Sylvain Bruni, SM, M. L. Cummings, PhD

Human Interaction with Mission Planning Search Algorithms

ABSTRACT

Assigning missions to missiles such as what is needed prior to Tomahawk strikes is an example of a complex resource allocation and optimization problem which includes a large problem space with elements of uncertainty. Attempting to achieve an "optimal" solution is problematic in the military due to high risk environments, dynamic constraints, and the need for rapid decisions under time pressure, which may or may not be improved as a result of the addition of an automated algorithm. In this study we report on the development of mission-missile planning decision support software, and how humans interact and collaborate with a heuristic search algorithm in the context of Tomahawk mission-missile planning. Of particular interest is how the human operator generates mission-missile assignments under different levels of automation: 1) One with low levels of automation (e.g., sorting and filtering assistance), 2) Interactive levels of automation in which the human and algorithm work together to solve a problem, and 3) A mission-planner that makes all assignments and the human must determine whether or not the automated solution is correct. Both human and the heuristic search algorithm performance will be discussed as well as the impact of different levels of automation on situation awareness and the implications for operational use.

INTRODUCTION

In command and control domains such as mission planning where resource allocation and scheduling problems are subject to time-sensitive constraints, computer algorithms are used to optimize a feasible solution. Automation can make computations quickly and accurately based on a predetermined set of rules, however, computer algorithms are notoriously "brittle" in that they can only take into account those quantifiable variables identified in the design stages that were deemed to be critical (Guerlain, 1995). In command and control scenarios, algorithms are likely to be only partially efficient or even completely ineffective if the current environment is not based on the constraints embedded in the computer in the design stages. In addition, because of rapidly changing constraints and significant uncertainty, no absolute optimal assignment solution exists.

In contrast to computers, humans can reason inductively and generate conceptual representations based on both abstract and factual information, thus integrating qualitative and quantitative information. Because of inherent difficulties in automated mission planning, strike planners often have to find "good enough" solutions that respect the current constraints while being easily adaptable to environmental changes. Human operators can quickly adapt their reasoning to come up with solutions that could not be reached using an automated algorithm. However, human operators have difficulty understanding whether or not a solution to a complex scheduling or path planning problem is truly optimal. In addition, human operators are susceptible to automation bias: there is evidence that despite known system limitations, humans are likely to approve erroneous computer-generated recommendations (Skitka et al., 1999). In military command and control domains where decision-making is influenced by external, dynamic constraints, integrating higher levels of automation is problematic because of the inability of automated decision aids to be perfectly reliable and the human propensity for biased decision-making (Cummings, 2004).

Because both humans and computers have different strengths and weaknesses in command and control decision-making, rather than a mutually exclusive assignment of tasks, what is needed is a collaborative approach to decision-making. It is possible that when the human and computer collaborate, they can discover solutions superior to the one either would have determined independently of the other. The goal of this research effort is to determine how humans and computer can collaborate to promote efficient, effective, and robust missile-mission planning.

BACKGROUND

An example of a mission planning problem that could benefit from collaborative human-computer decision making is the mission-missile assignment problem for Tomahawk strike planners. Tomahawk missiles are long range, subsonic cruise missiles used for land attack warfare. The Tomahawk can carry one of three different types of warheads: penetrating, unitary and submunition. They are launched from U.S. Navy surface ships or submarines, up to more than 1000 miles away from their intended targets, with an accuracy of meters (U.S. Navy, 2003). Currently, strike coordinators generate mission-missile assignment using a simple database called PC-MDS. The Personal Computer - Mission Distribution System is a computer and software used to display and to distribute the TLAM mission database. This database contains all the mission data uploaded into missiles, such as terrain contour matching data, GPS (Global Positioning System) and/or DSMAC (Digital Scene Matching Area Correlation) data. Assignments are manually generated by strike coordinators, using either pencil and paper, or their memory to keep track of the different factors and options to consider: PC-MDS does not provide any support for decision making (Cushing, 2003). This process is time and resource consuming because of the variety of parameters available.

In order to address this mission planning problem, an interface was developed known as "StrikeView" (Figure 1). StrikeView is divided into two sections. On the left is the topological map of the area. All items relevant to the situation such as Areas of Interest (AOI), No-Fly Zones (NFZ), Threats, Targets, Launch Baskets (LB) and Routes are symbolically drawn on this map. By selecting the appropriate menu, the operator can create and parameterize as many of those items as needed, as well as all the current available resources (the missiles). The primary goal of the interface is to allow a planner the ability to enter all relevant data for missiles, targets, and preplanned missions (Figure 1), and then match the missiles to the missions with the aid of some level of automated assistance (Figure 2). For each mission, the assignment task consists in finding the corresponding available resource (missile) that matches the mission requirements, such as warhead type, navigation equipment, and launch basket (and hence one or more ships that will be assigned to that launch basket).

Information Type

The main difficulty in the matching process is to reasonably take into account each piece of information that can potentially optimize the final assignment of missiles to missions. Several types of information are available to the operator to make assignment decisions: hard constraints, probabilistic information, and optimization information. "Hard constraints" refer to those pieces of information that strictly determine a mission-missile match: for example, some missions can only be assigned to a penetrating missile, because the targets of those missions require a penetrating warhead.

In addition, the operator may need to consider probabilistic information. In this model, we assigned to each missile a virtual firing rate corre-

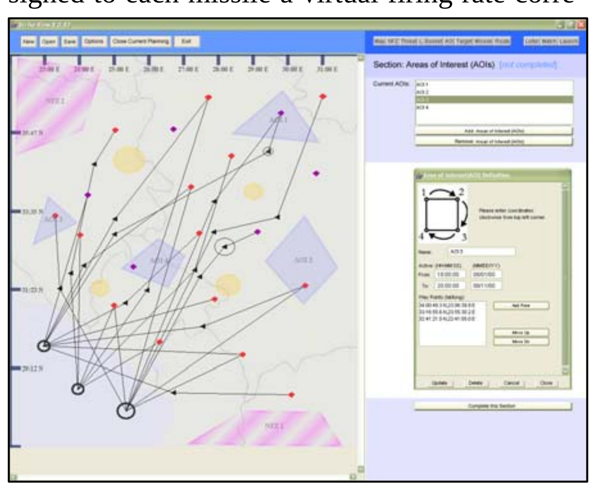

Figure 1 - StrikeView Map.

sponding to each ship's rate of success in missile launches (note that this is not a real world value but an academic representation of probabilistic information). For example, the firing rate of a ship, and thus the (un)certainty associated with the success of a missile launch, may be a crucial element to consider in the process. An operator may decide to prioritize those missiles coming from a ship with a high firing rate. However, the problem then becomes how to balance the assignments when using select missiles reduces the number of possible total assignments?

Finally, the operator can be asked to minimize or maximize some cost function. This optimization information would be used to achieve some "optimal" resource balance. For example, an operator may want to consider the number of days to port for the ships, in order to prioritize the use of missiles aboard ships that are due to port shortly (in order to minimize the number of weapons entering the port).

The complexity of the task results from the need to simultaneously consider and balance these three different types of information. For example, between two missiles that both meet the hard constraints, is it better to choose the missile that has a firing rate of 80% (success of launch) and 30 days to port, or a missile that has a 65% firing rate but 5 days to port? Depending on the current constraints, one might be more appropriate in a certain type of environment than the other. Indeed, if the strike coordinator is told that the corresponding target must be destroyed at any cost, then the former missile with the highest firing rate may be chosen, whereas in another situation the strike coordinator may decide to reduce the total quantity of weapons for a particular ship because it is going back to port and general safety concerns have arisen.

Hence, generating a solution (a mission-missile assignment) becomes very difficult for the human operator when all these parameters have to be simultaneously taken into consideration. The respective strengths of the human and the automation should therefore be balanced to create an effective and efficient assignment process that can generate a satisficing, or good enough, solution. The research question is to know what balance between humans and automation will be efficient.

Levels of Automation

Following Sheridan & Verplank's Levels of Automation (LOA) (1978), three LOA of interest could be embedded in StrikeView:

- LOA 2: "The computer offers a complete set of decisions / actions alternatives". In our case, this means that the computer only provides basic tools such as sorting or filtering, and the process is called "manual matching";
- LOA 3: "The computer narrows the selection to a few alternatives". For this level, we provide an interactive level where the human operator and algorithm work together to solve the problem (called "collaborative matching");
- LOA 4: "The computer suggests one alternative". The algorithm is a mission planner that makes all assignments and the human is left with the decision to approve or not the computed solution (called "automatch").

The Matching Interfaces

INTERFACE 1

The current matching interface (Figure 2) allows for manual matching and automatch. In the former setting, the operator selects a mission in the mission table and a missile in the missile table (among those which have been filtered out by the computer as satisfying the hard constraints). The tables display the primary characteristics of the missions (Target, Route, Launch Basket, Navigation Equipment Required, Priority, Warhead Required, and Number of Missiles Required), and those of the missiles (Ship, Launch Basket, Navigation Equipment Available, Warhead). Then the operator manually adds the match to the matching table. At the bottom left are warning tables that display the targets that cannot be reached (no missile can fulfill the hard constraints requirements), and the unused missiles. At the bottom right is a graphical summary of the current assignment, based on the matches included in the matching table. The horizontal bars fill in according to the number of targets as-

Figure 2 - StrikeView Matching Interface 1.

signed so far, with a breakdown by Target Priority.

LOA 4 is embedded in this interface by the "Automatch" button. When the operator clicks on Automatch, an algorithm instantly generates a mission-missile assignment and stores it in the matching table. Then, the operator has the option to manually modify this solution if deemed necessary. The heuristic search algorithm implemented in automatch sorts the missiles by priority. The missiles that have the fewest number of missions they can fulfill based on hard constraints are ranked first (this is to increase the number of assigned missions). Then, for each missile, the potential missions are prioritized in this order of importance: 1) loiter missions (the missile hovers over an area waiting for an emergent target to pop up), 2) high priority target, 3) medium priority target, and 4) low priority target. Firing rate and days to port information are not yet embedded in this search algorithm, but will in future developments of the software.

INTERFACE 2

Interface 1 does not allow for any real collaboration between the human and the computer, only basic filtering. Interface 2 (Figure 3) was created to leverage the computer's computational power, under human control. Interface 2 still includes the mission, missile, and matching tables, allowing for manual matching. The automatch button (LOA 4) is also available. But additional features have been included for LOA 3 purposes.

First, the automatch is customizable. Whereas in Interface 1 the matching algorithm was completely hidden from the operator, in Interface 2 the operator can actually choose what criteria to include in the automatch, as well as a prioritization order between these criteria. Also, tick boxes next to the mission and missile tables enable the user to select a subset of missions and / or missiles to be considered by automatch.

Furthermore, the assignment summary has evolved to include, in addition to the horizontal bars, two other graphics that synthesize the assignment through the probabilistic (e.g. Firing Rate) and optimization (e.g. Days To Port) data. Finally, this interface includes a "save" option. When used, the current assignment is stored at the bottom of the screen, and a new assignment can be generated without modifying the saved assignment. This provides the user with a what – if comparison between two solutions.

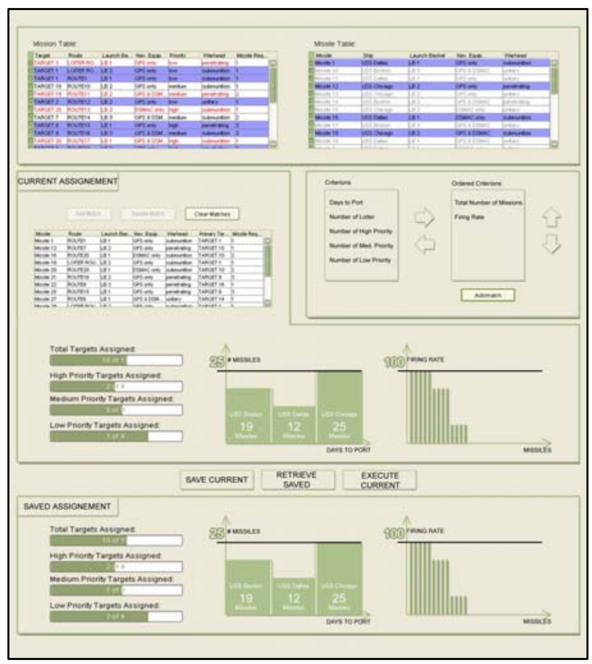

Figure 3 - StrikeView Matching Interface 2.

INTERFACE 3

Interfaces 1 and 2 are both based on the use of raw data. Interface 3 (Figure 4) is completely graphical and the user has no access to the mission and missile tables. The automatch button at the top is similar to that in Interface 1. However, the user can act on the level of prioritization of the probabilistic information (Firing Rate) and optimization information (Days To Port), in the automated algorithm, *via* the central screen sliding bar (the "prioritization bar") that represents what criteria (Firing Rate or Days To Port) should take precedence on the other in automatch.

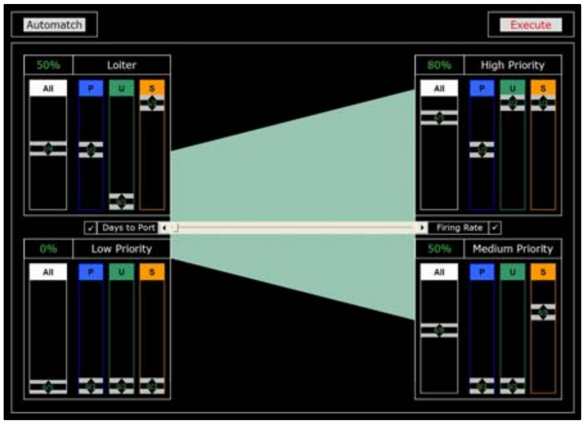

Figure 4 - StrikeView Matching Interface 3.

The result of the assignment computed by automatch is displayed in two ways. First, the breakdown by mission priority (loiter, high, medium, low) in the four corners shows numerically and visually (position of the cursor in the vertical column) how many missions have been assigned, with a secondary breakdown by Warhead type. Then, the green area above and below the prioritization bar metaphorically represents the level of assignment: the more missions have been assigned, the more filled in the central area is. A complete assignment (all missions assigned) would be represented by a completely shaded central area. When the automatch solution is modified by the user, the new solution appears in green, and the first automatch appears as a pale gray in background, for comparison purposes.

Additionally, the user can require the computer to search the solution space to accommodate specific needs: by clicking on the up or down arrows of the cursors in the vertical sliders, the user instructs the computer to find a way to increase or decrease the number of assignments corresponding to the specific slider. Automatch will then compute a new solution to accommodate for this requirement, by potentially modifying other assignments at higher priority levels.

METHODS

It is of primary interest to understand how operators would search the solution space using the different interfaces, how solutions would be generated, and how effective the combined performance of the human and the computer is for the overall mission goal. To this end, six subjects participated in a cognitive walkthrough of these interfaces, including a former TLAM Strike Coordinator, an Air Force ROTC Cadet, an Army Infantryman with 18 years of experience, as well as three graduate students with extensive backgrounds in UAV operation and Human-Computer Interaction, two of them being USAF 2nd Lieutenants.

Cognitive Walkthrough

This method is a classic implementation of User-Centered Design, a process in which the design is significantly influenced by inputs from end-users and field experts (Abras et al., 2004). The cognitive walkthrough evaluates how well a skilled user can perform novel or occasionally performed tasks. In this usability inspection method, ease of learning, ease of use, memorability, effectiveness and utility, among others, are investigated through exploration of the system (Polson and Smith, 1999). Specific questions to be answered using the cognitive walkthrough method include: "Will the user try to achieve the right effect?", "Will the user notice that the correct action is available?", "Will the user associate the correct action with the effect that user is trying to achieve?", and "If the correct action is performed, will the user see that progress is being made toward solution of the task?" (Wharton et al., 1994).

The Experiment

Subjects were first presented with a quick tutorial explaining the basic features of StrikeView's main interface (Figure 1). All items relevant to the environment were described including all their associated parameters (e.g. for missiles, parameters include Launch Basket, Navigation Equipment, Warhead, etc.) Then, subjects explored for a short period of time the actual software to put into practice what they learned during the tutorial.

Subjects were then successively engaged in an informal discussion and manipulation of the three matching interfaces currently in development for StrikeView. For Interface 1 (Figure 2), subjects were first presented its characteristics and main features. They were then asked to perform three quick scenarios in real-time: 1) use and assess the quality of automatch, 2) perform a manual reassignment from an automatch solution, and 3) complete manual assignment.

For Interface 2 (Figure 3), feedback was gathered using a paper prototype of the intended interface. In this case, subjects were asked to describe how they would use the interface in the same conditions. For Interface 3 (Figure 4), subjects were invited to explore an interactive PowerPoint prototype with basic features, following three sets of instructions, 1) exploration of the problem space using the prioritization bar, 2) manual reallocation of resources, and 3) re-evaluation of assignments due to new intelligence). Finally, subjects were asked to answer a series of usability questions by rating between 1 and 10 each of the three interfaces, which will be discussed below.

RESULTS AND DISCUSSION

USER FEEDBACK

In the mission table, the computer highlighted in red those missions which could not be assigned because of the lack of an available missile. Despite instruction during the description of the interface, all subjects felt compelled to click on these red-print missions, either because they did not understand they were unassignable or because they "wanted to make sure that the computer was reliable" (actual quote).

Most subjects asserted that LOA 2 (manual matching) would be unmanageable for an operator alone because of the information overload: it would take too much time to examine all characteristics and come up with solutions just based on hard constraints, let alone including probabilistic and optimization information.

Whereas half of the subjects appreciated the Interface 1 version of the automatch, the customizable automatch of Interface 2 was praised by all subjects because it allowed "the user to both understand how the computer computes a solution, and to be in total control of automation", the latter causing the former. This feature was qualified as extremely valuable, especially by the former TLAM Strike Coordinator: "This is an indispensable tool I wish I had!" Two subjects nevertheless expressed concern that "there should always be a manual way to assign, just in case".

Another main feature that was unanimously welcomed was the save option of Interface 2. Whereas Interface 1 forced the user to keep in memory what the previous assignments were, the save option reduced considerably the memory load for the operator. The save option could also be helpful if multiple criteria needed to be taken into account, if the operator was not sure about the instructions, or if these instructions were ambiguous. The operator could save assignments and compare them to make the best or good-enough choice with respect to the fuzzy constraints.

One subject related a personal experience to illustrate the indispensability of the save option. He was a mission planner for a strike, which was aborted at the last minute because the ground situation had changed. But 24 hours later, the same strike had to be replanned because the original conditions had been reinstated. With a save option, the operator can keep several solutions and basically have them ready to be used if needed, which allows the operator to focus on something else.

Finally, one subject mentioned that in his past work with UAV Predator Operators, he found out that users abhor scrolling through lists, but find it better to have the entire list at one glance, even if that requires the use of an additional screen.

MAPPING & INFORMATION DISPLAY

All these features bring up the more general question of what type and quantity of information to display. For example, the main problem of the automatch in Interface 3 is that operators have no knowledge of the rules behind the process (just like in Interface 1), nor is the influence of the prioritization bar completely transparent.
However, one subject found Interface 3 "very intuitive, with the right amount of information to present to the operator. Not too much information, just the right kind of information, that is the information on which you can easily act". Another subject said that Interface 3 is "a relief from all the raw data of [Interfaces 1 and 2]… but you lose in information precision, for example, what targets are reached".

Simplifying the interface by making it graphical is an important step for high levels of problem-solving. However, the operator should also be allowed to go deeper for more detailed knowledge of the environment if needed. This could be enabled by either an optional access to raw data (to allow manual matching), or a specific mapping between the general graphics to the detailed source of information, that is either the raw data in tables or a topological map.

Subjects had mixed opinions with respect to the display of the Days To Port and Firing Rate information. Some thought that it is too much information that could not be effectively handled when decisions have to be made under time pressure (this is too much detail), whereas others believed these graphics should be available on demand.

Usability Questions

Seven usability questions, partially based on the NASA-TLX questionnaire, were used to rate the three interfaces on a Likert scale from 1 to 10:

1) How much perceptual effort is required to understand and use the interface?
2) How much mental processing is required to understand and use the interface?
3) How well would an operator perform with this interface?
4) How confused would an operator be using this interface?
5) How well does the interface give feedback to the user?
6) How much in control is the operator using the interface?
7) How satisfied vs. frustrated an operator would feel using the interface?

Two-tailed paired t-tests were performed on the ratings of the interfaces, between interfaces 1 and 2, 1 and 3, and 2 and 3. Using the Bonferroni criterion, the 0.05 level of significance was divided by three and results were therefore considered significant at the 0.016 level. We assumed that the parent population of the sample is normally distributed. Results are compiled in Figure 5 (significant differences between interfaces) and Figure 6 (no significant differences).

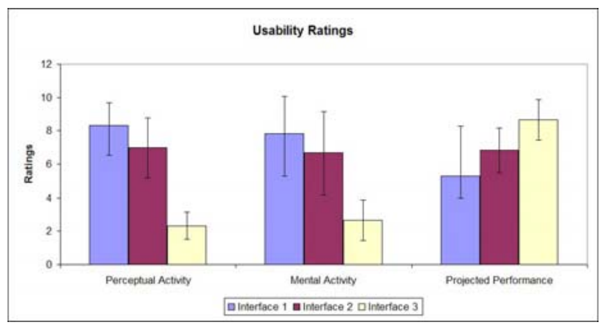

Figure 5 – Significant Usability Ratings

1) *Perceptual Activity* (Figure 5). The purely graphical interface (Interface 3) was considered to require less perceptual effort than Interfaces 1 ($p < 0.0004$) and 2 ($p < 0.003$). This result makes sense since the motivation behind the use of graphics is to minimize the need for and time spent on searching for information. But such an advantage has a cost. First, less information is available through the graphical interface, and then, the information is less precise, in that fewer parameters are visualized and accessible. Therefore, and as mentioned by the subjects, such a display would mainly be used for a rapid overview of the situation, with a few, simple interaction possibilities. This interface is good for conveying information, but insufficient for a comprehensive assignment task.

2) *Mental Activity* (Figure 5). Interface 3 required significantly less mental activity, such as thinking, deciding, calculating, remembering than Interface 1 ($p < 0.006$), and the difference with Interface 2 was almost significant ($p < 0.027$). This reinforces the perceptual activity results: a graphical interface is an efficient way to simply assess the situation without requiring the operator to add a mental process to build another layer of understanding. Indeed, using Interfaces 1 and 2 forces the user to interpret the data on the display: this delays the decision and is also subject to human errors, especially in a time-sensitive environment. In addition to ease of use and attractive to the eye, a graphical interface also simplifies the chain of cognitive processes required to understand and assess correctly the situation.

3) *Projected Performance* (Figure 5). The subjects estimated that Interface 3 would lead to better projected performance than Interfaces 1 ($p < 0.009$) and 2 ($p < 0.002$). But most subjects commented that the projected performance would be better with Interface 3 only if the instructions for the assignment were kept simple. With straightforward instructions, assignment tasks would be done quickly and efficiently. However, as soon as the requirements and constraints for the task increase, the limitation of this interface would surface as detailed information and low level parameters are not accessible.

4) *Confusion* (Figure 6). No significant difference was found between the interfaces regarding the confusion they may generate. Interfaces were rated between a score of 1 (very confusing) and 10 (not at all confusing), and an increasing trend was found: although more visually simple than the table-based interfaces, the graphical interface tended to create more confusion. This may be the result of the inability of Interface 3 to control low level parameters. It is simple and efficient to use in a certain domain, but users' actions are limited: they may get confused because they do not know how to use the interface for specific action (or they do not know that they cannot do these actions). Raw data tables are less confusing because all information is available, and although the interface is more complex, once learned, it may not be as confusing.

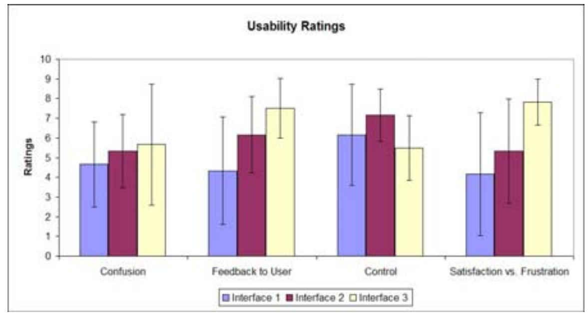

Figure 6 – Non-significant Usability Ratings

5) *Feedback to the user* (Figure 6). This criterion was rated between 1 (poor feedback) and 10 (excellent feedback). A trend emerges from the results: the graphical interface seemed to provide better feedback to the user than Interface 2, which in turn was better than Interface 1. The system's response to user's action is key in the assessment of an interface: the operator needs to know that the intended performed actions have actually been performed. The graphical Interface 3 favors this criterion because change in the appearance of the screen as a result of the action is noticed more by the user than a change in the information inside a huge table of resources. Also, since Interface 2 provides more tools than Interface 1, and thus more feedback, it is understandable that its ratings are slightly higher.

6) *Control* (Figure 6). As expected, Interface 2 was considered the interface users were most in control of, mostly because more options are included in this interface. It is interesting to see that this control issue applies to "how many" actions the user can perform, and not "how much" the user can decide on the assignment. Indeed, it can be that the operator is provided with several automated tools, and hence feels "in control", while the real control is held by the computer in the way those tools are implemented (which is transparent to the user).

7) *Satisfaction vs. Frustration* (Figure 6). The rating scale went from very frustrated (1) to very satisfied (10), and an increasing trend amongst interfaces can be seen. Satisfaction progressively overcame frustration from Interface 1 to Interface 2 to Interface 3. This may be explained by the trends noticed in all other areas: with a

graphical interface, the operator needs less perceptual and mental effort and is more in control, which contributes to an increased level of satisfaction. Conversely, with Interface 1, the range of possible actions was strongly restricted, hence causing frustration because of the inability for the users to do what they wanted.

CONCLUSION

Command and control resource allocation problems are too large in terms of number of resources for the human to explore and manually process. However, because of high levels of uncertainty, the problem is too complicated for the computer which cannot effectively integrate all dynamic variables, changing constraints and intelligence inputs. While preliminary, this research illustrated that collaboration between the operator and the automated algorithm for mission planning can lead to better solutions in terms of satisficing and robustness.

In order to generate an efficient collaboration which leverages the strength of both the human and the computer, a decision support interface should allow planners the ability to act at several levels of detail. The operator should be allowed to quickly explore the solution space at a high level that takes into account the main or most often used criteria, and also aids in identifying the Pareto Front for resource assignment. In terms of the Tomahawk mission planning tool, a customizable automatch feature is a potentially effective support tool because of reduced decision time. However, in order to prevent automation bias and provide flexibility for solution exploration, an option to save generated solutions should be implemented to allow for both an immediate comparison of solutions and future retrieval of assignments. The modality of saving (text or graphic-based) should be tested. Finally, refining the solution with respect to low level details and specific instructions or intelligence should be made available on-demand. The raw data should not be a mandatory step for the assignment, but should still be accessible, in case unusual situations occur. As mentioned by one subject, the intrinsic complexity of this problem may require that several complementary interfaces be developed and simultaneously accessible for complete adaptability, instead of either a fully integrated display, or a multi-layered display.

Although an informal way to gather experts' opinions on the design of human supervisory control interfaces for complex resource allocation problems, the cognitive walkthrough implementation provided insight into how operators would actually interact with such interfaces. An important lesson learned was that, whereas Tomahawk mission planning is a complex problem with many dynamic variables in a time-sensitive environment, it is still possible to create a collaborative interface simple enough (in terms of perceptual and mental activity, confusion, control and interactivity) for users to understand the constraints and solve a problem. From these results, future research will include developing a hybrid interface based on Interfaces 2 and 3, and its extensive testing with military personnel.

REFERENCES

Abras, C., Maloney-Krichmar, D. & Preece, J., "User-Centered Design", in Encyclopedia of HCI, W. Brainbridge Eds, Sage Publications Thousand Oaks, 2004.

Cummings, M.L., "Automation Bias in Intelligent Time Critical Decision Support Systems", AIAA 1st Intelligent Systems Technical Conference, September 2004.

Cushing, J., "A Measurement of the Effectiveness of an Overview Display for the Mission to Launch Platform Assignment Process Essential to the Tomahawk Strike Coordinator", Unpublished SM Thesis, University of Virginia, May 2003.

Guerlain, S.A., “Using the Critiquing Approach to Cope with Brittle Expert Systems”. Paper presented at the Annual Meeting of the Human Factors and Ergonomics Society, 1995.

Polson, P.G. & Smith, N., "The Cockpit Cognitive Walkthrough", 10th Symposium on Aviation Psychology, Columbus OH, 1999.

Sheridan, T.B. & Verplank, W., "Human and Computer Control of Undersea Teleoperators", Man-Machine Systems Laboratory, Department of Mechanical Engineering, MIT, Cambridge, MA, 1978.

Skitka, L.J., Mosier, K.L., and Burdick, M.D., “Does Automation Bias Decision-Making?”, *International Journal of Human-Computer Studies, 51(5)*, p.991-1006, 1999.

U.S. Navy website: http://www.chinfo.navy.mil/navpalib/factfile/missiles/wep-toma.html. Updated August 11th 2003. Last accessed February 25th 2005.

Wharton, C., Rieman, J., Lewis, C. & Polson, P., "The Cognitive Walkthrough Method: A Practitioner's Guide", in Usability Inspection Methods, Nielse and Mack Eds, John Wiley & Sons New York, 1994.

ACKNOWLEDGMENT

This research was sponsored by the Office of Naval Research Long Range Navy and Marine Corps Science & Technology Research Opportunity (BAA 04-001). The views and conclusions presented in this paper are those of the authors and do not represent an official opinion, expressed or implied, of the Office of Naval Research or the acknowledged individuals. Special thanks to Stéphane Essama, who assisted in data collection and experiment management.

Sylvain Bruni obtained his master's degree in Aeronautics and Astronautics at MIT in 2004, and his Engineer diploma from the Ecole Supérieure d’Electricité - Supélec (France). He is currently a PhD candidate in the Humans and Automation Laboratory at MIT.

Mary L. Cummings received her BS in mathematics from the U.S. Naval Academy in 1988, her SM in Space Systems Engineering from the Naval Postgraduate School in 1994, and her PhD in Systems Engineering from the University of Virginia in 2003. She is currently the Boeing Assistant Professor in the Aeronautics and Astronautics Department at MIT and director of the Humans and Automation Laboratory.