrated in Figure 3 for the engine.

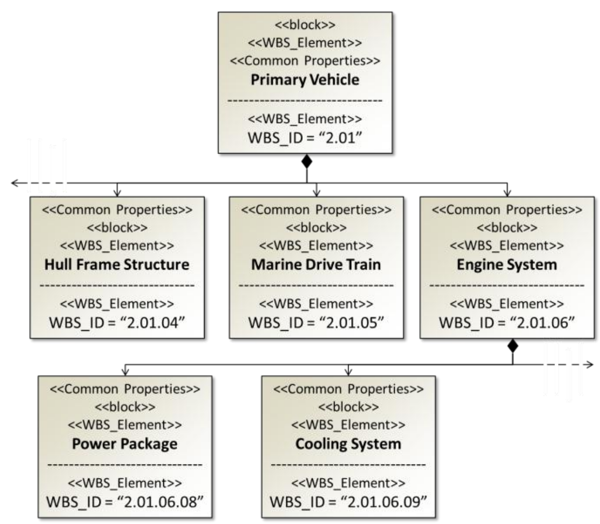

Figure 3. Notional WBS for an Armored Amphibious Vehicle as a BDD in SysML

Parametric Modeling

Figure 4 shows a notional SysML Parametric Block diagram, which captures the basic relationships to calculate maximum speed attainable during swim. Note all this diagram defines is inputs on the left, which here are vehicle weight and engine thrust, an output of water speed on the right, and the parametric block in the middle. The middle block uses those inputs to do any number of options, such as call on a simple regression equation or call on a separate modeling and simulation tool. Instructions for the framework on what code or model to call, given the set of inputs, is contained within the middle block. It is important to note, from the perspective of the framework, capturing the relationships between this and other parametric blocks (e.g. models) is the sole purpose of these SysML diagrams; underlying models are of no consequence.

In the example in Figure 4, one may notice the inputs to the parametric block; vehicle gross mass and water jet thrust, in and of themselves seem to be unlikely user inputs. In fact, they are not direct user inputs but rather values other parametric relationship calculate.

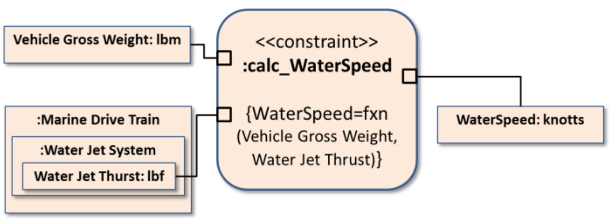

Figure 4. Notional Parametric Block

During the initial parsing of the SysML XML, the parser determines whether each system attribute is either an input or output. Inputs are those attributes which have a direct control on the web user interface. As stated, some outputs are inputs to other models. Certain use cases may call to have a user directly set such a value in place of the parametric output; to handle those scenarios the framework provides specific tools, as discussed in the following section.

TRADESPACE CAPABILITIES

FACT's current web-based tool provides several capabilities for managing systems and exploring the design tradespace. Each tool addresses a common design analysis use case or desired capability; this section describes in detail those capabilities. Additional features are in development for new stakeholders to enhance the overall toolset.

Real-Time Collaboration

The design team considered measures of efficacy and utility for each capability introduced to the FACT tradespace toolset. Each tool provides a chat widget, as seen in Figure 5, to see which other users are currently logged into FACT manipulating a system of the same class/prototype as oneself. As a user modifies an attribute or conducts some analysis, the chat history window provides a brief description of the action to notify a user of the change. In addition, some of the tools ensure a consistent state of all logged-in members by updating control widgets (i.e. slider bars) or output displays based on another users interaction with the toolset.

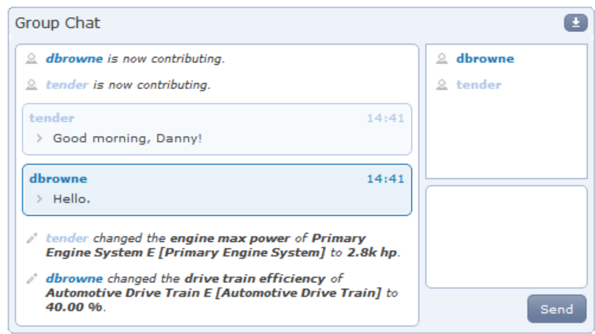

Figure 5. Collaboration Window to Provide Chat and User Action History

Point Solution Configuration

The FACT point solution configuration page allows a user to dial-in a set of attributes and view the various outputs in real-time. With all of the analysis pages, the far left panel provides a view of the entire system WBS in a tree structure. The center panel provides a 3D interactive model of the system (vehicle) at the bottom, as seen in Figure 6. By clicking on predefined regions within the model, the associated WBS element in the tree is highlighted and the top part of the center panel populates with that system's set of user-configurable attributes. For the Point Solution Configuration capability, the center panel attributes are either slider bars or dropdown menus, common to web-interfaces, to allow a user to dial-in the various values.

A stoplight approach provides the metrics output panel on the Point Solution Configuration capability to visually indication whether the outputs are failing to meet, meeting, or exceeding the threshold and objective values, as given in Figure 7. Although the framework is currently designed toward the common threshold/objective requirement approach, it is designed to handle a multi-step or tiered requirement approach. The QFD and Subsystem Scores section provides more details on the use case for this tiered approach.

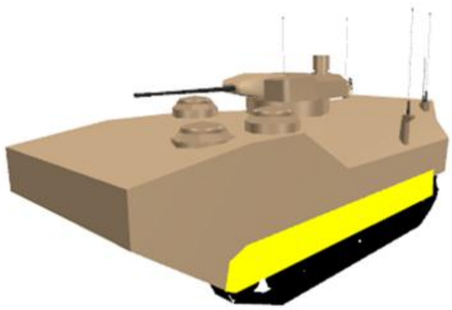

Figure 6. 3D Model of ACV to Provide Interaction with WBS and System Attributes

In the Comparing Alternatives section, an example of the output panel shows the stoplight approach of viewing metrics. Using this output view, those users with such privileges, may modify the threshold and objective values which are propagated to other users in real-time. Users define which metrics the panel displays and how the metrics are grouped; the Managing Requirements section discusses these two capabilities.

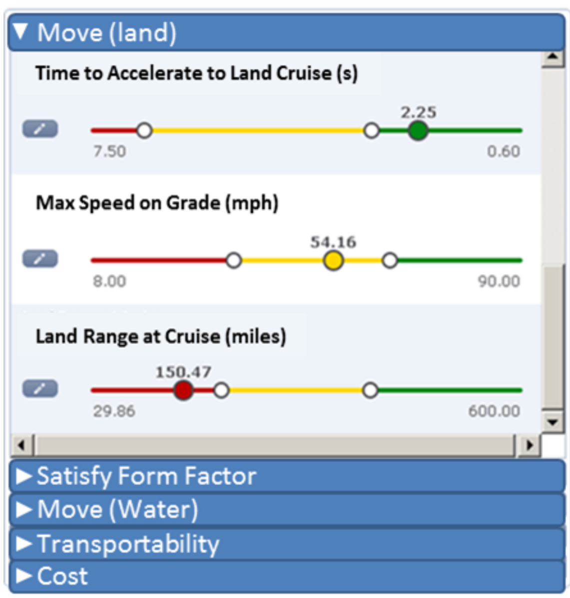

Figure 7. Calculated Metrics as Compared to Threshold and Objectives

Confidence Analysis via Uncertainty Quantification

Uncertainty and risk jeopardize the quality of a system or its ability to meet requirements consistently. A design's ability to achieve specific mission requirements or remain within specific product constraints is not the sole driver of the design. Rather, a robust design process, or one that leads to a design that is least sensitive to influence of uncontrollable factors, is needed to balance mission capability with other system effectiveness attributes. (Zang et al., 2002) describe those design problems that have a nondeterministic formulation, including the field of robust design, as uncertainty-based design. A notional process to quantify uncertainty includes applying probability distributions to any variable containing uncertainty, such as the example Figure 8 provides, then sampling a modeling tool many times via Monte Carlo simulation and plotting the Cumulative Distribution Function (CDF), as Figure 9 displays, of any response metric that varies due to the uncertainty.

Since the use of surrogate models enables quick evaluation of modeling and simulation cases, Monte Carlo investigations comprising hundreds of thousands of runs are conducted within several seconds on a standard desktop computer. This process enables the uncertainty quantification introduced earlier.

One of the authors shows this approach to be valuable in the design of a high speed cruise missile (Ender, 2002), whereby error propagation of calculated metrics using M&S tools provides the ability to quantify the risk associated with achieved target objectives. Engineering "control" variables, such as wing span and fuel tank capacity, are manipulated to maximize metrics of interest, such as range and speed, while minimizing the negative impacts of errors due to uncertainty in the values calculated using those M&S tools. Ender (2004) presents another example of uncertainty quantification based design concept. Here, manipulation of various designs for an air bursting munition allows for minimization of the uncertainty due to warhead fragmentation dispersion. These examples are given to provide the reader with an appreciation for the wide range application of uncertainty based design.

The study discussed in this paper provides a way for a user to apply uncertainty distributions directly to any variable of interest. Random distributions can be assigned to any subsystem-level attribute and sampled to explore the tradespace. Figure 8 shows an example of applying a normal distribution to engine specific fuel consumption, with a mean of 0.30 lbm/hp-hr and standard deviation 0.07 lbm/hp-hr. The web-browser interface allows a user to dial in a mean and standard deviation directly. Note the user can select one distribution from a variety of options: normal, triangular, uniform, discrete. Specific value properties are displayed when a particular distribution is selected;

for example, the interface displays mean (μ) and standard deviation (σ) for a normal distribution. Minimum and maximum values are displayed for the uniform distribution; minimum, mode, and maximum are editable for the triangular distribution.

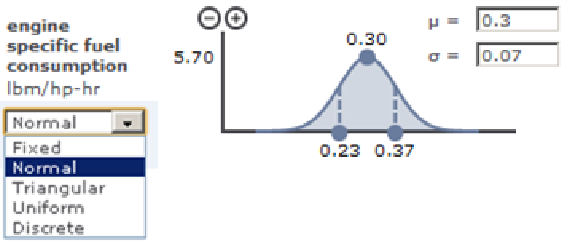

Figure 8. Normal Distribution Applied to Engine Specific Fuel Consumption.

The Cumulative Distribution Function (CDF) displayed in Figure 9 can be analyzed to show that, based on the uncertainty in the system-level engine specific fuel consumption given in Figure 8, there is a 68% likelihood of achieving the Total Operational and Support (O&S) Cost threshold and 10% likelihood of achieving the objective. Note the output distribution shape Figure 9 provides is a function of the resultant effect of the input distribution shapes applied to any variable, whether to the variable given in Figure 8 or any other variable in the model, and the sampling of the relevant M&S tools. This process may be dynamically applied to any metric calculated via surrogate M&S; it is this ability which demonstrates the specific value the FACT framework offers.

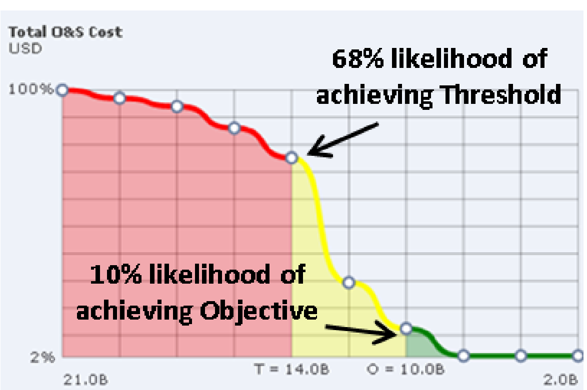

Figure 9. CDF of Total O&S Cost

Alternative Exploration via Scatter Plotting

While the confidence analysis feature provides insight into the likelihood of achieving a program's thresholds or objectives, the scatterplot feature displays many single solutions, also generated by randomly sampling system attributes. The framework allows top-down filtering of these random system-wide solutions against the defined requirements. Figure 10 provides an example of one scatterplot comparing max water speed at one foot significant wave height with respect to the water jet thrust. Note, however, these points were generated by varying a large number of other variables effecting overall system mass, cost, reliability and other performance factors.

By applying a maximum O&S cost filter against this plot, each point (e.g. system solution) is marked as a pass or fail: those points which remain colored fall within the valid region while gray solutions exceed the O&S cost filter. FACT provide multivariate scatter plotting, tying the various dimensions together so as a point is selected in one plot, all corresponding points in other dimensions are highlighted. Finally, users can chose to instantiate selected points which become vehicles for comparison against a baseline in the framework's other tools.

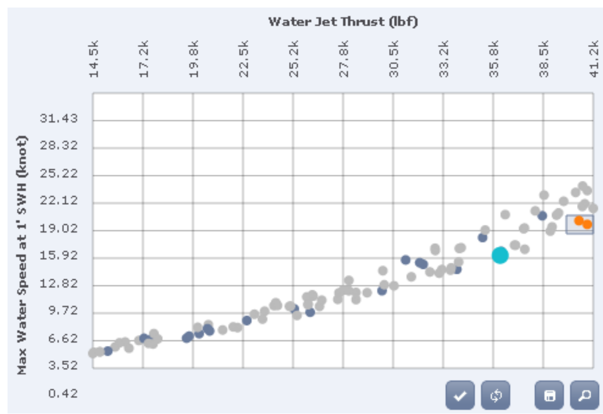

Figure 10. Single Scatterplot Filtered Against Total O&S Cost

Managing Requirements and System Scores

All programs begin by defining a set of requirements the eventual solution needs to satisfy. Often, these are grouped into absolute minimums (e.g. thresholds) and desired, but room for compromise, objectives. A comparison tool, discussed below, provides a standard stoplight chart output. Additionally, FACT assists decision makers in comparing a set of solutions by asking them upfront questions about how they value certain requirements, and, for each requirement, the threshold and objective value.

Currently, users can associate a requirement with any calculated value (i.e. an underlying model must calculate some numeric value). The framework requires the user to not only provide a threshold/objective pair, but also an overall requirement importance and threshold weight. For example, an importance of nine denote Key

Performance Parameters while ones or threes signify less important requirements, in the eyes of a decision maker. The threshold weight signifies how important achieving the threshold value is with respect to the objective. Using these various user-defined values, weighted sums provide vehicle-wide scores. First, the calculated vehicle metrics are normalized and a sub-score for that metric is determined by interpolating with the threshold/objective relative weighting. The total score is the weighted sum of requirement importance with the vehicle's related sub-score. By requiring the user to provide the qualitative weighting information in advance and in an environment isolated from viewing physical solutions, a user's inputs are less swayed by final scoring expectations.

The novelty of FACT is its ability to manage multiple sets of requirements for various scenarios or decision makers as well as update the vehicle scores in real-time based on swapping out subsystems (i.e. engines) or manipulating individual attributes. Rapidly, a user can realize the significance of upgrading an engine with respect to the qualitative scoring laid out in advance. Currently a user can compare any number of vehicle instances against a single scenario; future capabilities will allow multiple vehicles to be compared against multiple requirement sets.

QFD and Subsystem Scores

To capture qualitative importance or relevance of a subsystem FACT employs a Quality Function Deployment (QFD) which maps requirements to system attributes. Every requirement is mapped to every system attribute with a value representing its qualitative effect. For example, max speed on grade is highly dependent on engine horsepower, therefore a value of 9 (high impact, positive correlation) may be assigned. Gross vehicle mass, however, may have a value of -1 (low impact, negative correlation) with tire mass since achieving the low mass is negatively correlated, but only slight, by tire mass.

The QFD mapping provides a means to estimate qualitative scores for system instance (such as the relative score for a specific engine). Like with the vehicle score described earlier, system attributes are normalized and a dot product between the design parameter and the attribute score provides a relative score for each system instance with respect to all systems. For example, the score for a specific engine is relative to all components including the tires and armor panels. Figure 11 displays the component select dialog from the vehicle manager, showing the list of available engines as well as the relative score for each. Using the vehicle manager tool, users can update subsystem attributes in real-time and identify the systems which could have a positive impact on the overall vehicle's performance. Additionally, by reviewing an entire system's set of instances, since scores are normalized across the entire vehicle's WBS, insight into which subsystems are of highest importance can be determined. Note different methods are used for calculating vehicle scores and subsystem scores, so although high scoring systems compose a vehicle, there is no direct relationship between a vehicle's subsystem scores and its own score with respect to the requirement importance calculation.

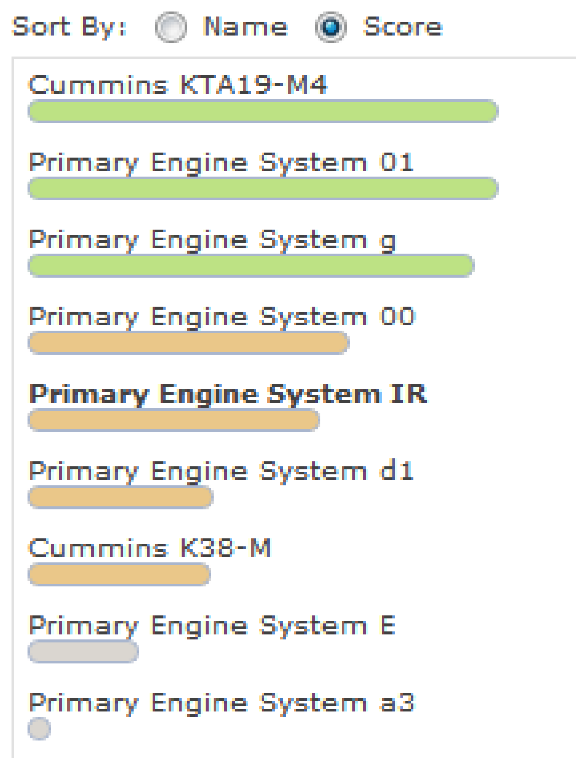

Figure 11. Display of Engine Selection Panel with Relative Score

Comparing Alternatives

A final feature FACT provides is two views for comparing vehicle solutions. The first view is a standard stoplight chart which lists all requirements as rows and all vehicles as columns. This tool utilizes the collaborative capability by updating the requirement thresholds and objectives, vehicle output metrics, and requirement importance in real-time to ensure the stoplight chart is up-to-date. Figure 12 displays an example of the stoplight chart. In addition, a view similar to a radar plot uses filled area of a honeycomb-shaped object (where each triangle is an output metric) to indicate if the vehicle is achieving a threshold or objective requirement value.

Tying this view in real-time to the other configuration pages allows a decision maker to monitor how subject matter experts' inputs for requirements or system parameters, dedicated to different subsystems, affects the overall value of vehicle alternatives.

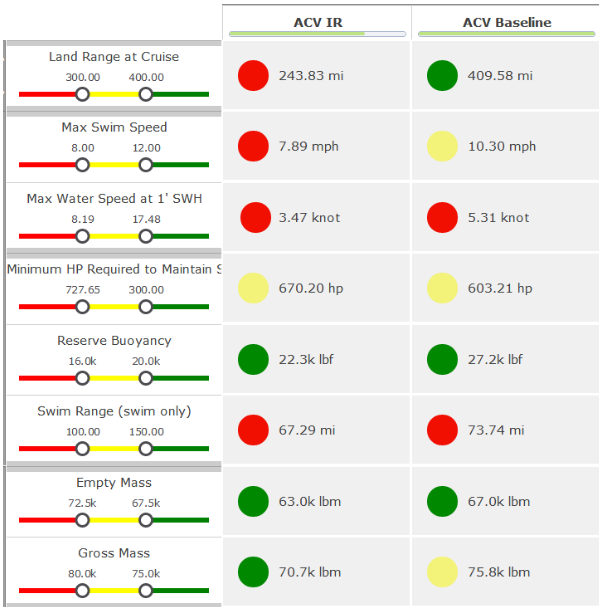

Figure 12. Alternative Comparison Chart

CONCLUSIONS

Acquisition professionals are often forced to make decisions with little information, though those decisions may have far reaching implications on the later life cycle stages of a system. FACT introduces an open architecture web services based environment that enables the interconnecting of models to provide a rapid exploration of the design tradespace in support of systems engineering analysis. FACT is government owned, model agnostic, and capable of linking disparate models and simulations of both government and commercial origin through the application of community established data interoperability standards. The methodology was introduced to characterize not only the system, but also the system engineering process in SysML. This paper described the utility of using FACT to achieve near real-time analysis for exploring the design parameter trades that affect the overall performance, reliability, and cost of a system design, all through a collaborative web-browser framework.

The authors plan to extend the various capabilities introduced in this paper. For example an open source collaborative development framework may be created, enabling users to develop and integrate new functionality into the FACT framework. This includes developing a standard Python based development environment for updates/improvements/enhancements to the underlying business logic (algorithms, calculations, etc…), and standard JavaScript environment to develop custom visualizations. Additionally, the authors envision extending FACT to enable configuration management of the underlying models, modeling and simulation tools, offline databases, and anything else used to build the concepts which are represented in the FACT framework. This includes building on open source distributed version control software and tools.

Finally, the authors hope to show the application of the methods and processes inherent within FACT beyond ground vehicles. The authors plan to publish results related to application of FACT to modeling delivering energy to a Marine Air Ground Task Force, showing the various trades between conventional and renewable sources of energy. Additionally, the authors plan to implement instantiations of other systems to include unmanned aerial systems and related weapon systems.

ACKNOWLEDGEMENTS

The authors wish to thank the Marine Corps Systems Command for funding the FACT framework, and Program Manager Advanced Amphibious Assault for additional vehicle specific funding and data. Additionally, the authors thank Justin Pearl and Joe Bricio of NSWC Dam Neck for being part of the FACT architecture development and deployment team, and Rick Tillotson and Lorri Bennett of Penn State University's Applied Research Laboratory for development of cost models.

REFERENCES

Assistant Secretary of Defense (Networks and Information Integration) and DoD Chief Information Officer (2007). DoD Information Assurance Certification and Accreditation Process (DIACAP), Department of Defense Instruction 8510.01.

Assistant Secretary of Defense (Networks and Information Integration) and Chief Information Officer (2009). Clarifying Guidance Regarding Open Source Software (OSS), Department of Defense Memorandum.

Ender, T.R., McClure, E.K., Mavris, D.N. (2002). A Probabilistic Approach to the Conceptual Design of a Ship-launched High Speed Standoff Missile. *Proceedings of the AIAA 2002 Missile Sciences Conference*, Monterey, CA.

Ender, T.R., Mavris, D.N., Massey, K.C., Holder, E.J., O'Leary, P.A., Smith, R.A. (2004). Predicting the Effectiveness of an Air Bursting Munition Using Uncertainty Quantification. *Proceedings of the AIAA 2004 Missile Sciences Conference*, Monterey, CA.

Ender, T.R. (2006). A Top-Down, Hierarchical Approach to the Design of an Air Defense Weapon. PhD Dissertation, Georgia Institute of Technology.

Ender, T.R., McDermott, T., Mavris, D. (2009). Development and Application of Systems Engineering Methods for Identification of Critical Technology Elements During System Acquisition. *Proceedings of the 7th Annual Conference on Systems Engineering Research*, Loughborough University, United Kingdom.

Forrester, A., Sóbester, A. and Keane, A. (2008). *Engineering Design via Surrogate Modelling: A Practical Guide*, West Sussex: John Wiley & Sons, Inc.

Luskin, P. and Berlin, R. (2010). Systems Engineering Methodology for Fuel Efficiency and its Application to the TARDEC Fuel Efficient Demonstrator (FED) Program. *Proceedings of the 2010 NDIA Ground Vehicle Systems Engineering and Technology Symposium*, Dearborn, MI.

Maier, M.W. (2009). The Art of Systems Architecting, Third Edition. Boca Raton, FL: CRC Press.

McDermott, T.A., Ender, T.R., Bollweg, N. (2011). Collaborative Development of Systems Architecting Design Rules. *Proceedings of the 14th NDIA Systems Engineering Conference*, San Diego, CA.

O'Neal, M., Ender, T.R., Browne, D., Bollweg, N., Pearl, C.J., Bricio, J.L. (2011). Framework for Assessing Cost and Technology: An Enterprise Strategy for Modeling and Simulation Based Analysis. *Proceedings of MODSIM World 2011 Conference and Expo*, Virginia Beach, VA.

O'Neal, M., Ender, T.R., (2011). Framework for Assessing Cost and Technology: Integrating M&S into Ground Vehicle Acquisition. *Proceedings of the 14th NDIA Systems Engineering Conference*, San Diego, CA.

Xiaoqing, L., Raorane, S., and Leu, M.C. (2007). *A Web-based Intelligent Collaborative System for Engineering Design*, in *Collaborative Product Design and Manufacturing Methodologies and Applications*, Li, W.D., Ong, S.K., Nee, A.Y.C., and McMahon, C. (Eds.), London: Springer-Verlag.

Zang, T. A., Hemsch, M. J., Hilburger, M. W., Kenny, S. P., Luckring, J., Maghami, P., Padula, S. L., and Stroud, W. J., (2002). Needs and Opportunities for Uncertainty-based Multidisciplinary Design Methods for Aerospace Vehicles. Technical Report TM-2002-211462, NASA Langley Research Center, Hampton, VA.