Augmenting Model-Based Design with the HEAT Human System Integration Advisor

Marc Abrams
Harmonia Holdings Group, LLC
Blacksburg, VA 24136
mabrams@harmonia.com

Michael O'Neal
United States Marine Corps
Quantico, VA 22134
michael.oneal1@usmc.mil

ABSTRACT

While automated design methods based on model-driven design have produced enormous strides in efficiency and reduced lifecycle costs, human factors experts have traditionally performed separate but complementary analysis not accommodated by modeling tools. This paper describes a major effort funded by the U.S. Marine Corps to integrate human modeling into model-driven design. Example applications are to analyze the crew space in a vehicle, aircraft, tank, etc. for issues ranging from ability to accommodate individuals in a range of sizes, account for gear worn by warfighters, examine both static position and dynamic movement of human occupants, and identify safety issues.

We describe the Human Systems Integration (HSI) Engineering Analysis Tool, or HEAT. HEAT provides advice for engineers using material-design software on how their design choices affect HSI domains. While there are eight HSI domains—Environment, Habitability, Human Factors Engineering (HFE), Manpower, Personnel, Safety, Survivability, and Training—that HEAT's design can accommodate, we discuss here our recent work on Human Factors Engineering, Survivability/Safety, and Habitability. HEAT is designed in a modular fashion to integrate with the U.S. Marine Corps (USMC) Framework for Assessing Cost and Technology (FACT). We present a scenario of use for HEAT illustrating the derivation of potential HSI risks for a crew space, and the use of HEAT to monitor and detect changes in the engineering model that are made to address another concern (corrosion), but inadvertently have an HSI impact.

ABOUT THE AUTHORS

Marc Abrams is Chief Technical Officer of Harmonia Holdings Group, LLC. He has created the concept for several Human System Integration tools, including the HEAT system in this paper, the GUITAR (Graphical User Interface Test, Assess, and Remediate) system for user experience evaluation, and the Impact Modeler tool to model the usability of the designs of consoles used by warfighters when those designs are in their earliest stages. Dr. Abrams leads the implementation team for the HEAT prototype described in this paper. He received a B.S. in Electrical Engineering, M.S. in Computer Science, and a Ph.D. in Computer Science from the University of Maryland at College Park. He was a Post-Doctoral Researcher at Stanford University in Computer Science. Dr. Abrams was a Visiting Scientist at the IBM Zurich Research Laboratory in Communication Networking, and an Associate Professor of Computer Science with tenure at Virginia Tech. He received the 2013 CTO of the Year Golden Bridge Award.

Michael O'Neal is Modeling and Simulation Lead for the U.S. Marine Corps Systems Command (MARCORSYSCOM). Mr. O'Neal was commissioned an Ensign, United States Navy in 1975 after being awarded a B.S. in Physics by Miami University. Leaving active duty in 1979, Mr. O'Neal joined the Naval Reserve as an Engineering Duty Officer retiring with the rank of Captain. Mr. O'Neal holds an M.S. in Engineering Administration from The George Washington University and a Masters in Public Administration from Indiana University. His career includes service in both private industry and civil service. While in civil service, he has held various assignments including Deputy Program Manager for Directed Energy and Electric Weapon Systems, Director, Weapons and Weapon Support systems, where he co-chaired the Coalition Maritime Missile Defense working group for Modeling and Simulation (M&S); Director, Science and Technology Transition and Major Acquisition Program Manager for the Performance Monitoring Training and Assessment Office (PMS 430). In 2010, he joined the Marine Corps Systems Command as their Director, Modeling and Simulation. Mr. O'Neal's M&S Team was recognized in 2015 with the Secretary of the Navy's Innovation Excellence Acquisition Team of the Year Award.

Augmenting Model-Based Design with the HEAT Human System Integration Advisor

Marc Abrams
Harmonia Holdings Group, LLC
Blacksburg, VA 24136
mabrams@harmonia.com

Michael O'Neal
United States Marine Corps
Quantico, VA
michael.oneal1@usmc.mil

MOTIVATION

Systems that have a human component carry significant risk to the success of a program if Human Systems Integration (HSI) impacts are not detected and resolved as early as possible in the system lifecycle. Usability problems that are not exposed early enough to address in design changes can result in pain points of reduced human performance (e.g., due to fatigue or too many steps to load a weapon), higher training costs to teach non-intuitive procedures, and even loss of life. Currently the U.S. Marine Corps (USMC) lacks methods to consider HSI during engineering as they do other impacts (form factor, cost, safety, etc.). Consequently, the baseline approach for Department of Defense (DoD) acquisition programs in charge of bringing new systems from design to fielding is to use manual methods and/or niche tools that:

- Silo usability data in their own formats that lack traceability to design changes
- Do some evaluation late at the stage of test and evaluation (T&E), such as HSI expert evaluation or usability testing, when few program dollars remain to fix major usability issues
- Do not perform usability analysis at all (e.g., an attitude arising when a program reasons that if they have no money to fix problems, then why spend money identifying them)

This paper describes a system called the HSI Engineering Analysis Tool (HEAT), whose goal is to overcome the limitations above by integrating HSI analysis into design and development processes early and often throughout a program's lifecycle. That integration is achieved by coupling HEAT to a modeling and simulation tool. The coupling of HSI analysis to modeling and simulation is a powerful combination because it allows HSI considerations to be evaluated and HSI risks to be identified in the earliest stages of design. Furthermore, it gives credibility to HSI experts participating in the early design phase so that they may influence fundamental system design decisions to benefit human users, rather than providing input later in the lifecycle when few options remain.

HEAT implements of a set of algorithms to identify HSI impacts from a physical model (e.g., a CAD or SysML model containing geometric relationships between components). Examples include determining if a human of certain size, gender, limb length, etc. "fits" into a physical model of a crew space, and measuring if the field of vision of an operator allows viewing of all displays within parameters. We use guidelines and parameters from various standards such as MILSTD1472 revision G, ASTM 1166, and Section 508 for accessibility.

HEAT is intended to be integrated with a modeling tool used by engineers who may have little or no formal training in HSI. HEAT will then use the user interface style of the modeling tool, so that an engineer user does not have to learn how to use a second interface to access HEAT. Our vision of HEAT is to augment such modeling tools with an HSI "advisor" that transparently runs in the background, and alerts the engineer of the need to perform HSI checks. For model driven design tools, HEAT's execution can be triggered by a change in a model. For example, if a physical component in the crew space is relocated, HEAT can recompute whether a human operator will still fit the space. This example is illustrated later in a sample scenario.

Furthermore, HEAT assists Engineers not trained in HSI methods to understand impacts of their design choices on HSI in simple terms to highlight when there is a risk and the source of the risk. Once a risk is identified, the Engineer will then know it is time to engage an HSI expert to apply human guidance that no machine can replace, and help the user address the risk. Therefore, a secondary goal of HEAT is to seamlessly integrate HSI experts into engineering teams via a "just in time" paradigm (e.g., upon discovery of a risk). One final goal is to apply lessons from incidents (as serious as loss of life) in deployed systems to future designs, and alert users of new systems to the potential for those incidents to recur.

Application of HEAT to the USMC FACT System

The principles discussed above are illustrated in this paper, where we present an integrate of HEAT with a modeling tool developed by the USMC: the Framework Assessing Cost Technology (FACT) system. FACT was created to allow users to concurrently consider design trades and visualize a system's potential expense alongside its performance, reliability, and other factors (Ender et al, 2015). Although FACT aids impact analysis for engineering, such as the impact on cost, it does not consider HSI impacts.

HEAT can to help FACT users identify, pinpoint, track, and resolve HSI impacts early and throughout the engineering lifecycle as they use the FACT system. The benefits are to save acquisition program ownership costs and improve usability and safety for system operators.

OVERVIEW of HEAT

HEAT informs FACT users who are designing a system (e.g., a military vehicle) in near real time of how their design decisions impact humans. HEAT integrates human models (e.g., to overlay a physical model of a human next to a vehicle image in FACT's Model Viewer). HEAT helps expert and non-expert users to trace design quality and potential risks across one or more of the eight HSI domains: Environment (e.g., shock, vibration, extreme temperatures, or pressurization), Habitability (e.g., workspace parameters, working conditions, personal space), Human Factors Engineering (HFE, including biometrics, ergonomics, and user-experience parameters), Manpower (e.g., workload type and demand parameters), Personnel (e.g., user types, skills, abilities), Safety (e.g., personnel and system safety along with security parameters), Survivability (e.g., escaping from harm, fatigue, stress), Training (e.g., type, frequency, and results parameters).

HEAT is designed to enable FACT users—who may know little about HSI—to identify HSI deficiencies, risks, and metrics for a wide variety of engineering systems across the DoD. HEAT uses model-based design. Each time an engineering model in FACT is updated, either in System Model (SysML) or a Computer Aided Design (CAD) model, HEAT traces from the engineering model to check HSI rules and detects HSI impacts, providing the following functionality:

1. HEAT serves as a constantly vigilant advisor that scrutinizes each change in the maturing design from a set of HSI rules and lessons learned. HEAT advises the designer when a design change adversely impacts HSI (e.g., design change has created a safety problem, will lead to poor ergonomics, will change the habitability of a crew space, etc.).

2. HEAT aids novice and expert users in detecting HSI risks (automatically or manually), collaboratively tracking and resolving these risks, reusing their knowledge for solutions to future problems, and adding to HEAT's libraries to increase impact detection, which all work together to ensure reduction in HSI risk across USMC systems.

3. HEAT extends the model-based engineering approach underlying FACT to permit the HSI impact analysis in #2 above. Also, by leveraging models, HEAT is designed to integrate into FACT so that users see one unified system.

4. HEAT tailors its User Interface for ease of use depending on the role of the logged in user, specifically Engineer versus HSI Expert versus Program Manager. The roles are extensible. We have striven in the HEAT User Interface to explain results in layman's terms so that the user need not be trained in HSI.

5. HEAT provides role-tailored functionality to manage HSI requirements, integrate and manipulate the human model with the system model, and visualize the scope and quality of HSI and its domains (e.g., Habitability, Safety, etc.).

HEAT also addresses needs stated by the DoD Modeling & Simulation (M&S) discipline. These include: "Improve the military operational effectiveness and suitability usage," and "increase the quality, interoperability, and supportability of fielded capabilities." HEAT promotes data interoperability by tracing between HSI data artifacts and models in HEAT. It enables data to be understandable, which HEAT accomplishes by providing to users simply worded alerts of HSI problems, even for novice users that have no exposure to HSI. HEAT can support satisfaction of these goals through the creation of the HSI module, but more so through creation of highly usable interfaces, reuse of technology and functionality, presenting meaningful and usable data, and through various suggestions for enhancement to the existing FACT interfaces.

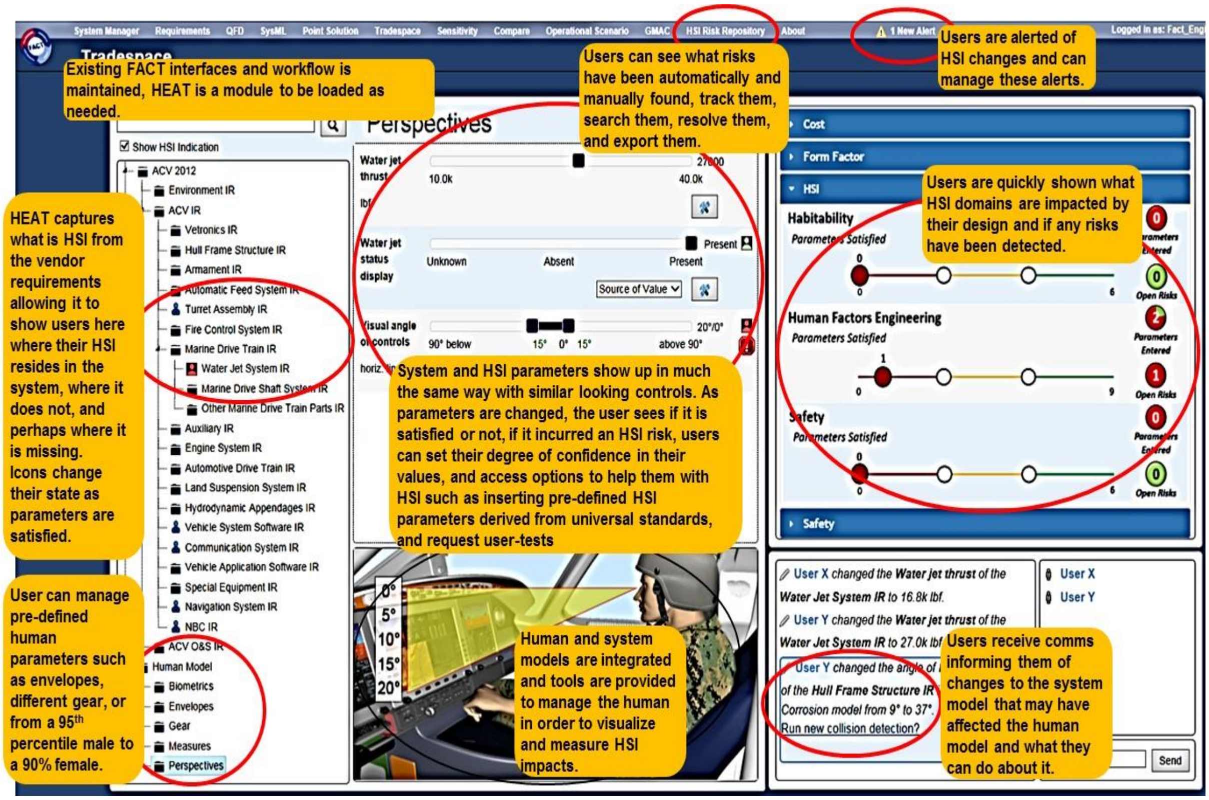
Summary of HEAT's Capabilities

We summarize next the capabilities of a prototype version of HEAT that we have completed by means of the illustration in Figure 1. The yellow overlays in Figure 1 describe ways in which an Engineer, an HSI Expert, or a Program Owner can use FACT with the HEAT module to identify, track, and resolve HSI impacts.

SCENARIO OF USING HEAT

We discuss in detail next a scenario of using HEAT to analyze HSI impacts of variants of a model of an ACV with a Water Jet IR submodule. We consider the roles of an Engineer, who is doing the design and has limited experience with HSI, an HSI Tester who at one point in the scenario will be enlisted by the Engineer to conduct tests with end users/human subjects that are the Operators who would use the system once implemented and fielded. The scenario here can be conducted at an early stage in the systems engineering lifecycle when the first SysML or CAD models have been created that include human interface points and crew spaces.

Before the scenario below starts, a FACT user will have entered a Work Breakdown Structure (WBS) for the system under design (tree structure on left of Figure 1), the requirements that the system must implement (which are associated with the WBS), and associated their SysML or CAD models with FACT. The starting point in our scenario is for the HEAT user to then identify which requirements are HSI related (e.g., crew space, gear worn by Operator) in the WBS. In our scenario, we first evaluate if a given design variant, as is, satisfies HSI requirements. This will reveal that the angle of view of the Operator in the crew space must exceed HSI recommended angles (see bottom center of Figure



Existing FACT interfaces and workflow is maintained, HEAT is a module to be loaded as needed.

Users can see what risks have been automatically and manually found, track them, search them, resolve them, and export them.

Users are alerted of HSI changes and can manage these alerts.

HEAT captures what is HSI from the vendor requirements allowing it to show users here where their HSI resides in the system, where it does not, and perhaps where it is missing. Icons change their state as parameters are satisfied.

Users are quickly shown what HSI domains are impacted by their design and if any risks have been detected.

System and HSI parameters show up in much the same way with similar looking controls. As parameters are changed, the user sees if it is satisfied or not, if it incurred an HSI risk, users can set their degree of confidence in their values, and access options to help them with HSI such as inserting pre-defined HSI parameters derived from universal standards, and request user-tests

User can manage pre-defined human parameters such as envelopes, different gear, or from a 95th percentile male to a 90% female.

Human and system models are integrated and tools are provided to manage the human in order to visualize and measure HSI impacts.

User X changed the Water jet thrust of the Water Jet System IR to 16.8k lbf.

User Y changed the Water jet thrust of the Water Jet System IR to 27.0k lbf

User Y changed the angle of of the Hull Frame Structure IR Corrosion model from 9° to 37°. Run new collision detection?

Users receive comms informing them of changes to the system model that may have affected the human model and what they can do about it.

Figure 1: Capabilities of HEAT in Gold Call-out Boxes

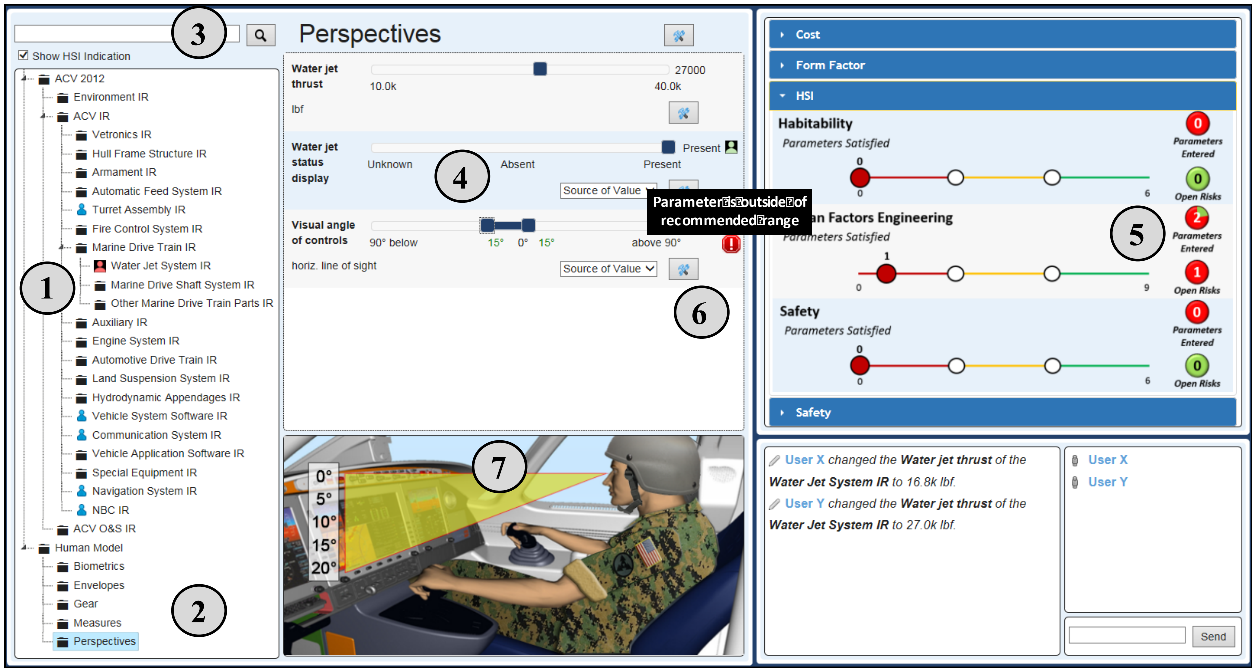
1); a display is located too low in the crew space. Then we consider a situation in which an Engineer changes the design (bottom right of Figure 1). Specifically, the Engineer will also use a Corrosion Model in FACT, and due to a desire to protect certain parts of the system from corrosion will make a change in their geometry (e.g., changing the angle of a part to allow water to roll off rather than pool). At that point in the scenario, HEAT will, from monitoring changes to the SysML/CAD model, detect the change in part geometry and compute a collision model that will show that the part will impact the Operator's helmet. Without HEAT this change may have been undetected until a late stage in the system lifecycle.

Capture HSI Requirements

The FACT system is strong on using requirements with traceability to designs. FACT imports vendor models that link requirements and their satisfaction criteria to those system components that realize the requirements. FACT's ability to help users assess the impact of changes provides a foundation for HEAT to analyze the HSI impact of changes starting with requirements.

HEAT includes a library of keywords to help identify which requirements are HSI-related and to which HSI domain they belong. Users can add keywords to this library to help capture more requirements as FACT and HEAT mature. Once requirements are tagged as HSI, the normal FACT process of defining them as parameters ensues using screens already provided by FACT.

With parameters defined, user-entered values per HSI parameter can be compared against their accepted criteria (e.g., acceptable values or ranges of values). Each HSI requirement that a user can document as being satisfied (i.e., meeting the criteria) in HEAT will reduce the overall risk to HSI for a design.



User X changed the **Water jet thrust** of the **Water Jet System IR** to 16.8k lbf.
User Y changed the **Water jet thrust** of the **Water Jet System IR** to 27.0k lbf.

Figure 2: FACT Workspace updated with User Interface Components for HEAT

Figure 2 illustrates a main user interface for HEAT that will be discussed throughout the scenario below. The content by Circle #1 illustrates system components that have one or more HSI requirements traced to them using a small blue icon of a person. (The blue icons are to the left of various lines in the tree view in the panel by #1, such as "Turret Assembly IR" and "Vehicle System Software IR".) Viewing system components is controlled through a toggle checkbox at the top of the WBS pane. As requirements become satisfied, the color of the icon changes to green.

A WBS could be quite long when expanded, depending on the complexity of the system and the degree to which the system has been decomposed. Furthermore, users looking to consider HSI impacts in their design may have a specific goal in mind, such as user safety. Therefore, the user interface (Circle #2 in Figure 2) provides an ability to search through the WBS and find parts using keywords and partial text matching.

Incorporating a Human Model

The FACT system has a capability to import models of a system under design (e.g., an amphibious vehicle), such as SysML or a CAD model. HEAT augments FACT with an ability to superimpose a model of a human into such models, if the models specify size and geometry, because HSI is concerned with the physical arrangement and dimensions of controls, crew spaces, etc. HEAT augments the visualization capability of FACT, as shown in #7 of Figure 2 to reveal human/machine collisions. By "collision" we mean that the physical volume of space required for a human Operator and their gear intersects some part of the physical components of the system under design; in layman's term the Operator "doesn't fit." For the Operator to "fit," the operator and gear cannot intersect with any of the system's components. The HEAT user can also apply the viewing pane capabilities in FACT such as zoom, pan, wireframe mode, and a distance measuring tool.

To implement this superposition of a human model in the SysML/CAD design, HEAT provides an open application programming interface (API) to connect to a human modeling tool (or HMT). The use of an open API continues the approach used in FACT of allowing 3rd party tool integration. Some examples are Jack (Siemens), Human Builder (CAD/CATIA), and Santos Pro (by Santos Human Inc.). We have found Jack to be the most capable. HEAT is agnostic to the HMT used. HEAT uses the API to import a 3D model of a human of specified proportions. The user can define the gender, size (e.g., 5th percentile, or 95th percentile), and pose of the human model within the HMT. HEAT also imports human attributes such as reach envelopes, joint movement, gear that is worn, etc.

Figure 3 illustrates the design for interfacing with an HMT. The FACT tool with HEAT is in the center and is capable of exporting some dimension information in several formats (e.g., JT (Jupiter Tesselation), CAD, SysML. The user of the HMT (left of Figure 3) then creates a human model with appropriate pose and scale for the model. HEAT will then import the resulting HMT files and display the intersection with the system under design in a FACT integrated visualization in the viewer (right of Figure 3).

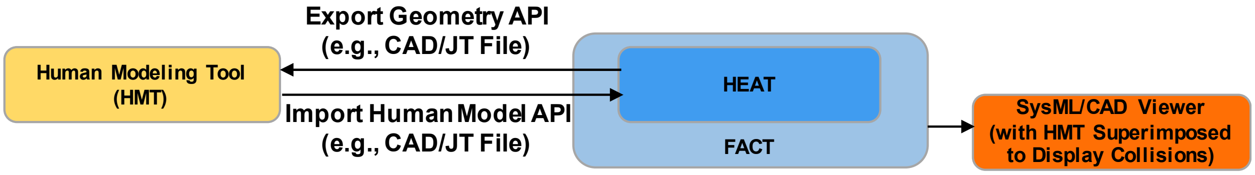


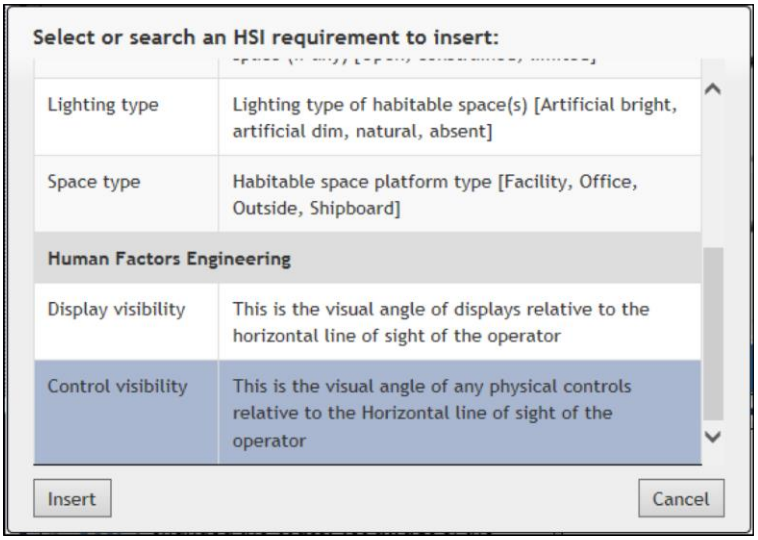
Figure 3: Interface Between HEAT and HMT to Derive Human Geometry for Viewer

Engineer Manipulates HSI Parameters and the Human Model

The existing FACT system on which HEAT is built supports the review and entry of system parameter values for an engineering model being designed. HEAT leverages this existing functionality and enhances it for the purposes of viewing and entering values for HSI parameters. The content shown around #4 in Figure 2 is an example of an HSI parameter that was defined from an HSI requirement tagged by HEAT in the Water Jet System IR subsystem. The Engineer can use the slider just like for system parameters, such as the first parameter shown.

When entered values meet the criteria set by the underlying requirements, HSI is satisfied for the related parameters, and as shown in the content around #5 in Figure 2, the number of parameters satisfied increases. When entered values do not meet the criteria, exemplified by the bottom-most parameter shown, it creates an HSI risk that is detected automatically by HEAT (and logged in an HSI Risk Repository that HEAT provides); the number of HSI risks being tracked as '1' within a red circle at the right of the Human Factors Engineering domain. Also note that the color of the HSI person icons change in the WBS and the parameter panes (green is satisfied, red is unsatisfied).

HEAT keeps track of the HSI parameters for a design because of the requirements tagging and definition process. Thus, the summary of HSI at the top right of Figure 2 (see #5) is a status of, at least in this example, all 9 HFE parameters even though only 2 of them pertain to the Water Jet System IR. Users can hover over the parameter tracker (2) to see a synopsis of all 9 parameters (2 of which have been entered in this example). Three HSI domains shown at the right in Figure 2 (Habitability, Human Systems Engineering, and Safety) were relevant to the example. Other designs analyzed in FACT may have all eight domains, some may have fewer; it just depends on which domains are applicable to the requirements set specific to each design variant.



Select or search an HSI requirement to insert:

Lighting type	Lighting type of habitable space(s) [Artificial bright, artificial dim, natural, absent]
Space type	Habitable space platform type [Facility, Office, Outside, Shipboard]
Human Factors Engineering	
Display visibility	This is the visual angle of displays relative to the horizontal line of sight of the operator
Control visibility	This is the visual angle of any physical controls relative to the Horizontal line of sight of the operator

Figure 4: HSI Requirements Selection Dialog

We are investigating an enhancement to the parameters pane: confidence level. Each parameter, whether a system or HSI parameter, should support the selection of how confident the user is in the values they have entered. This is particularly important when values have been estimated, when the values that have been entered are predicted, and/or when users are entering values for parameters that they do not have expertise in. The selection of confidence could be skipped, but if used, a simple selection can be made by the user from a dropdown of three choices: Low, Medium, or High. Since confidence itself is subjective, so too are the choices. In other words, each user determines how confident they are in the values based upon how they derived each parameter's value. A Program Manager can use overall confidence scores to help them evaluate designs, which may otherwise be very similar.

Adding HSI Parameters to Increase the Fidelity of HSI Impact Detection

If vendor requirements do not include HSI requirements or their requirements are not detailed sufficiently (i.e., too high level), HEAT includes a library of 92 pre-defined HSI requirements applicable to any type of system and Operator. The parameter noted by #6 in Figure 2 (Visual Angle of Controls) is an example of such a parameter.

Users can add requirements from a control widget provided for each parameter; the resulting dialog is shown in Figure 4. The Engineer (or any user) can search through this HEAT library and select relevant requirements (e.g., visibility of controls to an operator) to be added to the currently selected system component.

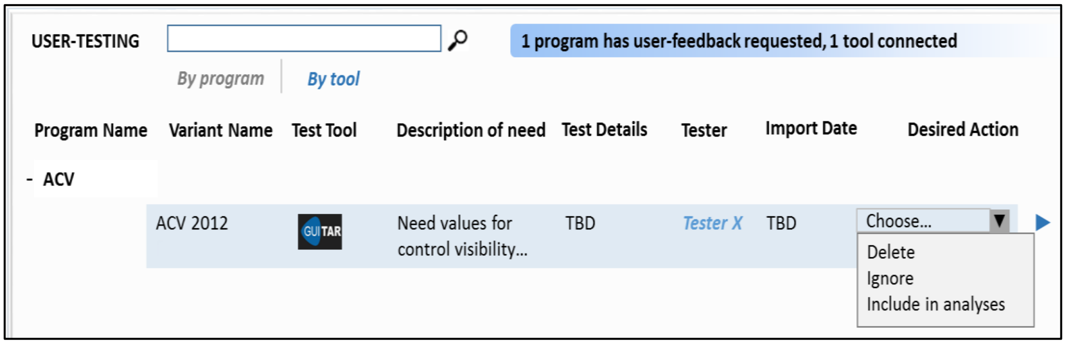
These HSI requirements are treated just like system requirements (enter values, satisfy thresholds, see impacts) because they come with meta-data for everything needed to create the parameters, to add it into the existing FACT interface dynamically, and to present default values if desired. The HSI requirements in HEAT were created by HSI experts on our team and are based upon military standards such as MILSTD1472 revision G (Department of Defense, 2012), ASTM 1166 (ASTM, 2013), Section 508 (General Services Administration, 2017), and commercial standards such as best web design practices. This has created a wide range of relevant HSI requirements in HEAT applicable across hardware, software, users, environments, processes, and operational contexts.

Using the Human Model and Modeling Tools to get Parameter Values

Selecting any of the Human Model children in the WBS pane opens a control panel over the FACT SysML/CAD viewer with options such as taking distance measurements, changing the human model's gender, adding gear, and so forth. By using these tools on the integrated human/system model, experts and non-experts alike can derive values to enter into HSI parameters. Figure 2, at #7, illustrates that the Engineer turned on a vision cone to help find the angle to the bottom-most controls in front of the operator, and evaluate if it satisfies the aforementioned requirement for visual angle of control. In Figure 2 the yellow cone at the bottom displays an angle of 0 to 20 degrees. Thus, HEAT displays a message (Figure 2, above #6) “Parameter is outside of recommended range” because the recommendation is -15 to 15 degrees.

Engineer Requests User Test to Check HSI

One goal of HEAT (Figure 5) is to foster closer collaboration between Engineers designing systems and HSI Experts. Therefore, HEAT provides an icon for each parameter (, at #6 in Figure 2) which when clicked will request that a qualified HSI Tester perform a user test with real end users. HEAT provides to HSI Testers the user interface shown in Figure 5 to display a queue of requested tests. Any FACT user can access the interface in Figure 5 to view the status of test requests. Also, the Engineer requesting the test has authority to decide, via the drop down menu in the lower right of Figure 5, if feedback from the tests conducted should be deleted (e.g., if no longer relevant to the design), ignored for the time being, or included/added the feedback into the program/design variant for the purposes of HSI impact detection.



USER-TESTING

1 program has user-feedback requested, 1 tool connected

By program | By tool

Program Name	Variant Name	Test Tool	Description of need	Test Details	Tester	Import Date	Desired Action
- ACV							
	ACV 2012	GUITAR	Need values for control visibility...	TBD	Tester X	TBD	Choose...

Figure 5: Dialog Box to Display Status of Requested User Tests

The goal of a test might be used to measure end user satisfaction, meaningfulness, cognitive loading, task timing, etc. End-user testing also helps to identify HSI impacts, to check that HSI impacts have been mitigated/resolved, provides insight to the severity of HSI impacts, and most importantly gets HSI values that cannot be determined with another method. When requested to perform a test, the HSI Tester can use whatever test article is available to them (e.g., a visualization of the system under design using FACT’s CAD/SysML model, a cognitive walkthrough of the design, wireframe models, a mockup, prototype).

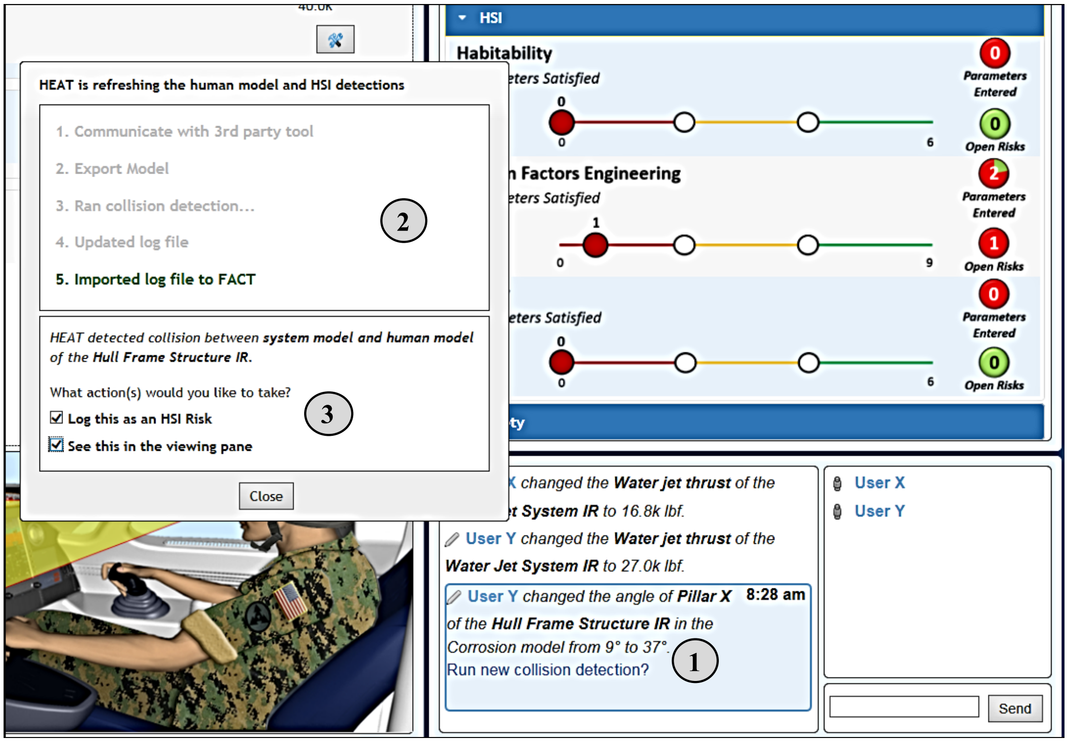
Detecting Design Changes and Updating HSI Impacts

During the design process, users may be collaborating, and the existing FACT communication panel provides a method to be informed of changes and for discussions between users. This is an area that HEAT leverages to inform users of changes that might have an HSI impact and provides a vehicle to do something about it.

For example, a user makes a change to the geometry of the cockpit because they are attending to a requirement or issue related to, say, corrosion. Since system models of FACT (i.e., Corrosion and HSI models) can share data, a geometry change to the cockpit design that an Engineer makes to address corrosion (e.g., relocating a model part) can be detected by HEAT and flagged as a potential impact on a habitable space. FACT has a middle layer that supports custom code writing (used in the past for logging and authentication) that HEAT could use to ‘listen’ for and pick up HSI-related changes.

The message #1 in Figure 6 is a communication sent by the FACT/HEAT system describing what change occurred and asking the user if they want to run collision detection, which would detect collisions between the human and system model based upon this geometry change.

The user can run the collision detection, and HEAT will present the user with a dialog box (Figure 6 #2) to show the progress and report (#3) if any collisions were found. In Figure 6 a collision was detected: “HEAT detected collision between system model and human model of the Hull Frame Structure IR.” The user can then elect (via checkboxes) (#3) to log the collision as an HSI risk and to use the integrated system + human model produced to show the impacts in the SysML/CAD viewer.



HEAT is refreshing the human model and HSI detections

1. Communicate with 3rd party tool
2. Export Model
3. Ran collision detection...
4. Updated log file
5. Imported log file to FACT

*HEAT detected collision between **system model** and **human model** of the **Hull Frame Structure IR**.*

What action(s) would you like to take?

☑ Log this as an HSI Risk

☑ See this in the viewing pane

Close

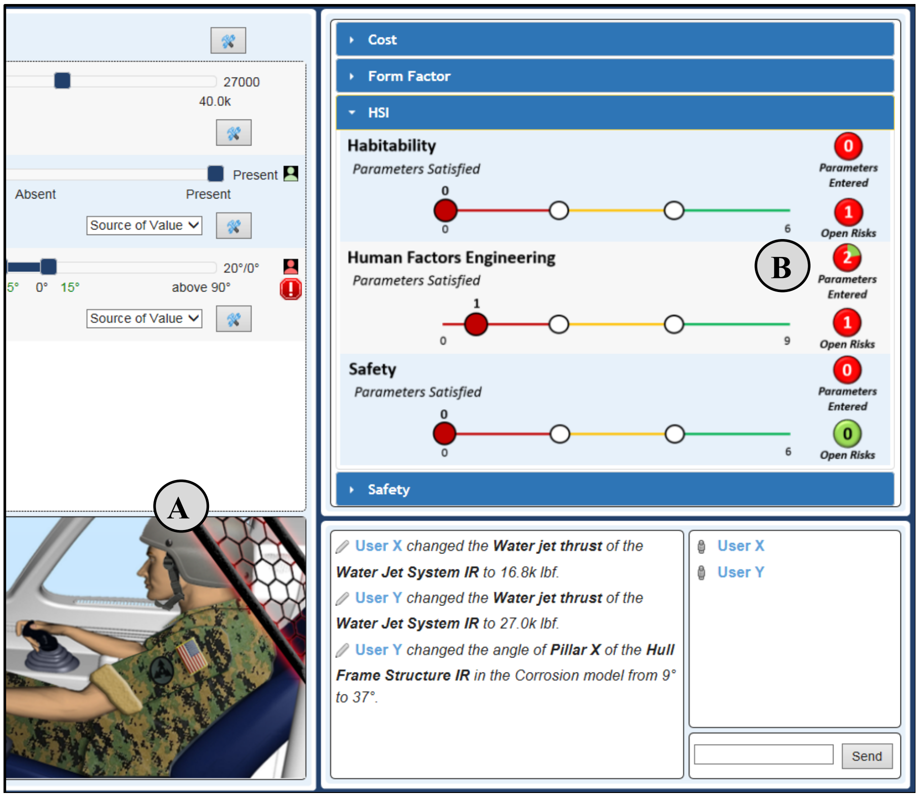
*...X changed the **Water jet thrust** of the ...t **System IR** to 16.8k lbf.*

*User Y changed the **Water jet thrust** of the **Water Jet System IR** to 27.0k lbf.*

*User Y changed the angle of **Pillar X*** 8:28 am *of the **Hull Frame Structure IR** in the Corrosion model from 9° to 37°.*
Run new collision detection?

Figure 6: Assessing the HSI Impact of another User's Design Change

Figure 7 illustrates the result of clicking both checkboxes in the dialog box and closing it. Comparing Figure 7 to Figure 2 earlier, one change is at the grey circle A, where a honeycomb-shaped hull frame bracket is now present (this is the part whose geometry was changed based on the corrosion model). The hull frame part now intersects with the operator's head. The second change (grey circle B) is for the number of risks for the design increase reflecting the user's choice to log the impact as a new HSI risk; the change is from 0 to 1 open risks in the Habitability domain.



*User X changed the **Water jet thrust** of the **Water Jet System IR** to 16.8k lbf.*

*User Y changed the **Water jet thrust** of the **Water Jet System IR** to 27.0k lbf.*

*User Y changed the angle of **Pillar X** of the **Hull Frame Structure IR** in the Corrosion model from 9° to 37°.*

Figure 7: Running Collision Detection (Fig. 6) Produces Changes at A and B

Summarizing HSI Impacts across Different Design Variants

Figure 8 illustrates how HEAT augments an existing FACT display for comparing design variants with an assessment of how well the design satisfies HSI requirements. The green, blue, and gold regions show three design variants for an ACV.

The FACT system provides comparison of design variants by rolling up system parameters into impact sub-groups. Similarly, HEAT groups all HSI parameters, whether they arose from vendor HSI requirements or from HSI requirements added from HEAT's library, by HSI domain. In Figure 8 the domains shown are Habitability, Human Factors Engineering, and Safety; see the grey circled number 1. HEAT rolls up HSI parameters into HSI impact sub-groups too, to indicate for each design variant if the requirements are satisfied, unsatisfied, or (grey circled 2) if the data collected so far is insufficient to determine if requirements are satisfied or unsatisfied. In terms of scoring, the adjective “unsatisfied” is displayed if one or more parameters within a given HSI domain do not meet the criteria laid out in their underlying requirements.

ONGOING WORK TOWARD FULL HEAT SYSTEM

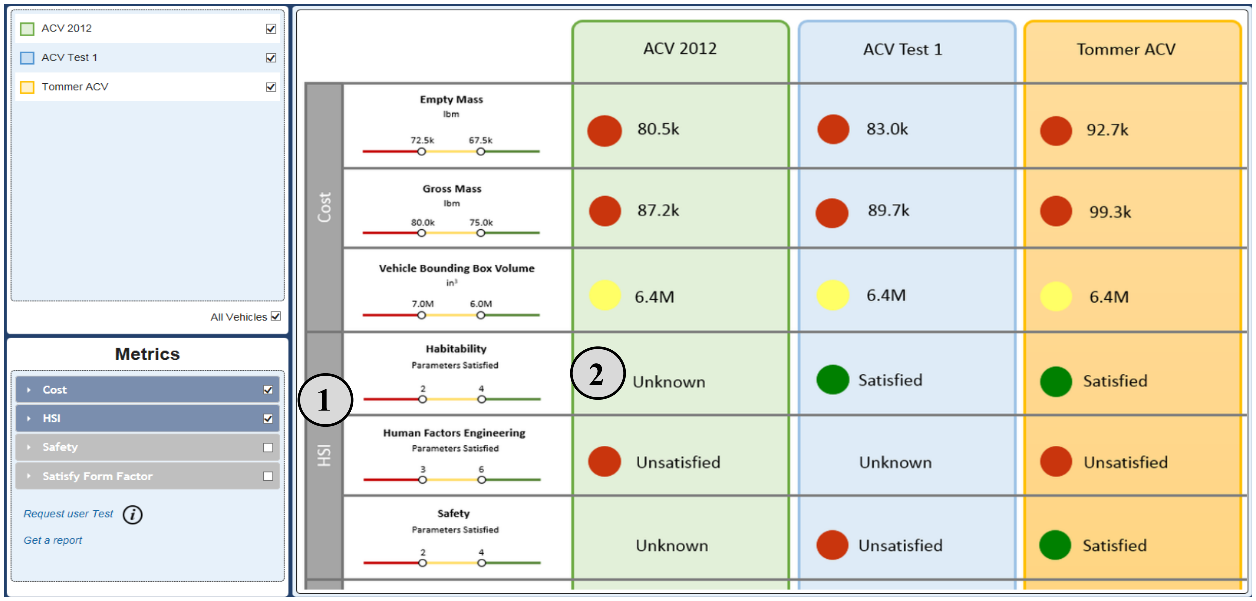
Our team is continuing to work collaboratively with the USMC to develop a fully functional version of HEAT, building upon the prototype discussed in this paper. Our prototype was developed based on FACT versions 1.7 and 2.1, and there are now newer versions of FACT. Our team, in building a full version of HEAT, is working with the FACT team to leverage the updated FACT’s more robust tradespace computations and analysis. HEAT’s data and computations will be inputs to FACT’s calculations reflecting HSI impacts to design factors such as cost, performance, and schedule.

We have encountered some challenges that transcend the FACT and HEAT systems. One is to identify what parts of a SysML or CAD model are human related because engineering designs normally do not flag those parts; we are developing visual means (e.g., for the user to designate where the boundaries are of the crew space). Another challenge is to identify efficient collision algorithms which will allow HEAT to generate an ensemble of human models (e.g., male and female, ranging from 5th to 95th percentile) from a single posed model, and alert the user to a collision for any model in the ensemble. Example algorithms we are considering are (Li, 1988) and (Yoon et al, 2004). One final challenge is the wide variety of formats for SysML and CAD models. An effort in the modeling and simulation community to agree on a small number of standards would simplify the construction of tools such as FACT and HEAT.

ALTERNATIVE SOLUTIONS

We list several alternative approaches, and why HEAT has advantages.

Cognitive models: Cognitive modeling is a big deal currently and it can be quite powerful in approximating human behaviors, such as decision making, task progression, and timing. The major drawback is needing a set of real observed/measured behaviors to either build from or test against in order to evaluate and predict HSI. Additionally, cognitive/mental models are most effective for simulating and evaluating behaviors similar to the behaviors that fed or tested the model, but attempting to apply the model outside of these behaviors greatly reduces the validity and accuracy of the model. Those in the HSI field recognize this shortcoming, and to compensate for this, some cognitive models use combinations of multiple individual models (e.g., a model for operator detection of enemy aircraft + a



- 🟩 ACV 2012 ☑
- 🟦 ACV Test 1 ☑
- 🟨 Tommer ACV ☑

All Vehicles ☑

Metrics

- Cost ☑
- HSI ☑
- Safety ☐
- Satisfy Form Factor ☐

Request user Test

Get a report

		ACV 2012	ACV Test 1	Tommer ACV
Cost	Empty Mass lbm 72.5k 67.5k	● 80.5k	● 83.0k	● 92.7k
	Gross Mass lbm 80.0k 75.0k	● 87.2k	● 89.7k	● 99.3k
	Vehicle Bounding Box Volume in³ 7.0M 6.0M	● 6.4M	● 6.4M	● 6.4M
HSI	Habitability Parameters Satisfied 2 4	Unknown	● Satisfied	● Satisfied
	Human Factors Engineering Parameters Satisfied 3 6	● Unsatisfied	Unknown	● Unsatisfied
	Safety Parameters Satisfied 2 4	Unknown	● Unsatisfied	● Satisfied

Figure 8: HEAT Design Comparison Screen Showing with HSI Impacts

model for response to a threat type), but this takes a lot of time and repetition to prove accuracy. The cost in time, money, and computing resources necessary for development of these models is high. Cognitive models simply are not practical, extensible, cost-effective, or reusable and thus not fit for the customer's desired solution.

Commercial/private HSI products: Some are practical, extensible, and can be cost-effective, but suffer from licensing, development/repurposing, and integration restrictions. Of the most relevant tools, many fell short by being too manual in nature, unsupportive of different user types, too expensive, having their own dedicated look and feel already, or having limited or non-existent developer toolsets to integrate them effectively into FACT. In addition, the toolsets we looked at such as RAMSIS or DELMIA did not provide HSI impact analysis beyond focuses on ergonomics analysis (most common) or certain vehicle types, and were not flexible enough to meet USMC needs.

CONLUDING REMARKS

We constructed an analytical model of the relative costs of finding HSI impacts at different stages in the lifecycle of designing systems, and examined data from our own organizational projects (e.g., to design, develop, test, and deploy hardware/software for the Littoral Combat Ships). From that we found that HEAT can enable a 10x savings on solving HSI related issues by finding impacts early in the lifecycle, versus later in test and evaluation. In fact, we believe the savings will be much more than 10x for complex systems such as vehicles. We also estimate a 60% reduction in time anticipated to find HSI impacts with HEAT versus using distributed tools and manual processes.

ACKNOWLEDGEMENTS

This work was funded in part by USMC MARCORSYSCOM under Small Business Innovation Research Phase I and II contracts M67854-15-C-6508 and M67854-16-C-6500. Some components used in the prototype described here were funded in previous work by the U.S. Army Research Laboratory under contract W911QX13C0075.

REFERENCES

1. ASTM, *F1166: Standard Practice for Human Engineering Design for Marine Systems, Equipment, and Facilities*. July 2013. Retrieved February 22, 1017 from https://www.astm.org/Standards/F1166.htm
2. Department of Defense. *DoD Design Criteria Standard - Human Engineering MIL-STD-147G.* January 2012. Retrieved February 22, 2017 from http://everyspec.com/MIL-STD/MIL-STD-1400-1499/download.php?spec=MIL-STD-1472G.039997.pdf
3. Ender, Tommer R; Browne, Daniel C.; O'Neal, Michael; and Yates, William W. Framework for Assessing Cost and Technology. In M. L. Loper (ed), *Modeling and Simulation in the Systems Engineering Life Cycle*. Chapter 26. 2015. Retrieved February 22, 2017 from http://link.springer.com/chapter/10.1007%2F978-1-4471-5634-5_26
4. General Services Administration. *Section508.gov: GSA Government-wide Section 508 Accessibility Program*. Retrieved February 22, 2017 from https://www.section508.gov
5. Li, Tsai-Yen and Chen, Jin Shin. (1998). Incremental 3D Collision Detection with Hierarchical Data Structures. In Proceeding VRST '98 Proceedings of the ACM Symposium on Virtual Reality Software and Technology (pp. 139-144). ACM Digital Library. Retrieved from http://dl.acm.org/citation.cfm?id=293719
6. Yoon, S., Salomon, B., Lin, M., Manocha, D. (2004). Fast Collision Detection between Massive Models using Dynamic Simplification. Eurographics Symposium on Geometry Processing. Retrieved from http://gamma.cs.unc.edu/MRC/MRC.pdf