Proceedings of the 2015 Winter Simulation Conference
L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, eds.

INCORPORATING AUTOMATION: USING MODELING AND SIMULATION TO ENABLE TASK RE-ALLOCATION

Tyler J. Goodman
Michael E. Miller
Christina F. Rusnock

Department of Systems Engineering and Management
Air Force Institute of Technology
2950 Hobson Way, (AFIT/ENV)
Wright Patterson AFB, OH 45433, USA

ABSTRACT

Models for evaluating changes in human workload as a function of task allocation between humans and automation are investigated. Specifically, SysML activity diagrams and IMPRINT workload models are developed for a tablet-based game with the ability to incorporate automation. Although a first order model could be created by removing workload associated with tasks that are allocated away from the human and to the computer, we discuss the need to improve the activity diagrams and models by capturing workload associated with communicating state information between the human and the automation. Further, these models are extended to capture additional human tasks, which permit the user to maintain situation awareness, enabling the human to monitor the robustness of the automation. Through these model extensions, it is concluded that human workload will be affected by the degree the human relies upon the automation to accurately perform its allocated tasks.

1 INTRODUCTION

In Systems Engineering, a significant step during preliminary system design involves the allocation of functions to various subsystems (Blanchard and Fabrycky 2000). At the highest level, this allocation decision involves assigning functions to a human operator or a machine. Because the quality of this allocation decision is subject to many constraints and considerations, and this decision is typically made very early in the system lifecycle, before system prototyping and in-depth understanding of the system is acquired, this process is often considered an art which cannot be addressed by analytic means (Fuld 2013; Dekker and Woods 2002). Due to the uncertainty inherent in this decision, low risk solutions, for example employing allocations similar to that employed in legacy systems, are often pursued. While low risk, such solutions are not particularly desirable when a primary goal of the system development is to improve human performance or reduce manpower to reduce operational costs. Fortunately, the quality of these allocation decisions can be improved through modeling and simulation of the system and human performance (Army Research Laboratory 2010). For example, modeling of operator workload can provide insight into the system performance consequences of various allocation decisions. While it is acknowledged that optimizing task allocation based upon workload is only one of many potential criteria (Older, Waterson, and Clegg 2010), this paper explores the use of SysML diagrams and human workload models to aid the allocation decision. Specifically, this paper seeks to address the effect that potential changes in allocation, or re-allocations, have upon the structure of task representations within a workload model. By addressing this issue, this paper improves the robustness of human workload models and allows for more accurate and effective task re-allocations.

Perhaps the most frequently cited reference in the function allocation literature is a technical report, which acknowledges that machines perform certain types of tasks better than humans and that humans perform other tasks (e.g., inductive reasoning, flexibility, judgment, selective recall) better than machines (Fitts et al. 1951). Equally important, however, Fitts and his colleagues acknowledge that humans cannot employ their capabilities properly when overloaded due to excessive task demands or when they are unable to maintain alertness due to underactivity, for example when not actively participating in system control. The relationship between human performance and perceived workload resulting from a level of task demand, commonly referred to as the Hebb-Yerkes-Dodson Law, indicates that human performance follows an inverted-U-shaped function with maximum performance occurring at moderate levels of arousal, which permit the human to concentrate on relevant cues within the environment (Teigen 1994). This relationship has been extended to explain the impact of stress and perceived workload on human performance, with human performance nearing an optimal for moderate perceived workload levels (de Waard 1996). Perceived workload generally increases with an increase in the number or complexity of tasks to be performed by the human and as the time available to perform these tasks decreases (Hart and Staveland 1988; Reid and Colle 1988). The level of perceived workload is thus highly linked to the allocation of tasks between the human and computer, which in turn has a significant impact on the performance of the human operator and therefore the performance of the entire system. As a result, Kaber and colleagues have suggested that a decision regarding the level of automation to be applied should be made to minimize a cost function which includes a nonlinear function of workload (Kaber et al. 2009).

Importantly, task load and the resulting perceived workload is not constant during system operation. Instead, changes in the environment, can influence the number and complexity of cues that an operator must process to correctly perceive the environment. For example, consider the number of potential hazards one can encounter when driving on a deserted rural highway versus driving in a crowded city center. The number and complexity of the tasks that must be performed also differ as goals change. For example, consider the complexity of maintaining level flight versus performing a landing, particularly in clear versus adverse weather conditions. This variability in workload is particularly important when investigating automation as the tendency of the automation designer is to automate the functions which are the easiest to automate, potentially creating systems in which the human operator is relegated to a monitor during times that they are easily capable of controlling the system, while performing unassisted during times that they experience peak workload (Colombi et al. 2012). Therefore, it is necessary for any model used for allocation to consider this variability within the context of the work to be performed by the human operator within the allocated system (Dearden, Harrison, and Wright 2000; Wright, Dearden, and Fields, B. 2000).

To account for this variability, this study uses Improved Performance Research Integrated Tool (IMRINT) a discrete event simulation environment (Army Research Laboratory 2010). This environment models human workload and performance as a function of time by tracking activities performed by a human or a machine. These activities are described in a task network, which captures the task sequencing and decision points.. The frequency of the tasks, as well as the time necessary to perform each task result from a stochastic process, permitting the modeler to represent the variability within the system. Different task networks can be derived for different goals and a workload level is assigned to each task performed by the human operator. Various system allocations can then be modeled by allocating specific tasks to be performed by the human operator or machine (hardware or software) component. However, to employ this tool to accomplish this goal, the modeler must begin with a activities to be performed by the system, allocate these activities to the human or machine and then derive the tasks or actions necessary to perform these functions. Once these activities are allocated to a component, human or machine, other inherent tasks may become necessary to facilitate communication of system state as control is passed between the human and machine (Bindewald, Miller, and Peterson 2014).

IMPRINT enables the quick re-allocation of tasks by simply changing the "assignee" for the task from a human operator to an automated component. However, attempts to incorporate automation from a simple re-allocation of tasks previously performed by a human operator to the automated system are unlikely to be sufficient. The current paper develops function and task networks to explore the impact of task re-allocation on changes in the task networks. Specifically, this paper demonstrates that re-allocating tasks previously handled by a human operator to a machine results in the necessary creation of new tasks. This creation of new tasks has implications for the design of the system as well as impacts to the operator's expected workload. While a simple re-assigning of tasks is expected to reduce operator workload and enhance system performance, to be truly accurate workload modeling must account for additional tasks caused by required communications and operator attempts to maintain situation awareness. Through this process we seek to understand and explain the considerations necessary when modeling human workload to support function allocation.

2 METHOD

2.1 Systems Modeling Language

Recent developments in Systems Engineering have led to increased adoption of Model-Based Systems Engineering (MBSE), which commonly includes a modification of the Unified Modeling Language referred to as the Systems Modeling Language (SysML) (Delligatti 2014). SysML captures process allocation through activities and actions within Activity Diagrams. Allocation decisions are captured in Activity diagrams with each actor indicated by unique partition--each partition is colloquially referred to as a "swim lane".

Elements within the activity diagram include action nodes, control nodes, pins, and flows. The actions are the "building blocks" of the diagram which accept inputs and transform them to outputs. The input and output buffers on each activity are pins. Flows connect the output pin of one action to the input pin of another action to enable the passage of information or objects. In the constructed diagrams, the control nodes consist of the decision, merge, fork, and join nodes.

Within this paper, activity diagrams were created within a systems modeling tool, called Enterprise Architect. As appropriate, these diagrams include not only the actions and control logic necessary to depict the necessary "functions," they also include swim lanes to depict particular allocations of these actions to performing entities. These diagrams provide the basis for task networks within IMPRINT.

2.2 IMPRINT

As noted earlier, the Improved Performance Research Integrated Tool (IMPRINT) provides an environment to enable discrete event modeling of human workload. The task networks developed in the activity diagrams were transferred to this modeling environment, capturing the flow of actions and decision logic. Completion of these models would then require development of task time probability distributions for each action and a mental workload value for each action performed by the human operator. Other values, such as action completion accuracy may also be captured for the human or computer as well. While we acknowledge that completion of a model requires the development of these distributions and workload values, as the focus of this paper is to understand the changes in function and task networks necessary to capture changes in allocation, neither the development of these model inputs or the results of the modeling activity are discussed within the current paper.

2.3 Application Environment

To explore the decision to re-allocate tasks from a human to an automated component, it was necessary to select an application environment which was simple enough to permit the task network to be depicted in small activity diagrams and complex enough to provide a series of activities which could be allocated to either a human or a machine. The environment employed in this paper is a tablet computer based game called Space Navigator, which includes a number of activities that can be allocated to a human or a machine (Bindewald, Peterson, and Miller 2015). The game contains four stationary planets present on the screen. Each planet has one of four colors: red, green, blue, or yellow. Spaceships appear at a set interval from a random location at the side of the screen. Each spaceship is red, green, blue, or yellow. The player must direct each spaceship to the destination planet of the same color by drawing a trajectory line on the game touch screen using their finger. The spaceship then follows this line at a constant rate. Spaceships continue to appear until an allotted time of five minutes is over. If desired, trajectories may be re-drawn, to avoid a collision and account for dynamic changes in the environment. Points are earned when a ship successfully reaches its destination planet or traverses any of a number of small bonuses that appear throughout the play area. Upon reaching its destination planet, a spaceship disappears from the screen. When spaceships collide, points are lost and each spaceship involved in the collision is lost.

Additionally, points are lost when a spaceship traverses one of several “no-fly zones” that move to different random locations on the screen at a set time interval. The objective of the game is to earn as many points as possible in five minutes. Figure 1 shows an annotated screen capture from Space Navigator, which illustrates various elements of the game.

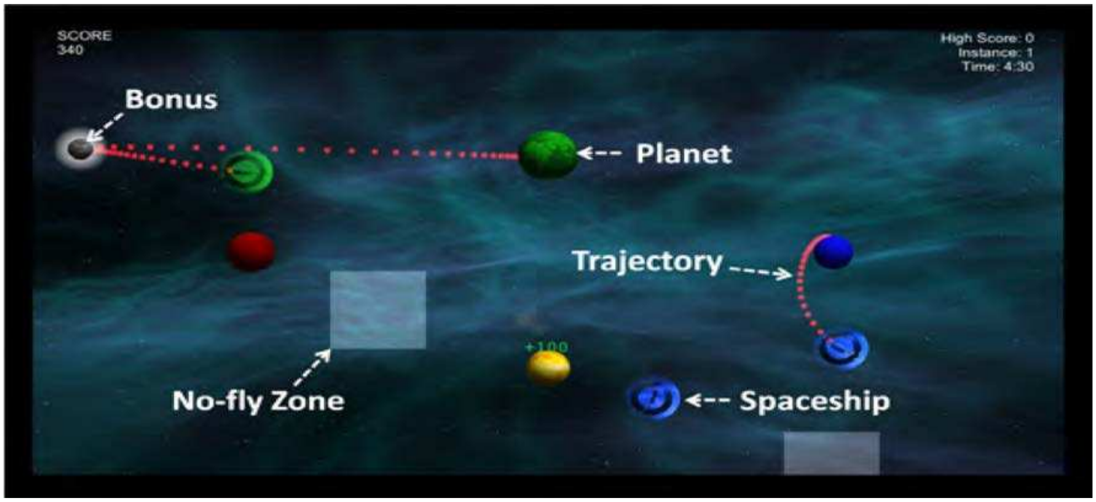

Figure 1: Pictorial representation of the Space Navigator application environment.

2.4 Procedure

We coordinated SysML, IMPRINT, and the Application Environment by creating task networks in Activity Diagrams for the application environment and transferring these task networks to IMPRINT. Initially, these activity diagrams were constructed with the assumption that the human operator was to perform all tasks associated with playing the game. This provided a baseline model, referred to as the Manual System, that accurately demonstrates the actions that were necessary for the human to successfully play the game. A second model, Automation with Direct Re-Allocation, introduces swim lanes to indicate the allocation of actions to the computer or human operator. In this model, the actions were split such that the machine was responsible for indicating the ship which required the most immediate attention by the human operator, while the human operator remained responsible for generating the ship’s route. This task allocation model was revised into a third model, Automation with Handover, to recognize that automation would change the human’s task management strategy, with the operator fully reliant on the machine to perform its target selection actions appropriately. This handover model provides additional actions which permits the automated system to communicate with the human operator. A final task network, Automation with Supervision, is explored for a condition where the human is not fully reliant on the machine but instead monitors the environment to maintain situation awareness, enabling the human to monitor and override the machine in the event of an error.

3 RESULTS AND DISCUSSION

The SysML model activity diagram of the Manual System, as displayed in Figure 2, displays one complete instance of two high level activities: 1) determining which ship to move and 2) which route to draw for it. The operator attempts to attain awareness of the current state of the game environment by identifying all bonuses, ships, likely collisions, no-fly zones, and ships heading for no-fly zones. Based on the operator's priorities, he or she will determine the best ship to move. Potential routes are created by the operator and one is chosen based on earning the highest amount of points possible. The desired ship is selected and the route is drawn. As shown in this diagram, each action performed by the human operator is depicted within an round-tangle. Parallel actions are enabled through the use of the horizontal bars within the figure, depicting splits and joins. The arrows (flows) show the information or control logic which is created within one action and is necessary for the performance of the receiving action. An IMPRINT model corresponding to the activity diagram in Figure 2 is shown in Figure 3. As shown, each of the actions represented in the activity diagram are depicted in the IMPRINT task network.

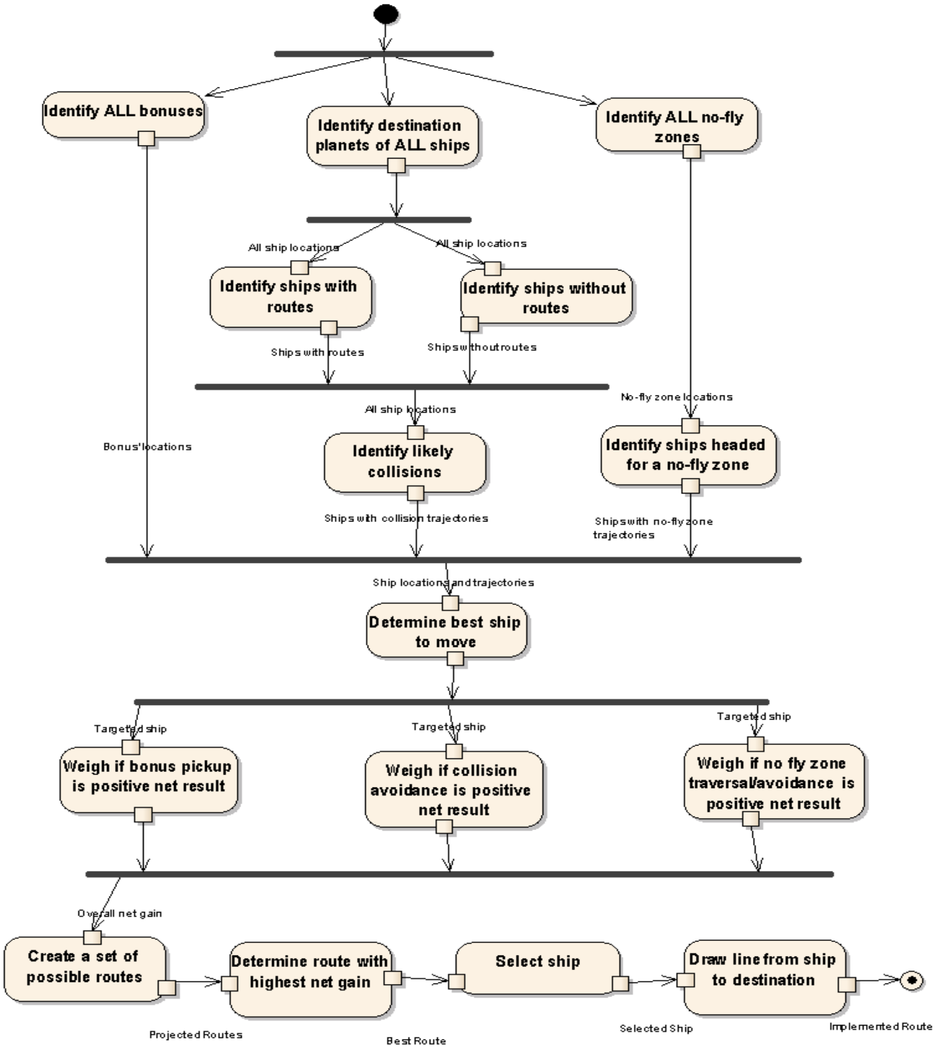

Figure 2: Activity Diagram for the Manual System.

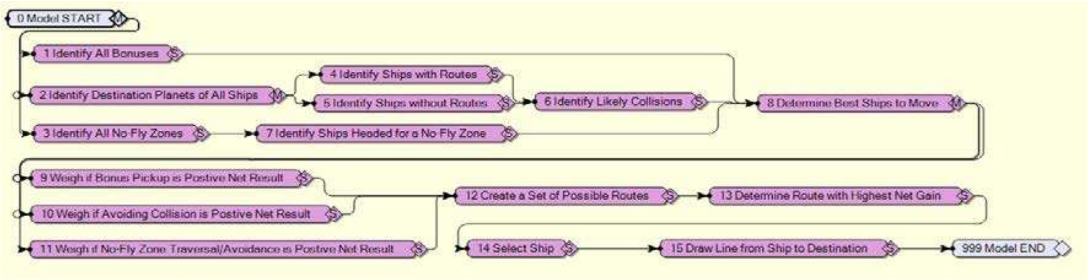

Figure 3: Imprint model illustrating the Manual System. Although not shown, the final model will also likely include the system events (e.g., ship spawn frequency), which are likely to influence human performance.

To explore the implementation of automation through re-allocation of some of the actions, we assumed that the first high level activities, i.e., determine which ship to move, was allocated to the computer and the second high level activity, i.e., determine which route to draw for it, remains with the human. As a default, this change in allocation can be depicted by simply introducing swim lanes to Figure 2 as shown in Figure 4, Automation with Direct Re-Allocation, which indicates the allocation through a "Computer" swimlane in the top of the diagram and a "Human" swimlane on the bottom of the diagram.

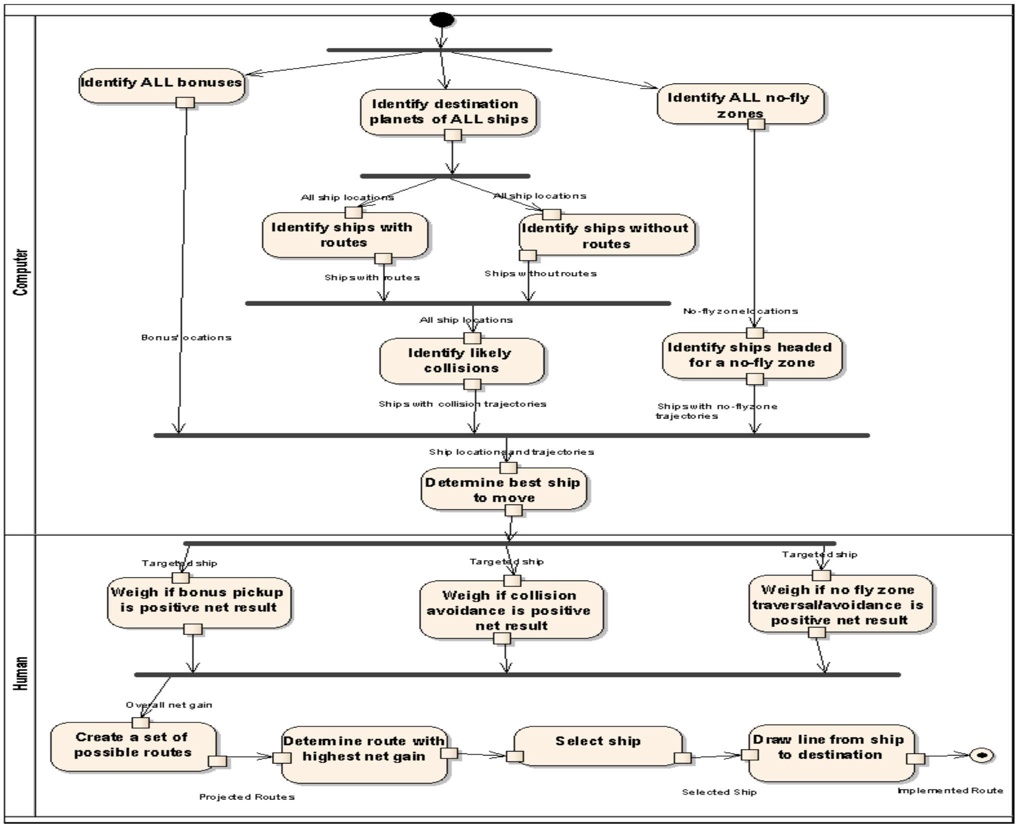

Figure 4: Activity Diagram for Automation with Direct Re-Allocation, with first order task allocation containing partitions indicating control over actions.

This reallocation can then be indicated in IMPRINT by assigning the actions associated with the Computer to the new entity, with this change indicated by the difference in the color of the nodes within Figure 5,where blue and lavender indicates computer and human control, respectively. In these diagrams, the computer is responsible for determining the best ship to move. Afterwards the human operator decides which route to implement.

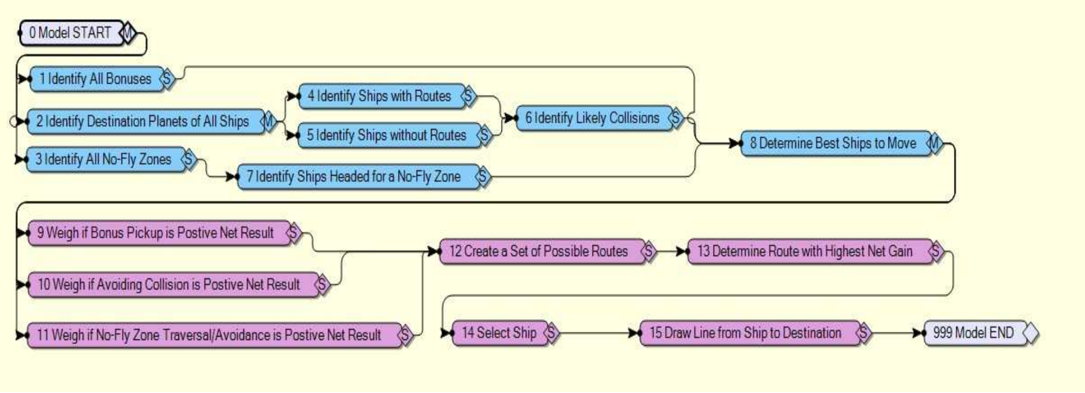

Figure 5: IMPRINT model for Automation with Direct Re-Allocation, displaying computer and human control by blue and lavender nodes, respectively.

Note, however, that in the Manual System model, the human scanned the entire set of objects on the screen and assembled all of the knowledge necessary to know for which ship to draw a route and the reason that a new route was necessary (i.e., new ship without a route, impending collision, heading for new no fly zone, new nearby bonus available). In the Automation with Direct Re-Allocation model, the human has no way of knowing which ship to move or why such a move is important, as the computer has assembled this knowledge but the information has not been transmitted to the human. The need to capture the communication of this information is inserted into a third set of models shown in Figure 6, Automation with Handover. Key adjustments to note are the replacement and addition of action nodes capturing human-computer communication, with the computer relaying to the human why it targeted a specific ship and which ship it targeted (for example by flashing a light around the targeted ship with the color of the light corresponding to the matter that is pressing, e.g., red is a collision, yellow is a no-fly zone, etc). When given this information from the computer, the human identifies the relevant information surrounding the targeted ship, and then creates a set of routes after perceiving the environment around that particular ship, not for the entire screen. As such, this automation aid has the ability to reduce the human's workload as he or she does not need to assess the state of the entire game, only the portion of the game relevant to the highest priority ship, as determined by the game. Unlike the Automation with Direct Re-Allocation model, this task network appropriately identifies additional communication nodes required to ensure an effective handoff between the automation and the human. However, the addition of these communication tasks adds workload beyond what is captured by the simple re-allocation in the Automation with Direct Re-Allocation model.

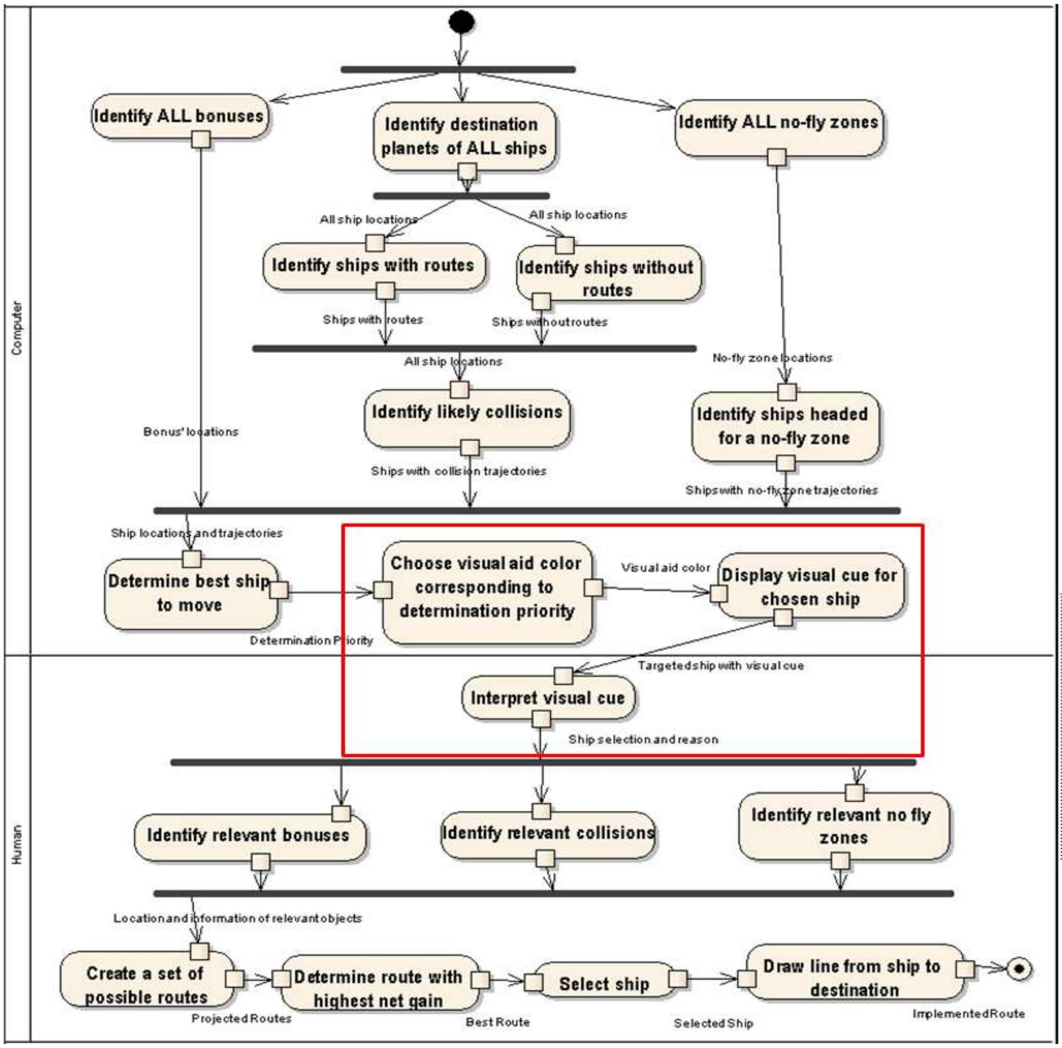

Figure 6: Model displaying automation with handover including a red outline highlighting the communication between the human and automation.

Although the Automation with Handover has the potential to improve the user's performance, assuming that the computer accurately identifies the most important ship to be addressed and the human and computer perform in complete symbiosis, this interaction has the potential to result in less than ideal performance. As (Stensson and Jansson 2014), has indicated, human interaction with automation is necessary since the computer cannot be held responsible, while humans which have the ability to feel remorse among other emotions are assumed to be responsible, particularly for catastrophic outcomes. As such, it will often be necessary for the user to maintain overall situation awareness of the environment to maintain supervisory responsibility over the actions. Unfortunately, as the human relinquishes all ability to verify that the computer has in fact chosen the most important ship to route, the human is unable to maintain responsibility for the task. To enable sufficient situation awareness, many of the functions allocated to the computer in the Automation with Handover model, must also be performed with some degree of regularity by the human to enable the necessary awareness, as shown in Figure 7, Automation with Supervision. Note that in this case, the human is performing as many actions as in the manual system, including actions from the first high level activity, determine which ship to move, which was allocated to the computer. In this scenario, which is not uncommon for automated systems under human-supervisory control, the automated system is unlikely to produce the expected workload benefits.

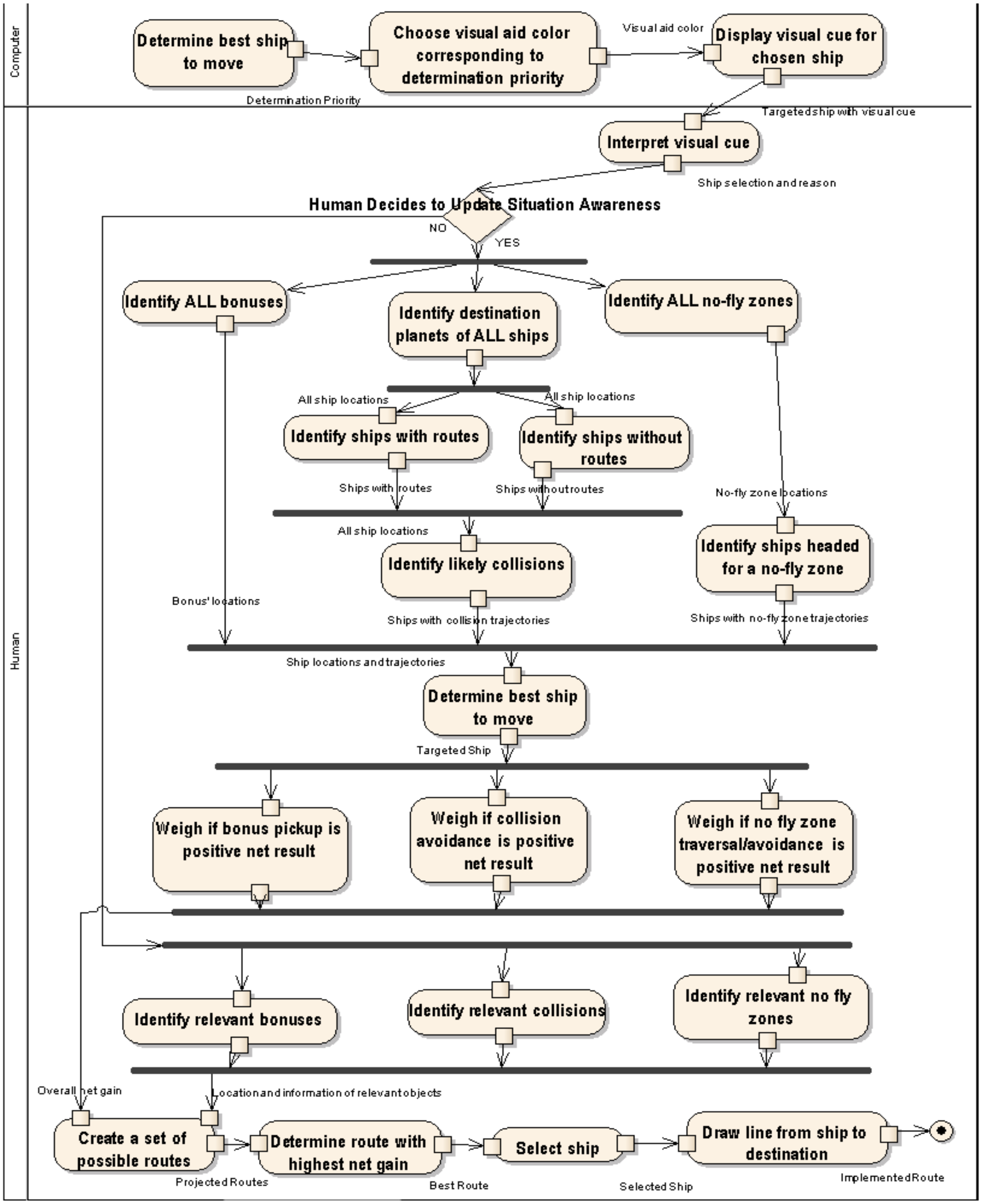

Figure 7: Partial display of the activity diagram for automation with supervision, incorporating communication between the automation and human, as well as the human's decision to update situation awareness.

4 DISCUSSION

The Automation with Direct Re-Allocation Model appears to be a simple and efficient method in adapting a workload model to account for task re-allocations, as it only involves the inclusion of "swimlanes" allocating necessary actions to the human or computer. This provides a model that makes it easy to comprehend which actor is in control of specific tasks. Although this simple modification appears beneficial, it does not accurately capture the true system interactions that will result for incorporating automation.

The major pitfall in the Automation with Direct Re-Allocation Model is that it displays the human operator as seamlessly interacting with the computer without gaining the knowledge necessary to perform the actions assigned to the human. This is a significant issue as it is recognized that the human must sense their environment, perceive relevant information from the environment, decide upon a course of action given this information, and then take action, with each of these phases requiring both mental resources and time (Parasuraman, Sheridan, and Wickens 2000). The lack of additional communication actions in the baseline model of the Manual System is accurate as the human operator completes each of these four steps on his/her own. However, in the Automation with Direct Re-Allocation, the computer gains awareness of all information necessary to select the best ship to move, and then the human implements a route without perceiving the information necessary to select or implement a route. If the operator lacks awareness of his or her surroundings and does not know why a ship is deemed the most critical to move, then he or she will not be able to draw a route that properly addresses the problem at hand.

The Automation with Handover Model fills in these assumptions by including actions which permit the computer to communicate the necessary information to the user during the exchange in responsibility and actions necessary for the user to gain awareness of the situation enabling the decision. This automation reduces human workload by reducing the number of objects in the environment that the human must attend. Unfortunately, this action reduces the user's situation awareness. The final model, Automation with Supervision, then adds additional actions the human must perform to regain this situation awareness. In the final environment, the time allotted by the human for gaining situation awareness versus route creation will depend on the human's trust in the automation, system reliability, time available, and the relative importance he or she assigns to each of these higher level activities, all of which will need to be captured in the workload model.

5 CONCLUSIONS

This paper has illustrated the potential use of SysML together with IMPRINT to illustrate the construction of models to assess task re-allocation. Although initial allocation of actions within these models appears simple and intuitive, only requiring designating responsibility for existing actions, key assumptions are not explicitly depicted in the model. Adaptation of a model to include task re-allocation requires careful consideration in the areas of human-automation communication and adjustments in behavior. It is significant for the developer to understand that task flow between a human and computer involves some type of input or output from both. Any adjustments in human behavior, arising as a result of automation, need to be addressed and input into an adapted model. Revision of action nodes and the inclusion of human-computer communication, as well as human monitoring to gain situation awareness, results in an activity diagram and IMPRINT model that is able to more accurately represent the system and project the workload of the human operator.

In the development of a new system, the accuracy of a model, or set of models, is critical to the further development of the system. Models and simulations are often made in the conceptual phase of system development, capturing the fundamental elements of projected system attributes and behavior in a cost efficient manner. Conceptual modeling is the cornerstone for Model Based Systems Engineering (MBSE), affecting nearly every aspect of the development and implementation of the system. If the model neglects certain aspects of the system, this could have a negative impact on the project's budget, schedule, requirements, functionality, and feasibility. Therefore, in the context of modeling human-computer interaction, one needs to apply careful consideration regarding communication, situation awareness, and behavior to properly capture system behavior and avoid undesired costs.

6 FUTURE RESEARCH

The current research primarily focused on modeling theory when considering human-computer interaction. The next step would be the application of these theories by using the models to estimate system performance and human workload for each of the system designs discussed (Manual System, Automation with Direct Re-Allocation, Automation with Handover, and Automation with Supervision). Model outputs regarding predicted system performance and workload could be validated using human test subjects for each of the system designs. This would enable a quantification of the negative impacts from inadequate automation task re-allocation.

ACKNOWLEDGMENTS

The views in this article are those of the authors and do not necessarily reflect the official policy or position of the Department of the Air Force, Department of Defense, nor the U.S. Government. The authors gratefully acknowledge the support of the Air Force Office of Scientific Research, Robust Decision-Making in Human-System Interface Research Area, which partially funded this research.

REFERENCES

Army Research Laboratory. 2010. *Improved Performance Research Integration (IMPRINT) Tool*. Accessed April 2. https://dap.dau.mil/aphome/das/Lists/Software%20Tools/DispForm.aspx?ID=58

Bindewald, J. M., G. L. Peterson, and M. E. Miller. 2015. “Trajectory Generation with Player Modeling.” *Proceedings of AI 2015 - Canadian Artificial Intelligence Conference*. Halifax, Nova Scotia, CA.

Bindewald, J., M. Miller, and G. Peterson. 2014. “A Function-to-Task Process Model for Adaptive Automation System Design.” *International Journal of Human-Computer Studies* 72:822-834.

Blanchard, B., and W. Fabrycky. 2000. *Systems Engineering and Analysis*. Englewood Cliffs, NJ: Prentice-Hall, Inc.

Colombi, J. M., M. E. Miller, M. Schneider, J. McGorgan, D. S. Long, and J Ploga. 2012. “Predictive Mental Workload Modeling for Semi-Autonomous System Design: Implications for Systems of Systems.” *Journal of Systems Engineering* 15 (4):448-460.

de Waard, D. 1996. *The Measurement of Drivers' Mental Workoad*. Haren, The Netherlands: Doctoral Dissertation, The Traffic Research Center VSC, University of Groningen.

Dearden, A., M. Harrison, and P. Wright. 2000. “Allocation of Function: Scenarios, Context and The Economics of Effort.” *International Journal of Human-Computer Studies* 52:289-318.

Dekker, S., and D. Woods. 2002. “MABA-MABA or Abracadabra? Progress on Human-Automation Coordination.” *Cognition, Technology and Work* 4:240-244.

Delligatti, L. 2014. *SysML Distilled: A Brief Guide to The Systems Modeling Language*. Upper Saddle River, NJ: Pearson Education, Inc.

Fitts, P., A. Chapanis, F. Frick, W. Garner, J. Gebhard, W. Grether, et al. 1951. *Human Engineering for an Effective Air-Navigation and Traffic-Control System*. Washington, D.C.: National Research Council Committee on Aviation Psychology.

Fuld, R. 2013. “The Fiction of Function Allocation.” *Ergonomics in Design*, 20-24.

Hart, S. G., and L. E. Staveland. 1988. “Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research.” *Advances in Psychology* 52:139-183.

Kaber, D., N. Stoll, K. Thurow, R. Green, S. Kim, and P. Mosaly. 2009. "Human-Automation Interaction Strategies and Models for Life Science Applications." *Human Factors and Ergonomics in Manufacturing* 19 (6):601-621.

Older, M., P. Waterson, and C. Clegg. 2010. "A Critical Assessment of Task Allocation Methods and Their Applicability." *Ergonomics* 40 (2):37-41.

Parasuraman, R., T. B. Sheridan, and C. D. Wickens. 2000. "A Model for Types and Levels of Human Interaction with Automation." *IEEE Transactions on Systems, Man, and Cybernetics* 12.

Reid, G. B., and H. A. Colle. 1988. "Critical SWAT Values for Predicting Operator Overload." *The Human Factors Society - 32nd Annual Meeting*, 1414-1418. Santa Monica, CA.

Stensson, P., and A. Jansson. 2014. "Autonomous Technology - Sources of Confusion: A Model for Explanation and Prediction of Conceptual Shifts." *Ergonomics* 57:455-470.

Teigen, K. H. (1994). "Yerkes-Dodson: A Law for All Seasons." *Theory and Psychology* 4 (4):525-547.

Wright, P., A. Dearden, and B. Fields. 2000. "Function Allocation: A Perspective From Studies of Work Practice." *International Journal of Human-Computer Studies* 52:335-355.

AUTHOR BIOGRAPHIES

TYLER J. GOODMAN is a Master of Science Student in the Department of Systems Engineering and Management at the Air Force Institute of Technology, Wright Patterson AFB, OH. He holds B.S. in Industrial Engineering from the University of Miami. His thesis research involves design and evaluation methods for systems employing adaptive automation. His email address is tyler.goodman@afit.edu.

MICHAEL E. MILLER is an Assistant Professor of Systems Integration in the Department of Systems Engineering and Management at the Air Force Institute of Technology, Wright Patterson AFB, OH. He holds a Ph.D. in Industrial and Systems Engineering from Virginia Tech. His research interests include display design, human-machine interaction, automation, and their applications in aircraft and consumer products. He is a senior member of IIE and SID, as well as a member of HFES and INCOSE. His e-mail address is michael.miller@afit.edu.

CHRISTINA F. RUSNOCK is an Assistant Professor of the Department of Systems Engineering and Management at the Air Force Institute of Technology, Wright Patterson AFB, OH. Her research interests include human performance modeling, mental workload, and situational awareness, with a focus on applications in human-machine interaction and adaptive automation. She is a member of IIE and HFES. Her email address is christina.rusnock@afit.edu.