HAL2017-01:
Determining UAV/UGV Training Effectiveness as Autonomy Increases
Final Report

W911NF-12-R-0011-02

December 21st, 2017

Mary Cummings (P.I.)

Humans and Autonomy Laboratory
130 North Building
Duke University
Durham, NC 27708

Determining UAV/UGV training effectiveness as autonomy increases

FINAL REPORT
W911NF-14-1-0632.

21 DEC 2017

Executive Summary

The Army and more broadly, the DoD is making significant investment in increasing the level of autonomy on board unmanned vehicles so that mission effectiveness can increase with a decrease in operational and training costs. However, to date there is no principled methodology to understand how to measure training effectiveness for unmanned system operators, particularly in terms of understanding how increased autonomy could change the training requirements and how such changes could be assessed in advance to provide feedback for autonomy and systems engineers.

In order to address this gap in assessing the effect of increasing autonomy on unmanned system training evaluation, a model is needed that explicitly considers training in the context of different levels of cognitive reasoning by either humans or automation. Such reasoning is commonly referred to as skill, rule, and knowledge-based reasoning. Any modeling strategy that does not assess whether the human, automation, or combination of the two are meeting mission objectives is necessarily incomplete. This is a critical gap since autonomy capabilities are rapidly changing and will necessarily impact training requirements but it is not clear how and what new latent issues could emerge as a result.

To this end and in support of Tasks 1, 2a, and 3a in the original Statement of Work (Appendix A), a model was developed that captures training requirements, human capabilities and performance, and changing levels of autonomy. This model, called the Army UAS Autonomous System Training Model, was able to successfully replicate the numbers and rates of Army trainees in Shadow Unmanned Aerial Vehicle training programs. This model also determined that overall training time was most affected by variability in Module C training, which focused on UAS Ground School. These results indicate if the Army wants to graduate people both faster and with more predictability, they need to address why and how often people are being recycled, as well as the extent of Module C training variability.

One problem encountered during the conduct of this research was representing model changes in vehicle autonomy over the life of current Army UAV training programs. Thus, to validate the part of the model that specifically addressed increasing vehicle autonomy and the impact on skill, rule, and knowledge-based reasoning behavior (which was Task 3b), an experiment was designed to be run on US Army UAV operators in a simulated multiple UAV supervisory control test bed called the Research Environment for Supervisory Control of Heterogeneous Unmanned Vehicles (RESCHU).

Unfortunately, the experiment was not conducted on the target population due to delays in administrative approval and reviews. However, other related experiments yielded important insights into the relationships between increasing cognitive levels of reasoning and the influence of autonomy. These studies indicated that there could be positive training benefit on tasks unrelated to the task specifically being trained to master, and that lower level skill training may most benefit novices. This is important because these results suggest that while there may be elements of negative transfer of training for experts when faced with high workload, experts also may have more acutely developed senses of situation awareness that allow them to make fewer mistakes, although they may need additional training time. These are very important finding for the Army in their UAS programs which retrains a high number of operators on new platforms,

called reclassified operators. Future work funded either through the Army or other sources, should investigate the generalizability of these findings and also look more closely at the cognitive needs of reclassified individuals.

Another significant milestone in this effort was investigating how machine learning could be used to detect previously unrecognized patterns in operator strategies that could be developed through training, as well as possible poor design (Task 2b). Also using the RESCHU test bed, the results demonstrated that indeed, the use of Hidden Markov Models yielded critical insight into those strategies adapted by people in the supervision of multiple UAVs. This research demonstrated that such models could be used to both identify strategy patterns not immediately observable as well as diagnose where strategy inefficiencies caused periods of high workload. This research could be groundbreaking in that such models clearly capture where human strategies may not align with those of system designers, highlighting either design or training inefficiencies.

Table of Contents

Introduction

A recent Government Accountability Office (GAO) report highlighted critical Army Unmanned Aerial Systems (UAS) training shortfalls, with 84% of UAS units meeting less than half of annual minimum training goals in FY2015. The report concluded that without revising its strategy to address its training shortfalls, the Army risks continuing to train at levels below Army goals (United States Government Accountability Office, 2017). Moreover, with the increase in both the number of Unmanned Aerial Systems (UAS) used by the Army, as well as the like increase of onboard autonomy for Unmanned Aerial Vehicles (UAVs), significant changes operations and training methods will be required.

In order to aid the Army in determining where inefficiencies and chokepoints in current UAS training pipelines exist, as well as how increasing autonomy could affect future training programs, a model was developed for predicting changes in training of UAV operators as the autonomy onboard the vehicles increases. The original proposal for this effort contained three tasks (the original Statement of Work is included in Appendix A). The first section of this report discusses this model in detail (Task 1), as well as the results from a parallel study that investigated the validity of the feedback model in a simulated task environment (Task 3b). This report concludes with a discussion of an effort investigating whether machine learning models can be used to identify training gaps (Task 2b).

Model Development

The addition of autonomous capabilities onboard unmanned systems shifts the role of the human from a hands-on operator to a supervisor that monitors relayed (and often delayed) sensor data via the control station user interface. It is essential, however, that the operator understands the interfaces, functions, and limits of the system such as to avoid automation bias and confusion (Lee & Moray, 1992; Parasuraman & Manzey, 2010). Insufficient training has led to numerous human-machine error-induced accidents (Tvaryanas, 2006), which potentially could have been avoided if training requirements aligned with requirements for proper human-machine collaboration.

Blitch (2007) tested the effect of operator automation assistance on UAS maneuvering skills in a Predator simulator and found that trainees who received automation assistance performed poorer when given a novel landing task. This finding suggests that training will need to be adapted to provide a broader understanding of the system when increasing autonomy is inserted into supervisory control systems. Case reports in other domains have found that introducing automation into production lines has decreased the initial training time for operators, but increased the amount of on-the-job (OJT) training received (Brown & Campbell, 2000). Mitchell, Yadrick, and Bennett (1993) designed and validated a decision support system (DSS) using historical Air Force training data that was used for planning training requirements but did not address the impact of new capabilities or increasing autonomy.

For several decades, many researchers have hypothesized how future training programs should better prepare operators of unmanned system for typical operations, as well as contingency situations (Liu, 1997; Singh, Tiwari, & Singh, 2009; Tsang & Vidulich, 2003). There is disagreement between whether the implementation of autonomy increases (Sarter, Woods, & Billings, 1997; Wiener & Curry, 1980) or decreases (Amalberti, 1999) the training time and requirements for operators. Models, such as the one described in this work, aim to set the foundation for understanding the impact of autonomy on operator behaviors that are then linked to training requirements.

Current UAS operator training

The U.S. Army Shadow and Gray Eagle UAS operator training programs were selected as the case studies since they share a well-structured set of objectives and training syllabi, and also are mature programs with more than 15 years of training for the Shadow and 5 years of training for the Gray Eagle. The crew of the Shadow, which conducts intelligence, surveillance, and reconnaissance (ISR) missions, consists of an aircraft operator (AO) and payload operator (PO). The role of the AO is to set mission waypoints and monitor the trajectory of the aircraft, while the PO's role is to monitor and control the aircraft payload.

A relatively small tactical unmanned aerial vehicle (UAV), the Shadow can be launched and recovered without a constructed runway. The Gray Eagle is a larger UAS and requires a runway for take-off and landing and requires the AO to manually taxi the aircraft from a Ground Control Station (GCS). Also an ISR UAS, the Gray Eagle can carry weapons but is used less in the U.S. Army due to its upfront and operational costs and take-off/landing constraints.

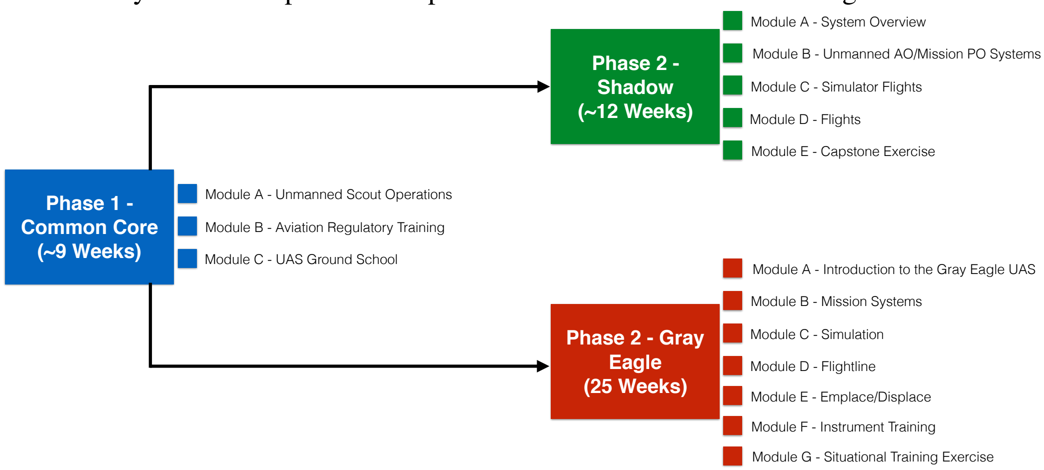


Figure 1. UAS operator training program for the U.S. Army. The modules are listed next to the phase boxes, along with descriptions.

Figure 1 represents the multi-phase process of U.S. Army UAS operator training. The first step towards graduation is to complete a Phase 1 - Common Core program (represented in blue in Figure 1). Phase 1 training occurs primarily in a classroom setting and consists of flight safety, navigation, mission planning, aerodynamics, and aviation regulation. Phase 1 lasts just over two months and includes three modules composed of sets of lessons, tasks, and exams to measure trainee proficiency. Upon graduation from Phase 1, trainees move into Phase 2 where they receive platform-specific training for either the Shadow or the Gray Eagle, which consists of a 5-module, 3-month course, or 7-module, 6-month program, respectively.

The aforementioned modules are comprised of lessons that have a discrete number of tasks that trainees must complete before being able to train with their units. The tasks may occur throughout multiple lessons within each training phase. Tasks vary in duration, setting, and complexity depending on the phase and module. For example, learning Federal Aviation Administration (FAA) regulations occurs in Phase 1 since all operators will work in the same regulatory environment, while performing a simulated launch and recovery flight takes place during the platform-specific Phase 2 training within the simulator module.

Linking operator behaviors to training

In order to understand how operator behaviors in various modules could be translated into training requirements, the Skills, Rules, and Knowledge (SRK) framework (Rasmussen, 1983) was applied. The SRK framework was created as a model for understanding how systems should be designed such that information displays align with human perception and cognition information processing requirements. The resulting taxonomy is a three-tiered continuum that abstracts the ways in which human operators reason and interact with machines.

Skill-based behaviors are defined as those that are highly automatic and lowest on the cognitive continuum, and thus serve as the best candidate behaviors for machines to perform (Cummings, 2014). An example of a Skill-Based Task (SBT) in UAS training is target tracking, which requires the PO to keep the UAS camera sensor locked onto a target using a trackball. This task is psycho-motor driven and highly automatic.

Rule-based behaviors are next highest on the cognitive continuum and are those that require the operator to follow prescribed procedures or checklists. Rule-Based Tasks (RBT) are common in U.S. Army UAS training operations, such as reviewing pre-flight checklists in a logical order before every flight. Tasks such as UAS emergency procedures require human judgement and reasoning, which embody knowledge-based behaviors, the highest on the SRK framework. These behaviors are the most abstract and require planning under the presence of uncertainty to reach a desired goal.

Scenarios that are novel, uncertain or have unpredictable exogenous variables influencing planning and decision-making fall under the Knowledge classification (Cummings, 2014). An example of a Knowledge-Based Task (KBT) would be planning how to react to an unpredictable meteorological event. While the operator may resort to underlying Rules to aid in the decision process, the abstract planning of how to apply Rules requires a higher level of cognition, i.e. Knowledge-based reasoning.

Prior work has since adapted the SRK framework to many aviation applications. Kilgore and St-Cy (2006) investigated the potential of using the SRK framework for analyzing an air traffic control (ATC) rerouting task. The analysis involved the systematic segmentation of the rerouting task into sub-tasks that could have one, or a combination of operator behaviors, depending on the complexity of the situation. Fleming and Pritchett (2016) found that with systems of multiple levels of automation, humans are needed for KBTs for abstract decision making and deciding when automation does not need to be used. Their findings align with Cummings's (2014) assertion that the best candidates for automation are SBTs and some RBTs due to the limited uncertainty in performing rehearsed, psycho-motor actions and procedures. For increasingly autonomous system training operations, it is essential that operator behaviors be linked to training to understand where future autonomous capabilities could or should be implemented to leverage the strengths of the system and the human operator, and to what end this will affect overall training requirements.

To better understand the cognitive complexity associated with UAS operations, each of the Shadow and Gray Eagle operator training tasks was categorized as SBT, RBT, and/or KBT. Some tasks include multiple operator behaviors, such as reacting to emergency scenarios, which potentially requires SBT, RBT, and KBT. Table 1 gives a sample breakdown of how tasks that constitute a lesson can be categorized using the SRK framework. Figure 2 shows an example of key words that assist in identifying a specific operator behavior, or set of operator behaviors, for a task. For SBT, the task description should describe some physical interaction with vehicle controls or equipment. RBT, which make up the majority of UAS operator tasks, require procedures that the trainees are required to follow. KBT are more abstract and require complex reasoning to react to high levels of uncertainty. Tables linking all tasks to operator behaviors, such as Table 1, were completed for all training phases in Figure 1.

Figure 3 shows the percentages of SBT, RBT, and KBT across each of the operator training phases. The allocation of times for SBT, RBT, and KBT were determined by analyzing each of the tasks for the lessons within the modules and assigning uniform times with respect to the total documented time of the lessons. For example, a lesson that takes 7.2 hours and has 10 tasks is assumed to divide evenly into 0.72 hours per task. This assumption of uniform distribution of task time per lesson is one of the limitations of our process of linking the SRK framework to actual tasks for the UAS operator application. However, this was an assumption made in the lack of more data. Training programs that have finer detail for task training time within lessons would permit this assumption to be lifted.

Table 1. Sample Phase 2 lesson operator behavior-based task breakdown.

Lesson – Basic Simulator Flight	
Task	Operator Behavior
Perform UAS Mission Operations	R
Prepare UAS Reports	R
Identify Data Collected by UAS	R,K
Perform Tactical UAS Launch Procedures	R
Perform Tactical UAS Recovery Procedures	R
Perform Tactical UAS GCS Power Down Procedures	R
Perform Tactical UAS Portable GCS Power Down Procedures	R
Operate Tactical UAS Mission Planning System	S,R
Perform Tactical UAS Preflight Procedures	R
Perform Radio Communication Procedures	R
Track a Moving Target	S
Respond to Inadvertent Instrument Meteorological Condition	S,R,K
Perform Radio Failure Procedures	R,K

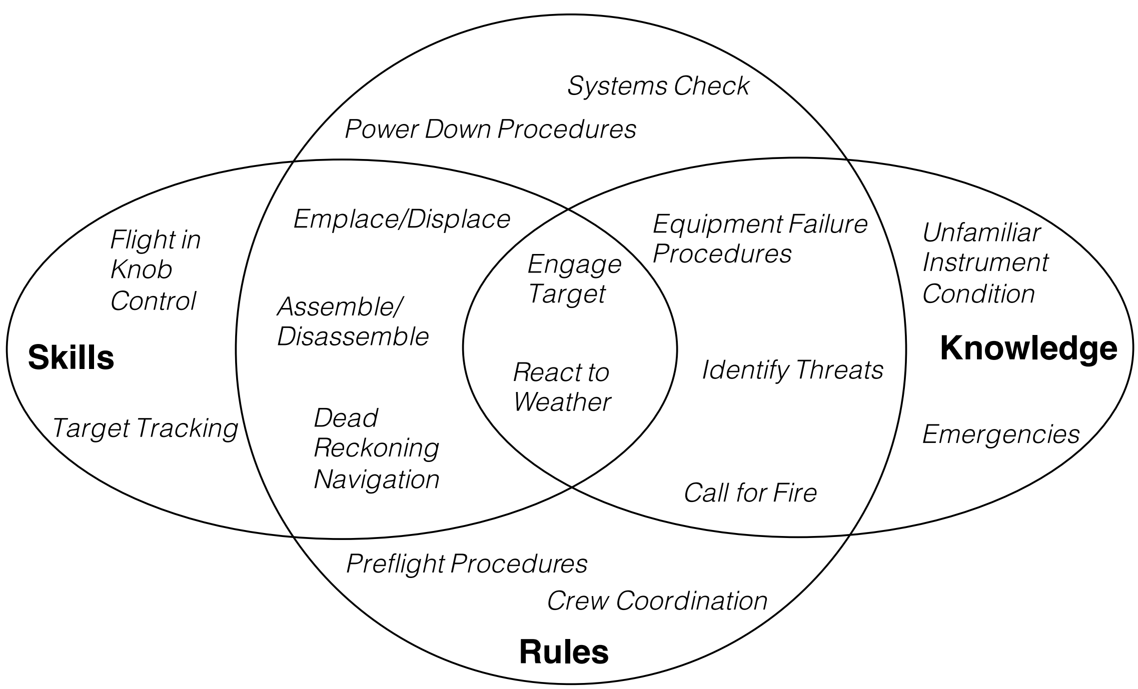


Figure 2. Varying allocation of operator behaviors from task keywords.

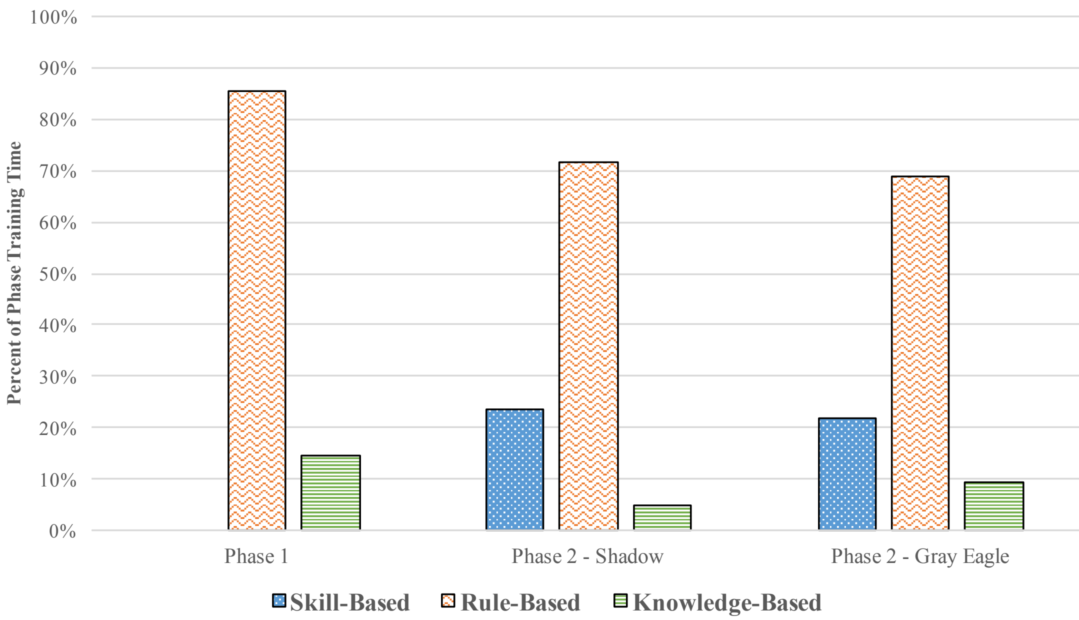


Figure 3. Percentages of overall training time for Skill-, Rules-, and Knowledge-based tasks across various phases of training.

Using this uniform time allocation method, it was determined that the distribution of SBT, RBT, and KBT fractions of the overall Phase 2 platform-specific training times were comparable. In both of the Phase 2 training programs, emphasis is on following procedures (RBT) while in the GCS and basic skills (SBT) to control the aircraft alongside the automation. Knowledge-based reasoning was primarily present during emergency, target identification, and weapons release (for the Gray Eagle) tasks, which occur during simulation exercises.

The categorization of these tasks into these different levels of abstraction allow for a more principled and objective assessment of what trainees in a UAS program need to master prior to graduation. This categorization is a key step in model development, which is explained in the next section.

Autonomous System Training Model (ASTM)

In order to capture SBT, RBT, and KBT as well as the influence of increasing autonomy, a model was developed called the ASTM (Autonomous System Training Model). Figure 4 represents a generic version of this model, which incorporates training requirements, the influence of increasing autonomy, both training and operational errors, and trainee attrition.

The primary output variable of ASTM is time to train, given that the Army wishes to optimize training both in time and resources. Thus, all variables that constitute the model must be framed in the context of time. So, in the training requirements section that specifically accounts for those SBTs, RBTs, and KBTs in various training modules, these tasks would be represented by how much time it takes to train people to master them. The training requirements section also accounts for the resources required to conduct this training such as instructors and classrooms, but again, they are represented temporally, so that, for example, more instructors would translate into a decrease in training time.

In the ASTM, trainees leave the training process in one of two ways, they successfully graduate or they fail. If they fail, trainees then have two options, they can recycle for remedial training or they are permanently removed from training. However, even after successful training, graduates' operational performance is considered in the model if they experience significant operational errors. This is a critical feedback loop as commanders can track operational errors and notify the training departments when they see repeated problems during actual operations. In theory, increased operational errors should increase overall time to train for humans, assuming no technology is introduced to reduce errors.

Indeed, during interviews of USMC Shadow operator in Yuma, AZ who participate in joint training between USA and USMC, it was revealed that prior to FY14, dead reckoning training had been removed from the syllabus as it was deemed not necessary. Once these newly trained Shadow operators reached the field, they began to have so many problems in basic navigation tasks, that the commanding officers of various squadrons insisted that the training be

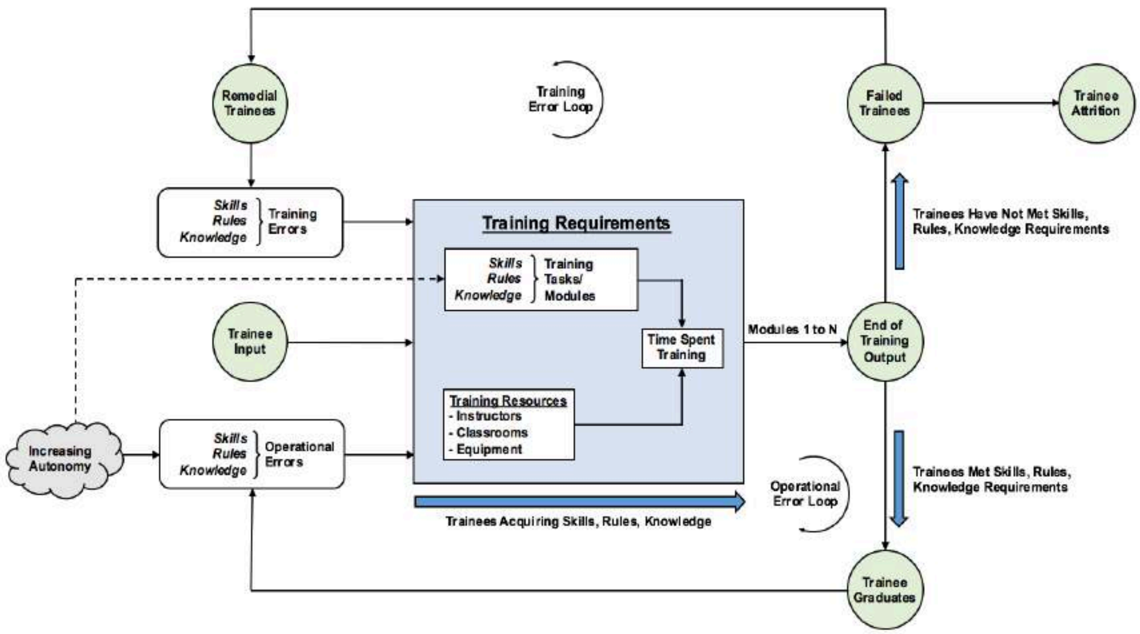


Figure 4: The general Autonomous System Training Model that incorporates training requirements, the influence of increasing autonomy, both training and operational errors, and trainee attrition.

reinstated, which occurred in FY15. This feedback prompted the inclusion of the operational performance feedback loop in ASTM.

One unique aspect to ASTM is that it can account for new technology insertions or improvements because of the skill, rule, and knowledge-based task representation (as outlined in the previous section). Because these increasingly complex behaviors can be executed by humans or autonomous agents, they provide a mechanism to explicitly study how increasing autonomy could affect training programs.

For example, in current UAS navigation training, there exists multiple modules to teach map reading, the use of the specific tools to plan paths, fuel computations, etc. This training includes significant SBTs and RBTs that take significant time to train. So if a new form of artificial intelligence was introduced for the task of path planning, all those SBTs and RBTs could theoretically be dropped from the training program, thus speeding up training.

The Army UAS Application of ASTM

In order to demonstrate how ASTM could be used to model Army UAS systems, the Army UAS (AUAS) ASTM was designed using a System Dynamic (SD) model. SD is a modelling method for understanding non-linear behavior of complex systems by taking advantage of causal feedback loops and stocks and flows (Forrester, 1961; Sterman, 2000). Various modelling approaches could be used to frame an ASTM model, but we elected to use the SD approach for Army UAS training programs due to its ability to capture quantitative and qualitative parameters (Cummings & Clare, 2015), such as operator behaviors, as well as its batching capability (Mikati, 2010). Moreover, the foundation of SD is centered around modelling the continuous, time-dependent behavior of a system (Sterman, 2000). Since training occurs as a continuous flow of batches that respond to causal relationships, SD modelling is well suited to capture the rate of trainees moving throughout a training program.

Prior work has shown that SD models can be applied to human-machine systems with accuracy (Clare, 2013; Gao, 2016; Gao, Clare, Macbeth, & Cummings, 2013; White, 2003). One problem with SD models, however, is that they are deterministic in nature and must take into account averages of parameters. In addition, SD models are poor at managing parameter variability (Brailsford, Desai, & Viana, 2010; Cummings & Clare, 2015), which makes them most applicable for understand average behaviors, which is reflective of the training environment we are modelling.

One goal of the AUAS ASTM is to model how the number and type of tasks and training time affect the number of UAS operator graduates and overall training times in the context of operator behaviors. Given that the overall training time for an individual is directly impacted by both the number of tasks and the requirements of the tasks, the AUAS ASTM needed to capture the impact of alterations in training programs on training time, including the recycling and attrition rates of trainees.

However, the inputs and outputs of ASTM depend on the nature of research questions, i.e., ASTM can generate the number of classrooms needed for given class sizes and training times as an output. Conversely, if the Army wants to investigate how quickly people could be moved through the system given a fixed number of resources, then the output would be class size. But for the purposes of this effort, the focus will be on modeling the output of a training program in terms of graduates, given a fixed number of classrooms, instructors etc,

To construct and validate this implementation of the AUAS ASTM, two datasets were used. The first, UAS operator training data from FY14, was used to initially design and parameterize the model and the second, data from the same UAS operator training program in FY15, was used for model validation. The primary difference between these 2 data sets is that the FY15 Phase 1 training program was reduced by ~1.5 weeks and the FY15 Shadow training program was reduced by ~2 weeks. Thus, an effective confidence builder for AUAS ASTM will be that it can accurately capture the change in the data set, which significantly different from the data used to build the model.

The general structure of the AUAS ASTM can be seen in Figure 5. The stocks (box variables) represent the trainees that move through training. Flows that connect the stocks are controlled by rate equations and are derived from parameters that are linked with causal arrows. Trainees move from one stock to the next as designated by the direction the flow arrow is pointing and the value of the rate.

The rate in which trainees flow between stocks is controlled by parameters that make up rate equations. SD models, such as AUAS ASTM, are time-dependent models that calculate new data points every time step, which is constant throughout the entire simulation. The AUAS ASTM calculates the flow rates via Euler integration method with 0.03125 week time steps, which was selected by testing the sensitivity of the model to varying time steps. It was determined that time steps less than 0.03125 weeks showed no variation in model outputs, and, thus, would serve as an accurate selection (Sterman, 2000).

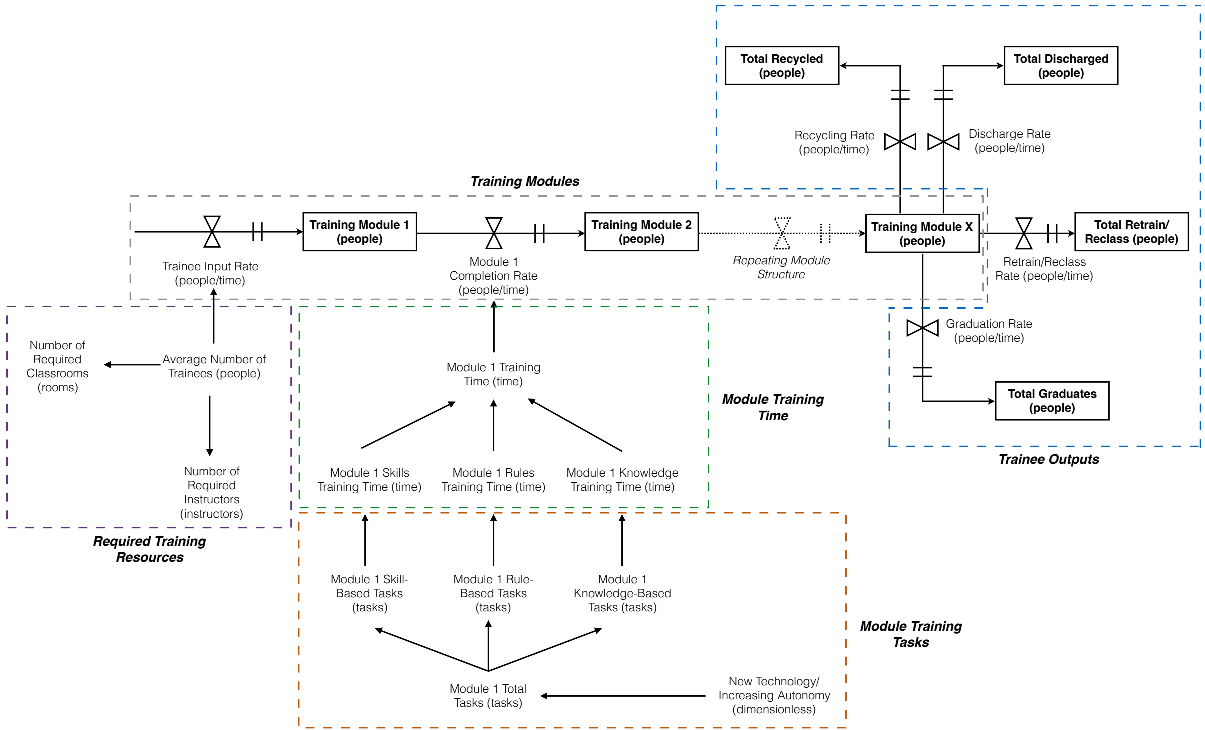


Figure 5. General structure of the AUAS ASTM with units noted in parentheses. The Repeating Module Structure arrow and valve represent the repetition of the Module Training Time and Module Training Tasks sections for the respective number of modules within a training program.

The structure of the AUAS ASTM is made of five distinct sections (Figure 5): Training Modules, Required Training Resources, Module Training Tasks, Module Training Time, and Trainee Outputs. The Training Modules, Module Training Time, and Module Training Tasks sections are repeated for each of the modules that are in the training syllabus. Inputs to this application of AUAS ASTM include the average number of trainees per group, frequency of new trainee group arrival, tasks per module, and module training times. The AUAS ASTM includes a number of variables including number of resources (classrooms and instructors), the number of graduating trainees, the number of recycled trainees, the number of discharged trainees, the number of retrain/reclass trainees, and the total training time.

The Training Modules section is made up of trainees that are moving through the training program. The Required Training Resources section provides the user with the ability to plan and calculate the number of classrooms and instructors required to match the number of trainees. The Module Training Tasks contains the tasks that the trainees must complete in each module. Those training tasks are converted into training time in the Module Training Time section. Finally, the Trainee Outputs section gives a breakdown of how many trainees successfully completed training as well as those that did not meet satisfactory requirements.

To account for the number of trainees that successfully graduate, AUAS ASTM estimates percentages for all possible trainee outcomes in each FY, including recycled, discharged, and retrain/reclass. Trainees that are recycled are those that are required to go back and repeat training from the beginning of the phase with an incoming trainee group. This occurs due to unsatisfactory performance on the tasks or missing too many days due to illness or personal issues. The discharged output are trainees that are dismissed from the U.S. Army due to negligence toward superior officers, legal issues, or repetitive attempts at graduating without meeting satisfactory requirements, which are a very small number.

The retrain/reclass output is reserved for trainees that are sent to train in other divisions of the U.S. Army. The graduating operators are the remaining trainees left after the recycled, discharged, or retrain/reclass outputs have been subtracted from the trainee group size. The training times for each of the modules are calculated by taking the total number of tasks per module (the Module Training Task section in Figure 5) and converting those tasks to training time (the Module Training Time section in Figure 5). Appendix B details specific elements and computations of AUAS ASTM flows.

AUAS ASTM Results & Validation

The purpose of the AUAS ASTM model is to understand how the selection of tasks, number of modules, and flow of people (including retraining needs and attrition) impact training times and the number of graduates from the training program. In order to validate that such a model is an effective training planning tool, we needed to measure the model's performance against real world data.

For the Army UAV training data, AUAS ASTM generates the number of graduates over one fiscal year in ~4 week intervals, which is equal to the frequency of new trainee group arrivals. The Army's monthly class start date drives the need for a batch modelling process. A single fiscal year (FY) was chosen as the primary metric of interest because U.S. Army UAS operator training operations are funded by Department of Defense (DoD) budgets. These budgets change on a FY basis and in doing so, can alter the demand for the number of UAS operators and the amount of time that can be spent to train UAS operators. Thus, by modelling only one FY at a time, the political impact on the number of trainees and training requirements is reduced. Two validation data sets were provided from FY14 and FY15.

The training data for both FYs provides details into the number of graduating, recycled, discharged, and retrain/reclassed trainees for both Phase 1 and Phase 2. Since the phases represent two distinctly different training programs, we elected to use two different model representations. Phase 1 results are depicted in Figure 6, which shows the actual graduation rates for FYs 2014 and 2015 (solid lines) as well as the predicted graduation rates as determined by AUAS ASTM (dotted lines), demonstrates that the model accurately represents the numbers of graduates.

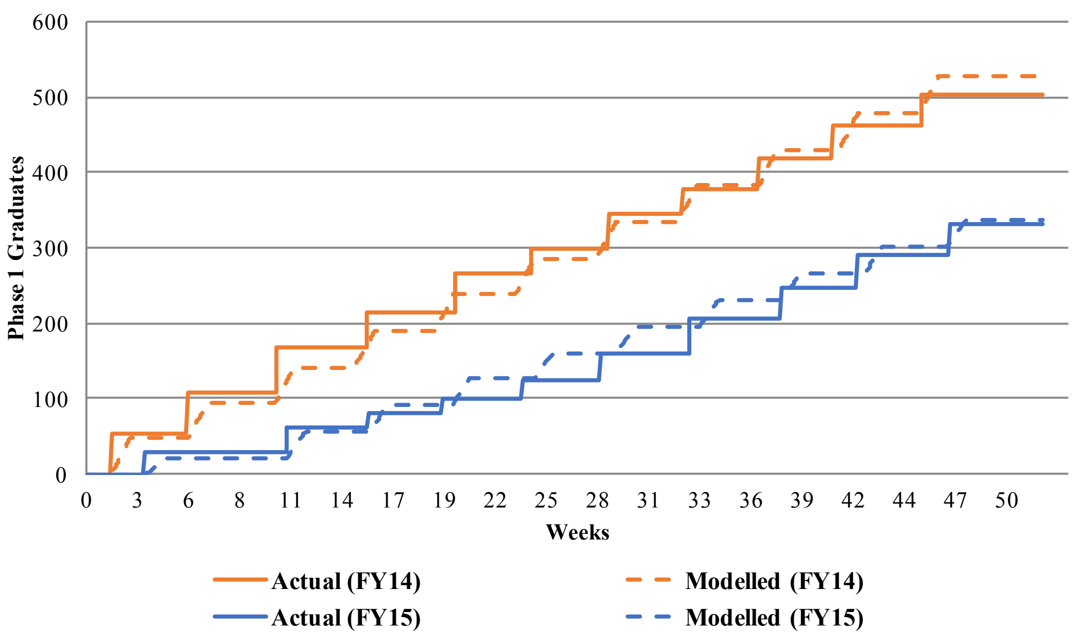


Figure 6. ASTM performance for Phase 1 graduates in FY14 and FY15.

To adapt the ASTM to run for FY15 data the structure remained the same, but the values of the number of tasks, SRK behavior to task allocation, output percentages, average incoming trainee group size, and time per SBT, RBT, and KBT were modified to match that of the FY15 syllabus, which changed from the FY14 syllabus and led to a change in the total training time. Many of the tasks between FY14 and FY15 were the same, but some tasks, such as dead reckoning, were added to the training requirements for FY15 due to trainees struggling to navigate in GPS-denied environments.

The maximum deviation of the FY14 data is 16.1% during weeks 12-14, and for the FY15 data the maximum deviation is 23.4% in weeks 30-32. These deviations between the number of operational graduates and predicted number of graduates are due to the fluctuating sizes of UAS operator trainee groups. Demand for UAS operators to fly missions alters over time, which is reflected in the numbers of incoming students. The model assumes constant incoming class numbers, leading to some model error.

At the end of the annual cycle, the AUAS ASTM error was 4.84% in FY14 and in FY15, 2.02%. The ASTM was also able to adapt to changes in the training program as illustrated by the model's ability to accurately reflect the decrease in Phase 1 output in 2015, driven by changes in the tasks (and thus in the required operator behaviors) and overall Phase 1 training time.

For Phase 2, the ASTM performed well, but not quite as well for Phase 1 (Figure 7). For the Shadow operators, ASTM's average errors were 0.17% and 4.31% for FY14 and FY15, respectively. However, the maximum deviations were 29.5% and 10.4% for FY14 and FY15, respectively. The percent error for Phase 2 Gray Eagle number of graduates was 1.30% for FY14 and 5.59% for FY15, with maximum deviations of 17.4% and 43.9%, respectively.

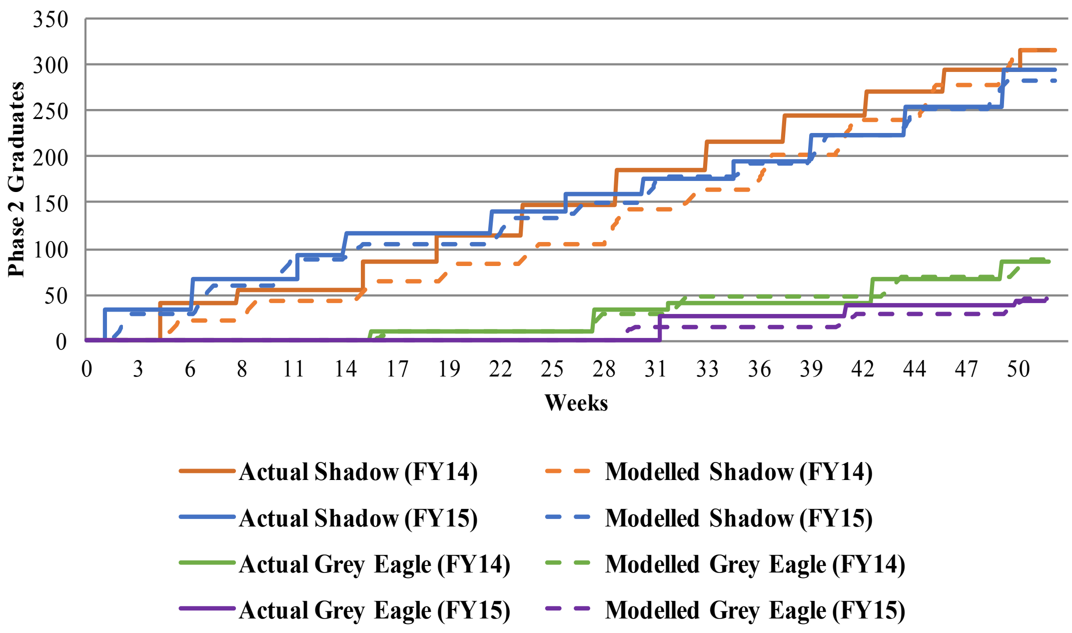


Figure 7. AUAS ASTM performance for Phase 2 Shadow and Gray Eagle graduates in FY14 & 15.

While Figures 6 and 7 depict the graduation rates, the AUAS ASTM also produces other outputs for recycled, discharged, and retrain/reclass. Table 2 shows an overview of the errors from the outputs in FY14 and FY15. The errors reported in Table 2 can be attributed to inexactness in reporting. This information was provided by the Army and could not be directly verified. The personnel who assisted us were given instructions on how to assign people into graduated, recycled, discharged, or retrain/reclassed categories, but due to information sensitivity concerns, we could not directly access the data. Thus, there is likely error in how people were assigned to the various categories.

Table 2. Percent error between actual vs. model outputs for FY14 & FY15 UAS operator training

ASTM Output	FY14	FY15
Phase 1 – Common Core Recycled	0.63%	13.41%
Phase 1 – Common Core Discharged	6.25%	16.5%
Phase 1 – Common Core Retrain/Reclass	0.80%	13.22%
Phase 1 – Common Core Graduates	4.84%	2.02%
Phase 2 – Shadow Recycled	1.56%	8.73%
Phase 2 – Shadow Discharged	1.71%	8.75%
Phase 2 – Shadow Retrain/Reclass	6.50%	9.00%
Phase 2 – Shadow Graduates	0.17%	4.31%
Phase 2 – Gray Eagle Recycled	1.72%	2.88%
Phase 2 – Gray Eagle Discharged	0.50%	2.89%
Phase 2 – Gray Eagle Retrain/Reclass	5.14%	0.00%*
Phase 2 – Gray Eagle Graduates	1.30%	5.59%

*The 0% error reported for Phase 2 – MQ-1C Retrain output in FY15 is due to no trainees retrained/reclassed that year.

Additionally, the high maximum deviations in both the Phase 1 and Phase 2 graduates are likely partially attributed to the difference in expected training time and operational training time. While the Army's UAS operator training syllabus specifies an expected time to complete the program, actual completion times are affected by holidays and various disruptions in daily schedules. In addition, the size of the trainee groups that move through the module stocks in the AUAS ASTM is constant, as it is an average group size calculated from the entire FY. Operationally, the size of the trainee groups fluctuates throughout the year due to demand for UAS operators and the number of Soldiers completing basic training. As discussed previously, one of the downsides to using SD is the required averaging of parameters instead of sampling from a distribution.

AUAS ASTM Sensitivity Analysis

For developers of training programs, it is important that the parameters that most influence training efficacy are highlighted. To this end, a sensitivity analysis is necessary for any model to understand where potential errors are present and where the boundaries exist for model adaptation. An essential step towards validating any human-systems performance model is to analyze the performance of the model when exogenous parameters fluctuate due to uncertainty (Forrester & Senge, 1980; Sterman, 2000). A sensitivity analysis addresses the question of how uncertainty in estimated parameters affect the final model output (Sterman 2000).

For the AUAS ASTM, a numerical sensitivity analysis was chosen to measure model robustness under +/-10% univariate perturbations of the exogenous parameters, which are those parameters that are constant values and do not have other parameters influence them. Perturbations less than this did not capture the variability in training time since due to the discrete time steps in which the ASTM runs (0.03125 weeks). The +/-20% sensitivity tests resulted in identifying the same variables that were identified with the +/-10% test.

Two plots for FY14 are shown in Figures 8 and 9 for the percent change in the total number of graduates from FY14 Phase 1 (Figure 8) and the graduation time of the first fully trained FY14 Phase 1 group (Figure 9) caused by the perturbations. These variables were selected for the deeper analysis since the Army is interested in increasing their training throughput, and so overall number of graduates and training time reflects this desire. Phase 1 is highlighted here since it is the core stage that affects all training pipelines, but sensitivity models were constructed for both phases.

Perturbations of task parameters resulted in no significant change in the number of Phase 1 graduates. However, as seen in Figure 8, perturbing the *Average Phase 1 Monthly Group Size*, *Phase 1 Recycling Percentage*, *Average Phase 1 Rollover Group Size*, *Phase 1 No Show Percentage*, *Phase 1 Retrain/Reclass Percentage*, and *Phase 1 Discharge Percentage* parameters was found to alter the number of Phase 1 graduates. The *Average Phase 1 Monthly Group Size* was found to be the most sensitive parameter (~ +/- 8% of the modeled FY14 number of graduates), while the *Phase 1 Recycling Percentage* was determined to provide the second most variability (~ +/- 3% of the modeled FY14 number of graduates).

The likely reason the *Average Phase 1 Monthly Group Size* had the greatest impact is that this parameter controls the number of people that enter the model, i.e. the batch size. By altering this parameter, the size of the trainee groups will be altered accordingly. Less obvious, however, was the importance of the recycling percentage. Recycling is a common training occurrence in that when a trainee fails a module, they are given an opportunity for remediation. In FY14 for example, approximately ~20% of incoming students required some level of remediation in the form of recycling. This number of recycled trainees in any given year is much greater than the number discharged or reclassed.

This is an extremely important finding in the AUAS ASTM model application in that while it is important to accurately represent the number of incoming students, the model is highly sensitive to fluctuating recycling rates, which are not well studied or understood. Very little data exists to connect the recycling rates to the course content, which in our experience was anecdotal. However, most instructors we spoke to also complained about the time needed for remediating the recycled students. It is clear from the modeling effort that recycling represents a significant source of uncertainty both in the model and likely in real life, thus more work is needed in both stabilizing the recycling rates and determining the root causes of such recycling to achieve a more predictable output of graduates. Remaining percentage and group size parameters in this sensitivity analysis were found to have less than +/- 2% effect on the modelled FY14 number of graduates.

The second sensitivity analysis plot in Figure 9 shows the impact on the time of the first graduating group for Phase 1 in FY14. Parameters that most influenced the training time were numbers of tasks and average time to complete tasks, with the number of tasks in Module C (Ground School) having the greatest effect on the overall training time (~ +/- 4.25-4.5% of the FY14 Phase 1 training time). During this module, all of the lessons are made up of RBT, thus, the *Phase 1 Module C Rule-Based Task Fraction* and *Phase 1 Module C Week: Rule* parameters had the same ~ +/-4.25-4.5% impact on the FY14 Phase 1 training time. The perturbation of the remaining task and time parameters resulted in little variation in the total Phase 1 training time.

It is important to note that Figure 9 represents the variability in the training time output as a result of perturbing a particular variable. Thus, the variation in Phase 1 Module C rule based tasks demonstrates that if trainees struggle in this phase, this will likely cause much more variability in overall training times. Other than Phase 1 Module C, only Phase 1 Module A rule based tasks also appeared to significantly affect training times. This is an important finding as it provides clear guidance to training development personnel as to where their quality improvement efforts would provide the greatest benefit.

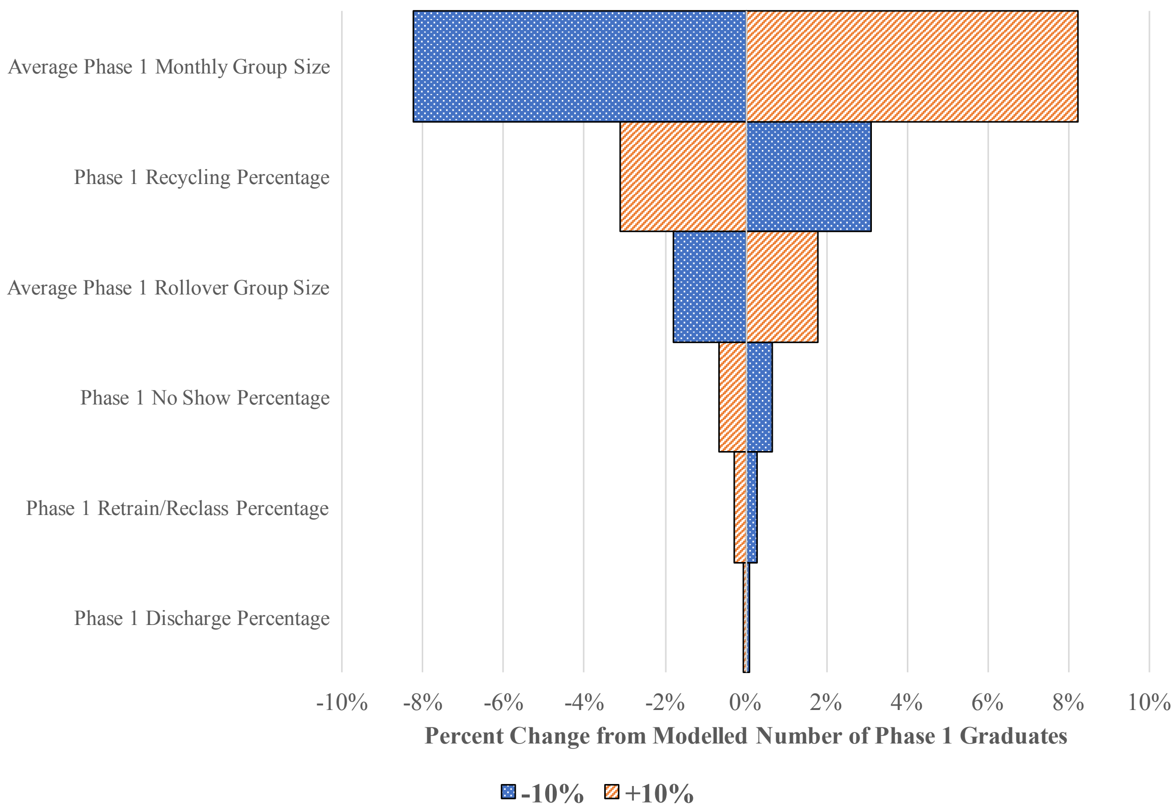


Figure 8. Percent change of total number of FY14 Phase 1 graduates with a +/-10% perturbation of exogenous variables. Parameters not in the plot had no effect.

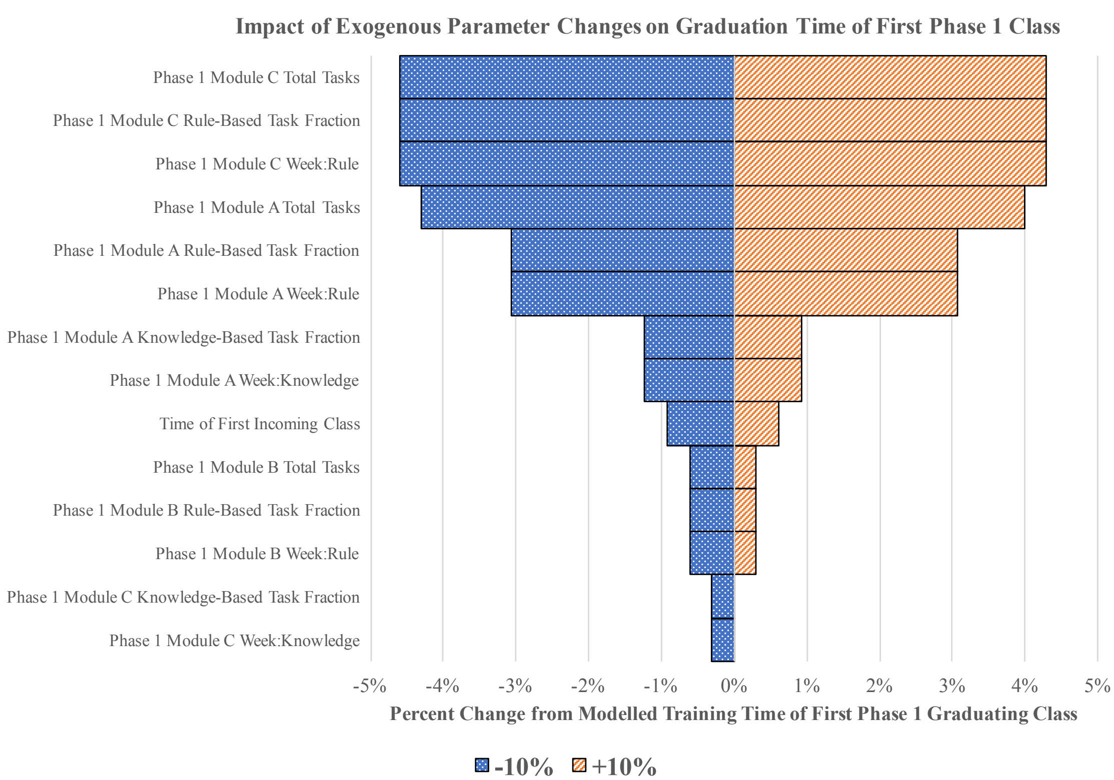


Figure 9. Percent change of graduation time of the first fully trained FY14Phase 1 group with a +/-10% perturbation of the exogenous variables. Parameters not in the plot had no effect.

Model Conclusion

An adaptable method for modeling the training of operators of autonomous systems has been designed and initially validated for UAS operator training operations. The current version of the ASTM has the capability of predicting numbers of graduates and training times with changes in the numbers and types of tasks, number of incoming trainees, and fractions of attrition, retrain, and graduating graduates. This model has the potential to allow training designers to proactively determine how changes in resources like classroom size, number of instructors, and attrition rates affect overall training projections. However, further data and applications are required that include the number of operational instructors, classrooms, and training equipment.

Through the application of the ASTM to a specific Army application, the Phase 1 and 2 training programs for UAS operation (called the AUAS ASTM), the modeling approach was validated with FY14 and FY15 training data, although the Phase 1 model was more accurate than the Phase 2 model. Given these results, a more in-depth sensitivity analysis demonstrated that overall throughput numbers are most affected by how many people start each class, but also by the numbers of people requiring remediation (known as recycling). Overall training time was most affected by variability in Module C training, which focused on UAS Ground School, especially for rule based training. These results demonstrate that if the Army wants to produce people both faster and with more predictability, they need to address why and how often people are being recycled, as well as whether Module C training is causing significant variability in their process.

The unique element of ASTM is how it links operator behaviors to training through the SRK framework. Classifying tasks taught during training can aid developers of operator training programs to better understand the types of tasks that could and should be trained and their influence on training throughput and resources. Additionally, this method allows designers of unmanned system operator training programs to use the SRK taxonomy as a way to identify where autonomy has the potential to influence future system development (through which SRKs are automated), as well as what tasks could and should be left for human operators.

As the Army inserts more autonomy in UAS, such as automated path planning, one question that is raised is how should training programs be adjusted to provide the best, but most efficient, level of training? By understanding which SRK tasks the automation affects, the AUAS ASTM could then demonstrate how the addition or reduction of tasks due to new technology introduction affects overall training throughput and time. However, this approach highlights one important limitation to the ASTM model, which is that the SRK tasks are considered to be independent and module-based.

For example, if a new technology is added that replaces a set of skills and rules required of a human, like automated path planning, the current AUAS ASTM model simply assumes that all related human training requirements are zeroed out and there is no effect on error rates. The original ASTM model in Figure 4 does recognize there could be a relationship with various SRK tasks and errors, however given that no such data was available for the AUAS ASTM, it does not currently reflect this.

One important related question is whether the removal of a particular group of tasks due to automation of autonomy ultimately negatively affects the operator because they miss fundamental training? So in the case of automated path planning, if mission planning training is shortened because of the addition of an automated path planner, do operators miss out on important skills that would have added to their overall core base of knowledge? While anecdotal stories from the USMC concerning the problems with dead reckoning discussed earlier would suggest that there could be, there is no current data that explicitly links SRKs to autonomy, and ultimately operator performance.

In order to address this gap in the modeling process, we elected to design and conduct an experiment to potentially begin to address the relationships between autonomy, training of SRK tasks, and error rates, which is discussed in the next section.

Elucidating the Relationship between SRK Training and Autonomy

One common assumption is that advanced autonomy allows less training to occur, as functionalities are shifted from human to automation. While the general ASTM model allows for errors to be introduced into the model because of this potentially flawed assumption, there is no existing data to inform how such changes could cause increased error rates. It is not clear whether skill-based training is a fundamental requirement for developing more global situation awareness that could be needed in the advanced knowledge and expert stages of reasoning (Cummings, 2014). In one step to help address this gap which also would directly inform the model developed in the previous section, an experiment was designed to test how the presence or absence of skill-based training affects operator performance, especially in the presence of advanced automation, which in this case was Automated Target Recognition (ATR).

In this experiment, ATR is defined as an automated computer vision system that searched a predefined area to find a suspected target and present this target to an operator for either confirmation that the correct target was found, or the operator has the ability to search for what he or she felt was the right target.

The basic experiment design included dividing subject groups into those that received skill- and/or rule-based target search training in a multiple UAV control task. These groups also would experience the presence or absence of ATR in a target search task. The goal was to see how different levels of training would affect human performance given different levels of automation. One reason many agencies state that increased automation is beneficial is that, in theory, it should reduce training costs since there are one or more functions that automation takes over. However, there is some military evidence that shows that there can be tangible performance costs when shortcuts in training occur.

The USAF recently demonstrated that its UAV pilots perform best when they have received just 2 years of regular flight training in addition to their UAV flight training, as opposed to those who receive no flight training. The group with basic flight training performed even better than seasoned pilots with thousands of hours who transitioned to flying UAVs after multiple tours of duty in actual aircraft (United States Air Force Scientific Advisory Board, 2011). These young drone pilots receive significant basic training as actual pilots that will never be used as drone pilots, but such training provides them with situation awareness and perhaps confidence that they would not have received otherwise.

But this same USAF study also demonstrated that too much experience also carried a negative benefit, likely due to a concept known as the negative transfer of training. It is possible that if a person has too much experience in another domain, that this training can interfere with the ability of the person to learn new training correctly in a similar domain (Dahai, Blickensderfer, Macchiarella, & Vincenzi., 2008). Thus, the goal of any training program should be to train to the optimal level, as there can be too little or too much. And also, it is best to select a group of trainees that will not experience negative transfer of training.

In the experiment for this effort, training was designed to be shorter for those subjects with just rule-based training than those with skill- and rule-based training. Given the two different levels of automation in the search task and the results seen in the USAF study, we should see higher task performance with both skill- and rule-based training, as opposed to just rule-based training, regardless of the presence of ATR. Indeed, those people with skill and rule-based training should achieve peak performance with ATR since they should have increased cognitive capacity since automation is reducing their workload by helping them find targets.

We elected to introduce a nested variable into this experiment which was whether the ATR was correct or not. Automation complacency is a known problem for operators supervising UAVs (Cummings, 2004), thus it was important for us to determine if automation bias existed, and if there was any interaction between type of training and degraded autonomy.

The last experimental factor we wanted to represent was that of knowledge-based reasoning. Given that knowledge-based reasoning is linked to experience (Rasmussen, 1983), the only way we could represent this level of "training" would be to use people with significant experience in UAVs and compare their performance to people without significant experience. It was our original intent to use Army UAV operators vs. college students to represent this condition, since the groups are approximately the same age.

The details of the experimental design are summarized below.

Experimental Design

Hypotheses

1. Participants without ATR will experience the highest workload and correctly identify fewer targets, but those that receive skill + rule training (as opposed to just rule training) will have higher situation awareness as measured through correct responses to a secondary workload metric (chat messages), and fewer incorrect target identifications.
2. The subject groups that receive only rule training with ATR will demonstrate the least workload levels via utilization, and but will incorrectly identify more targets as a function of automation bias when the ATR fails.
3. The best performance in terms of correct targets identified will be seen by the group that has Skill+Rule training but with ATR (which should reduce workload).
4. Army personnel should perform better in terms of higher situation awareness and target identification since they are actual UAV operators and have advanced basic knowledge.
    a. One caveat to this hypothesis is that there are no true experts in the control of multiple UAVs from a single ground station, which is the testbed used in this experience. Thus, there is the possibility that novices could do better than the Army personnel due to negative transfer of training.

Statistical Model

The original experiment was designed as a between-subjects 2x3(2)x2 statistical model, which represents knowledge groups x ATR quality nested in autonomy training x trial, described below:

- Knowledge groups = people with UAV experience vs. those without, i.e., Army UAV operators vs. college students
- Autonomy training = skill- + rule-based search training w/o ATR, skill- + rule-based search training w/ ATR, rule-based search training w/ ATR. This condition represents varying autonomy with varying training. There is no rule-based training w/o ATR since for the purposes of this experiment, rule-based training only exists when ATR is present.
- ATR quality = This variable represents working vs. a failed ATR. In all factors with an ATR, 70% of the ATR-enabled identification were true positives, and 30% were false positives. In this experiment, the ATR always found something, so there were never any misses or false negatives. This is a nested condition since it only applies to the two factors with ATR.
- The last factor was trial number, where each subject experienced the same experimental conditions for another session. This was included to increase statistical power.

Dependent variables would be overall number of targets correctly identified, subjective workload, utilization (percent busy time), average time spent searching for targets, response time to chat messages (as a secondary workload metric), and correctness of responses to chat messages (a situation awareness proxy). In addition, given the presence of faulty ATR, for those people with ATR, the number of incorrect identifications as compared to those people with no ATR would likely yield important insight into the effect of automation bias.

Participants

The original experiment was designed for 12 Army personnel and 12 college students that theoretically have different levels of knowledge due to their backgrounds. The goal was to have 8 people per autonomy training block (half Army UAV operators/half college-aged students).

Testbed

The software that this experiment uses is called RESCHU (Research Environment for Supervisory Control of Heterogeneous Unmanned vehicles). This test bed, described in more detail below, allows a single operator the ability to supervise multiple UAVs. RESCHU is a discrete event simulator that can be used to test operator methods and strategies for controlling, scheduling, and planning missions using multiple unmanned vehicles.

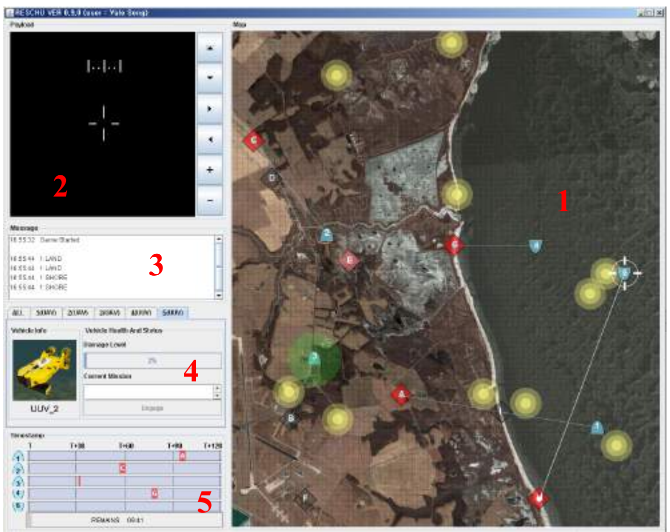


Figure 10: RESCHU Interface

Figure 10 shows a screenshot of the RESCHU interface with its five components labeled. The five components consist of: 1) A map of the simulated region in which the vehicles are operating in, 2) An image from the camera onboard the simulated unmanned vehicles, 3) A chat message board that gives up-to-date information pertinent to the mission, 4) Unmanned vehicle status window, and 5) A schedule for future unmanned vehicle actions. The subjects of the experiment will be required to monitor the map (1), the camera image (2), and the chat messages (3) to assist in target search and identification. For this experiment, subjects will monitor four UAVs simultaneously where each have their own respective schedule in (5), damage assessment in (4), and camera image in (2).

The goal of operators using RESCHU in this experimental setting is to visit as many targets as possible and successfully identify objects through the camera when the UAVs reach targets. In order to accomplish this, RESCHU essentially has two different subtasks embedded in its design. The first subtask is successfully navigating multiple vehicles to different targets in order to search the targets for the desired object of interest, which primary involves looking into sections 1 and 5 in Figure 10. The second subtask is then to engage the UAV's camera (section 2, Figure 10) and look for a target, who is assigned by a "supervisor" (section 3, Figure 10), which is an actually just a bot telling the operator what to search for, i.e., a red car, a helicopter landing pad, etc.

Skill vs Rule Training

In order to test the interactions between increasing autonomy and skill and rule training, we needed a scenario where skills would be required under the lower autonomy case of no ATR, which would not be needed in scenarios where operators had access to ATR. We decided that the best way to experimentally control this was to use a new input device that required dedicated training to be good at using, especially in the search task, but could still be used with some on-the-job training.

Figure 11 shows this device which is called the Kensington Expert Trackball Mouse®. It requires users to learn a new manual control skill in the control of the trackball which is much faster than a traditional mouse, and participants also had to relearn which buttons map to various functions. This device is very unfamiliar to most people and requires time and effort to learn to use. Such an equipment change is reflective of real world decisions to add hardware to a new system that requires a skill set which takes time to learn.

Figure 11: Kensington Expert Trackball Mouse ®

For this experiment, those people in the skill + rule training group would be given ~20 minutes of dedicated training to learn to use the Shark® mouse in an abstract training task. An abstract training task was needed to allow participants to become experts at using the mouse without giving them significant experience inside the RESCHU environment. Thus, we developed a Fitts's Law training environment (FLTE) (Figure 12), which embodies the well-known Fitts's Law relationship. Fitts's law states that the movement time (MT) required to rapidly move to a target area is a function of the distance and the size of the target. (Fitts, 1954; MacKenzie, 1995)

$$\text{Movement Time} = a + b*ID \qquad ID = \log_2 (D / W_e + 1) \qquad \text{(Equations 1 \& 2)}$$

where ID is the index of difficulty, D is the distance to the target for selection and W_e is the effective width of the target. Coefficients a and b are the slope and intercept coefficients, determined through empirical tests conducted in a pilot study.

In the training environment in Figure 12, participants in the skills + rule condition (with or without ATR) conducted 6 blocks of 75 clicks each in FLTE with 30s breaks in between blocks. This protocol has been shown to effectively train people learning a new mouse input device to relatively stable levels of performance (MacKenzie, Kauppinen, & Silfverberg, 2001).

People in the rule-based training group received no specific training on the new mouse, but were able to spend 15 minutes in the practice session (described in detail in the next section) acclimating to the new device. People who received mouse training also had the same 15 minutes of RESCHU training. In RESCHU, participants with no ATR would have to pan and zoom to find the target much more than people who had ATR.

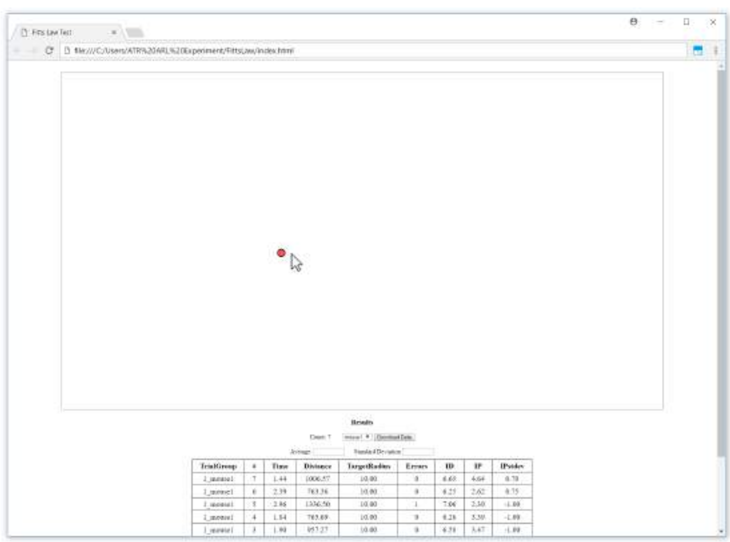


Figure 12: Fitts's Law Training Environment, the 10 pixel target dot moves randomly across the screen.

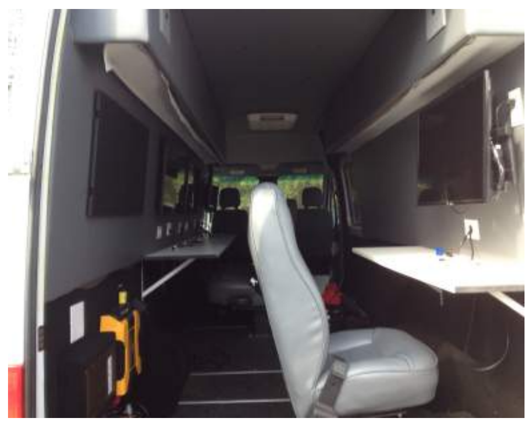
The rule-based training consisted of the procedures needed to successful search and identify each target assigned. These procedures included engaging the camera (which was different if ATR was present), searching for a target, and then the final steps for submitting a final answer. All participants received rule-based training, although the ATR condition required slightly different rules.

Conduct of the Experiment

The original experiment was designed to last for 60 minutes total, with 30 minutes for briefing and training and 30 minutes for experimental testing (2 missions, 15 minutes per mission). This experiment was designed to take place in two places, Duke University for the student population and either Fort Bragg, NC or Fort Campbell, KY for the Army operator population. The Humans and Autonomy Lab has a mobile command center van (Figure 13) that is equipped with multiple displays and wireless communication capabilities in which the training and testing will take place. Given the need to run experiments in two locations (Duke + an Army location), all experiments were to be run in this van using the two OptiPlex 5050 Minis with 4GB RAM and an Intel Core i5-7500T processor. They are hooked to Samsung® displays (DM32e model, 32”, 1920x1080 resolution).

Immediately following training, each subject would participate in two, 15-minute simulated missions. The objective of these missions is to identify as many targets as possible. The subjects will supervise four UAVs during the missions, representing a future desired Army capability. In RESCHU, the navigation for each vehicle happens automatically, but the human must monitor each vehicle to make sure they do not fly into a pop-up threat area. They also must replan the routes accordingly. The chat message board in Figure 10 alerts the participants when a vehicle is approaching a target and to inform them when the camera will be online. In the ATR condition, subjects will be presented with what the automation believes to be the target. The error rate of the automation is designed at ~30%, which has been shown to be a problematic area for human trust of automation (Wickens & Dixon, 2007).

(a) HAL Mobile Command Center Exterior



(b) HAL Mobile Command Center Interior

Figure 13: HAL Mobile Command Center

After each test session, a brief post-experiment survey will be administered to assess participants’ perceived workload (Appendix E). The detailed checklists used for each of the three different factor levels are included in Appendix F. The training slides for the different conditions are in Appendix G.

Results

Unfortunately, the planned experiment did not materialize due to the inability to gain access to Army operators before the end of the study. However, one group of students used a similar protocol for a class project in the Human Robot Interaction graduate class. Instead of testing the trackball, they tested a vertical mouse on 15 participants in three testing groups (normal mouse, vertical mouse with the same Fitts's training as described above and the vertical mouse without training). This study demonstrated that there was a tangible benefit of training on multiple metrics including contingency planning, task management, and information seeking and filtering (Saxena & Wang, 2017). This means that those participants who had to train for a significant period of time on a new unfamiliar device performed better in many respects to not only those people who did not have any training (which is expected), but this group also performed better than those people who used the old, more familiar mouse. Thus, a group of 'novices' with a new technology outperformed the 'experts' on many tasks, even though the experts should have been able to commit additional cognitive resources since their cognitive workload was lower. These results suggest that the act of training – even in a very abstract environment - adds additional benefit beyond just the act of teaching a specific skill. Indeed, these results mirror those of the US Air Force's study cited earlier where those people with training in undergraduate flight training did better than either complete novices or advanced experts in UAV operations.

It remains to be seen whether this positive effect is due to increased self-confidence since participants may feel more prepared for a cognitive test after having undergone recent training. Such increases could be attributed to a heightened state of awareness that the training induces, suggesting that there may be some benefit to intense warm up mental exercises. Testing the temporal influences of this effect are an important future area of inquiry. In addition to a temporal effect, it will also be important to better understand the effect of training content, i.e., is increased performance due strictly to additional training time or is there some element of the training material that is particularly important?

One other interesting result in the Saxena and Wang (2017) study was that most of the performance benefits of the training were actually not for the specific search task best supported by intense mouse activity. Thus, the training had an unintended positive benefit for other tasks that required significant less mouse input. More research is needed to look at this finding in more detail.

In another related study that used the same experimental test bed and protocol outlined previously, but that used college students and FAA Part 107 commercial UAV pilots (Cummings & Huang, in progress,) there were several intriguing results.

When looking at the overall performance comparisons between the three experimental groups (skill + rule training without ATR, skill + rule training with ATR, rule training with ATR), there were no statistical differences in success rates of finding targets and successfully navigating the vehicles around hazard areas. However, while not statistically different, participants with just the rule-based training had the overall lowest error rates of 5%, while those with skill + rule training w/o ATR were at 9% and skill + rule training w/ ATR were at 11%.

Commercial pilots without ATR took the longest to search each target, which was statistically higher than all other categories, and likely because of this delay, this caused the other UAVs to experience longer wait times for service. Commercial pilots without ATR took, on average, 10s more to search than students, and 15s more than their commercial counterparts who had ATR.

These temporal results are interesting because without the help of the automation, the commercial pilots could not perform as quickly as their counterparts or the student pilots who had automated target search assistance. The student pilots without ATR performed the same as their peers and the commercial pilots with ATR so the student pilot group with no ATR was much more resilient and able to keep up. This result suggests that the commercial pilots, who generally had more experience with supervising real UAVs, struggled in the temporal aspects of multiple vehicle control when they did not have ATR. They still performed the same in terms of overall accuracies, but the commercial pilots without ATR just took longer. This result could be an indication of an element of negative transfer of training, in that because the commercial pilots were not used to supervising multiple vehicles, they had difficulty with dealing with the multiple tasks and added workload, but only when they did not have automated assistance.

When looking just at those people that had ATR, the FAA Part 107 pilots, who were higher in KBB, were almost three times more likely to correctly detect targets than the students. Moreover, operators with *reliable* ATR were 7.5 times more likely to be successful, so consequently people's odds of getting a wrong answer under unreliable ATR were very high. Thirty percent of all operators with unreliable ATR got wrong answers, as compared to only 5% of errors when the ATR worked correctly. In comparison, operators with no ATR had an error rate of 9%. Most of the errors made in the unreliable ATR conditions were caused by student pilots (N=12). Only one commercial UAV pilot with reliable ATR failed to accurately complete a search task.

These results clearly indicate a problem with deskilling and complacency when operators have too much automation, which has been shown in other supervisory control domains (Ferris, Sarter, & Wickens, 2010; Parasuraman, Sheridan, & Wickens, 2000). When the automation was only 70% reliable, so were the humans, which could be extremely dangerous in safety-critical settings.

Overall, the Cummings & Huang study found that additional skill training in the presence of automation primarily benefited a subset of population, the novices. With both skill and rule training but no automation, error rates on a target search task were slightly higher as compared to those of people with the same training but with access to ATR. However, there was no clear benefit to having skill and rule training in terms of reduced errors in the presence of automation, and thus, a significant challenge remains in terms of preventing automation complacency and deskilling, especially as more automation is brought online.

Evaluating Training through a Machine Learning Approach

For this effort, a side task to determining the best way to model training was to explore the utility of using a machine learning approach for training evaluation. To this end, we elected to use a RESCHU-derivative testbed to determine if a hidden markov modeling approach, which is a machine learning technique, could accurately capture operator strategies. If there is an automated way to determine what operator strategies are being used, then this opens the door to determine whether trainees are developing strategies as they should, and then indicate where such strategies might be inefficient.

Hidden Markov Models

A Markov model is a stochastic model of state transitions in the state space, where all state transitions are observable (Asmussen, 2008). Many studies have used Markov models to investigate low-level human actions (Galata, Johnson, & Hogg, 2001; Pentland & Liu, 1999). However, Markov models capture only low-level interactions between human operators and control systems, instead of high-level human cognitive states. An extension of the Markov model is the Hidden Markov model (HMM), which is a stochastic model that describes a Markov process with some states and variables that are not observable (Rabiner & Juang, 1986). While system states and state transitions are observable for Markov models, in a Hidden Markov model, system states are not directly observable (thus are 'hidden') and the only observable variables are emission probabilities that are determined by hidden system states.

An HMM model can represent both higher-level human operator cognitive states and lower-level operator interactions with human supervisory control systems. For instance, in supervisory UAV control scenarios, hidden states of a HMM can represent operators' strategies in high-level UAV tasks, and observable emissions can represent low-level interactions between operators and UAV control interfaces. HMMs have been used previously to develop human operator behavior models (Boussemart, Cummings, Las Fargeas, & Roy, 2011; Suzuki & Jansson, 2003; Suzuki et al., 2005), but none of these previous efforts attempted to determine actual strategies, particular in the control of UAVs, which is the focus of this effort.

The SA-RESCHU experiment

In order to determine if HMMs can detect UAV control strategies in a useful manner, a data set that included a significant set of observable interactions with a UAV system was needed. We elected to use a data set that was recently generated for the Navy that looked at how well operators could assist in the detection of potential UAV hacking events. The interface used in this study is called Security-Aware Research Environment for Supervisory Control of Heterogeneous Unmanned Vehicles (RESCHU-SA), which is a Java-based simulation platform for a single-operator with multi-UAV supervisory control scenarios. It is a derivative of the RESCHU interface in the planned experiment detailed in a previous section.

RESCHU-SA provides capability for simulating UAV GPS spoofing attacks, in which hacked UAVs deviate from their originally assigned path and move towards other unexpected destinations, and it is the operator's job to determine if suspected path deviations are due to hacking or are a false alarm. Such a method of hacking detection is possible since the camera video feeds on UAVs are a separate one-way transmission from the UAV, while the navigation/GPS system is a transmit and receive system that can be relatively easily spoofed if not protected. Thus, the location of a UAV on a map may not be correct if it is GPS-derived, but the transmitted view from a downward camera is generally legitimate, so the two can be cross referenced.

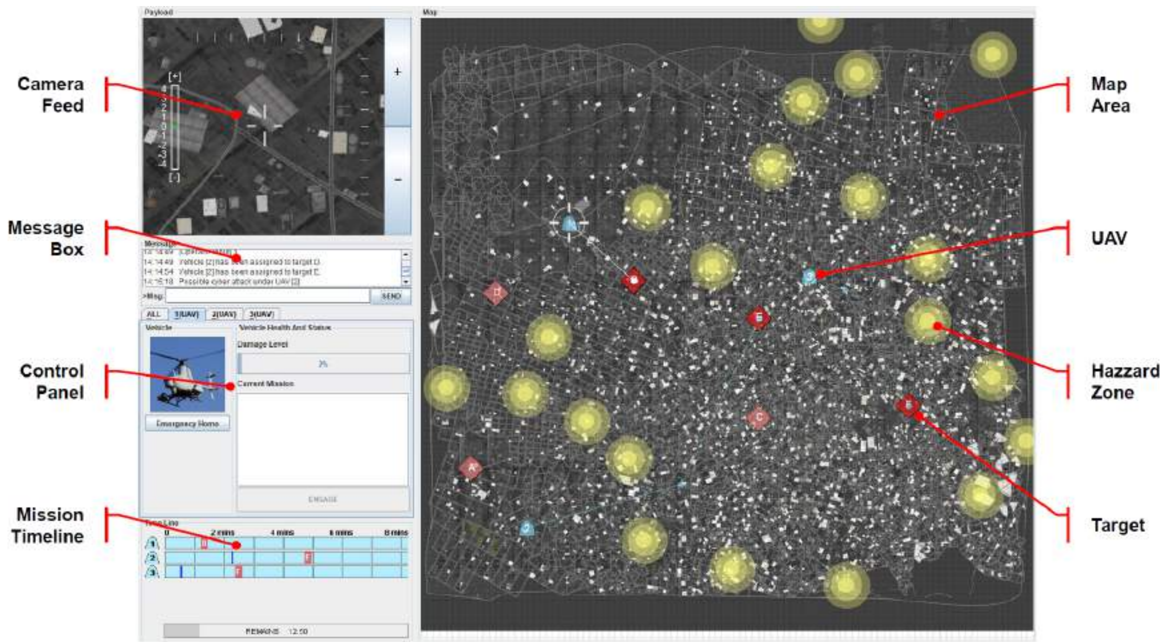


Figure 14: The RESCHU-SA interface

The interface of the RESCHU-SA platform is shown in Figure 14. Five main components are featured in the interface, including the payload camera view, message box, control panel, timeline and map area. Specifically, the camera view displays the video stream from the payload camera of the selected UAV. The primary purpose of this view is to conduct imagery analysis tasks and to determine the actual location of UAVs for detecting potential hackings. And the map displays the area of surveillance with real-time locations of all UAVs, hazard areas and targets.

In the Navy experiment, 36 participants took part with 22 males and 14 females. Age ranged from 19 to 34 years with an average of 25.2 and a standard deviation of 3.83 years. Among the participants, 18 had little video game experience, 6 participants had monthly gaming experience, 5 participants played video game several times a week, another 5 participants had weekly gaming experience, and only 2 participants had daily gaming experience. All participants experienced two 20 minute test sessions which generated 72 full data sets. When developing HMM models, one limitation is data so such relatively large data sets are critical to ensure that pattersn are well enough established to be detected.

The primary objectives of operators in RESCHU-SA are to control multiple UAVs to: 1) determine whether UAVs are under GPS spoofing attacks, 2) perform reconnaissance imagery tasks of counting road intersections when UAVs reach assigned targets, and 3) ensure UAVs do not encounter hazard areas. For this experiment, GPS spoofing attack events followed a pre-defined schedule. A hacking notification appeared 10 to 20 seconds after the attack event, simulating an external agency detection of a possible GPS spoofing attack. However, the notification could be a false alarm. In fact, 50% of all notifications were false alarms.

Operators were responsible for ensuring UAVs visited the most targets possible, which required them to ensure the automated path planner was correct in its UAV-target assessment. The RESCHU path planner for the UAVs is intentionally suboptimal, meaning that often the UAV paths were very inefficient, which allowed experimenters to assess how much spare attention participants could devote to optimize the navigation and target assignment.

When UAVs that were not hacked reached a target, the operator engaged in an imagery task of counting the road intersections from the UAVs camera view at a pre-specified zoom level. This task represents the primary purpose of the mission which is information gathering. While engaging in a counting task, the operator was required to enter an answer before the counting task was finished. The imagery counting allowed us to assess performance based on the number of attempted tasks and the task correctness percentage.

Once the operator received notifications from the system that a certain UAV was under possible cyber-attack, the operator could then investigate the UAV by checking the UAV camera view and matching it against the position of the UAV on the map. If the operator decided the UAV was hacked, the operator was expected to override the hacked UAV and send it home.

The results showed 23 out of total 72 experiment sessions (32%) resulted in 100% of successful hack identifications, with another 24 (33%) reaching above 80% successful attack identification. Thus, 65% of total experiment sessions exhibited 80% correct hacking detection or better without having any prior formal training. In terms of incorrect hacking identifications, 12 (17%) participants lost one or more UAVs, meaning that these UAVs were successfully hacked and could not be further controlled.

For hacking events in both test sessions for each participant, 7 out of 15 events were pre-defined as false alarms, which meant the threshold for correct notification of a problem was 47%, approximately a half. Out of the correct notifications, the overall success rate was 78%, and for the false alarms, the success rate was 84%. Thus, operators were slightly better at detecting false alarms than identifying correct notifications. The second important question was what factors affected operators' performance. For the three performance scores of vehicle damage, correct percentage intersection counts, and correct percentage of hacking events, the only variable task load affected was vehicle damage, where people under high workload experienced more damage than those in the low workload condition.

Operator HMM Models

Descriptive human operator models were developed using the Hidden Markov model (HMM) approach to investigate if and what operator strategies emerged given the parallel tasking of the primary information search mission and the need to address hacking events. There were two basic models developed, one that looks at the overall cognitive flow for how operators approached the entire RESCHU mission, including navigating the UAVs, engaging in the intersection search task at each target, and performing potential hacking analyses. Then another HMM was developed that looked at the hacking strategies in detail. These models are explained in the next sections.

Overall Mission HMM

To develop the HMMs, an unsupervised model training approach of the multi-sequence Baum-Welch algorithm was used in model training (Rabiner & Juang, 1986). Then the final number of states was selected using Bayesian information criterion (BIC) (Schwarz, 1978) and the number of rare states (NRS) (Rodríguez-Fernández, Gonzalez-Pardo, & Camacho, 2016) to get models with high model likelihood values with reasonable model structures. The HMM models were trained from 72 different test sessions, using 12 observations of operator interactions with the system, as shown in Table 3.

Table 3: Observable states in RESCHU-SA

Index	1	2	3	4	5	6
Observations	Add waypoint	Move waypoint	Delete waypoint	Move endpoint	Switch target	Engage task
Index	7	8	9	10	11	12
Observations	Select UAV	Acknowledge notification	Ignore notification	Consider UAV hacked	Consider UAV not hacked	Adjust zoom level

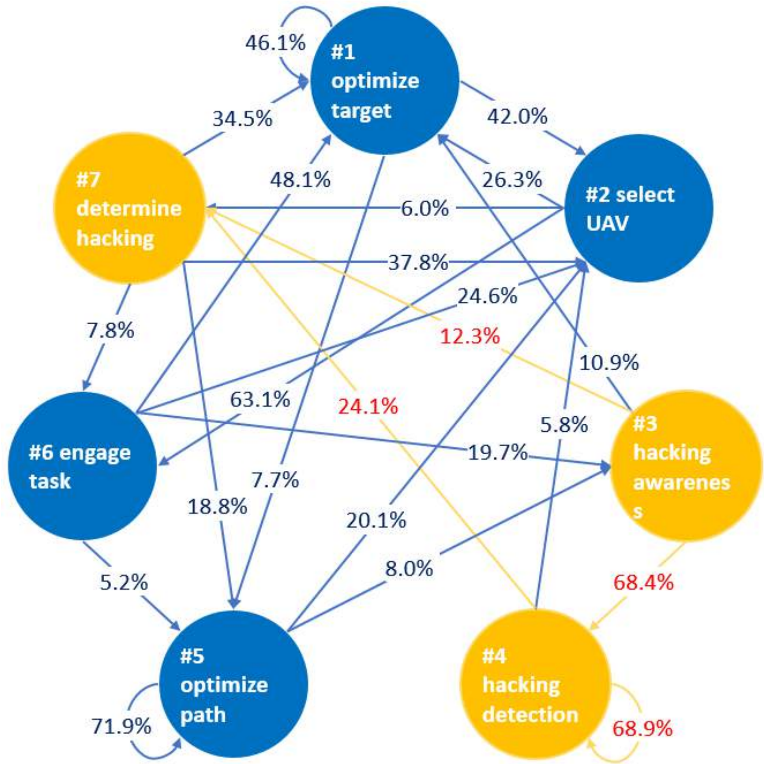


Figure 15: Overall Mission RESCHU-SA Operator Model

The overall mission operator behavior model is a 7-state HMM model (Figure 15). The first hidden state represents a cluster of clicks that represent an operator switching UAV-target assignments, which was an expected behavior since the automated target assignment was intentionally suboptimal.

The second hidden state is interpreted as "select UAV" which is the same as the 7th observable in Table 3. Operators would select this state to indicate that a particular UAV was the point of focus.

The third state is interpreted as "perceive hacking" because only this state had clusters of clicks of "acknowledge hacking" and "ignore hacking", which were selections operators made to indicate they knew they had received a hacking notification.

The fourth state is interpreted as "hacking detection" because it included "add waypoint", "move waypoint" and "adjust zoom level" from Table 3, which usually occurred in hacking detection tasks just after receiving a hacking notification. Recorded videos helped to confirm this interpretation that operators mostly adjusted the video feed zoom level in hacking detections.

The fifth state is interpreted as "optimize path" because it related to UAV navigation clicks, and notably had a high self-transition probability of 72%, which was also confirmed by recorded videos that operators tended to group path optimizations for multiple UAVs together.

The sixth state was interpreted as "engage task" because this was the cluster of clicks where people would go into the targeting window and pan and zoom to count the road intersections.

The seventh state was interpreted as "hacking decision" because it contained the cluster of clicks for "consider UAV hacked" and "consider UAV not hacked", which were operators' final decisions to hacking events.

The various states in Figure 15 are colored blue and yellow to indicate two functional groupings. The hacking detection functional group, which is highlighted in yellow in Figure 15, can be clearly distinguished from the primary UAV management functional group, which is highlighted in blue. In fact, the blue functional management group is very similar to another HMM developed from a completely different experiment using RESCHU (Boussemart et al., 2011). In this experiment, operators just had to manage the paths, target assignments, and search tasks, but did not have any hacking detection tasks.

This is an important finding because it demonstrates that HMMs can be used to potentially model humans in a modular fashion, meaning that when using the same interface, if a new capability it added on, new global models do not need to be built and that it may be sufficient to just build new models of the new functionality. This is explored in detail in the next section.

Hacking Detection HMM

The 3-state hacking sub model in Figure 15 is interesting because at this level of abstraction, it is clear that the hacking cognitive model parallels the typical information process model of perception-cognition-action, in that people perceive a hacking event, investigate the event to determine causality, and then make a decision. However, this broad brush is not very diagnostic, so in order to gain better insight into the hacking sub-model highlighted in Figure 15, only those observable states that were executed from the onset of hacking notification to the submission of a hacking detection judgment (hacked or not hacked) were used to generate another HMM in the same way the first one was determined. This model (Figure 16) shows more detailed cognitive processes for how people approached the hacking detection problem.

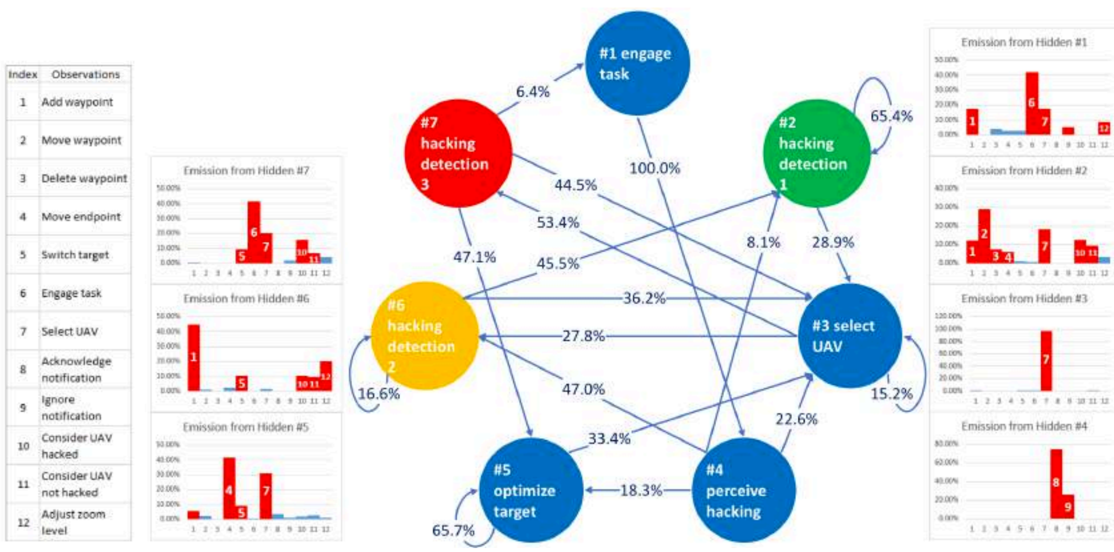


Figure 16: Detailed hacking detection model with click probabilities per hidden state

All the blue states inside Figure 16 are similar to those in Figure 15 and represent the time during a hacking event that people divided their attention to attend to the other UAVs not under attack. Figure 16 also demonstrates that there were 3 different prominent strategies in hacking detection determination. In the histograms near each state, the individual clicks that make up that state can be seen.

For hacking detection strategy #1 (hidden state 2), given the concentration of operators on adding, moving and sometimes deleting waypoints, these people were physically moving the UAV to a new location to cross check the motion of the UAV on the map against what they were seeing in the camera. We call this strategy "motion verification" since people were primarily focusing on the vehicle movement as the cue to verification.

In strategy #2 (hidden state 6), people tended to add waypoint and then zoom their camera view to see the ground in more details. This is a more organized strategy as seen in the high click count, and people were much more focused on "landmark verification", given their use of the camera to zoom in.

For strategy #3 (hidden state 7), people did not really change the physical location of the vehicle that much, and preferred to attempt to determine if the vehicle was hacked by focusing on matching the target on the map to what they see in the camera view. We call this strategy "stay the course verification".

Of the three strategies, we can see that the "motion verification" strategy was very inefficient and took the longest since there is a 65.4% chance that people would keep transitioning back into this state repeatedly (this is called a self-transition). We know from the experiment statistics that the time consumption in hacking detection was negatively correlated with the hacking detection success rate (Pearson=-0.375, p=0.001). In contrast, "landmark verification" operators experienced significantly fewer self-transitions and "stay the course verification" operators generally committed early to this strategy and with few changes to the environment.

In order to determine how well these hacking strategies helped individuals perform, we examined how the best performers' strategies aligned with the three overarching strategies. There were two expert performers who both achieved perfect hacking detection and also performed well on the intersection tasks. Their individual hacking detection strategies, as compared to the overall HMM model in Figure 16 are detailed in Figure 17.

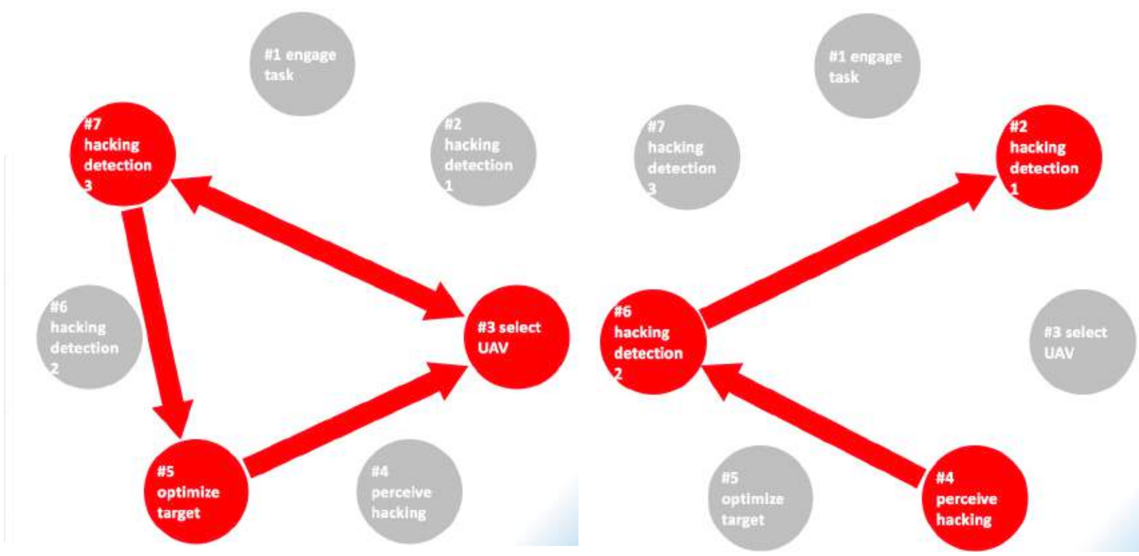


Figure 17: Expert Strategies for Expert A (left) and Expert B (right)

In Figure 17, Expert A adopted strategy 3, which was to optimize target-UAV assignment and then use the arrival of a UAV at a target as confirmation of the geo-spatial position. Expert A was also able to multi-task in that he continued to balance UAV/target assignment for other non-hacked UAVs while waiting for a potentially hacked UAV to get to its target. Expert A was consistently ~0.5 standard deviations (SD) faster in hacking detections, but was only slightly higher that the average percentage for number of targets successfully hacked.

Expert B adopted a hybrid strategy which included redirecting the UAV both to check its motion path but also the landmarks underneath specific waypoints. Expert B was very focused on the hacking task and spent very little time on managing the other UAVs. This resulted in hacking performance even better than Expert A, in that Expert B was ~1 SD faster than average operators in terms of hacking detection. However, Expert B scored only average on the successful number of prosecuted targets.

One common thread between these two experts was their video game experience. Both of these operators self-identified as daily gamers, and these were the only two participants in the entire subject pool at this highest level of video game experience. They were also the only participants with 100% successful hacking detection. However, while they were superior to others in hacking detection, they were only average to slightly above average in the other primary mission of prosecuting as many targets as possible.

The strategy analysis for this effort is still underway but these preliminary results are very encouraging. These results showed that strategies can be mapped, and that inefficient as well as effective strategies can be identified. This could be very important for identifying whether trainees adapt effective/efficient strategies, and how the best way to correct any problems. Moreover, such models could provide suggestions for engineers for how to use automation to assist human operator, as the models indicate where people get bogged down, and thus the models could point to steps where time spent in hacking detection manipulations, especially in self-transitions, could be reduced. We are currently investigating the best ways to do this.

Summary

The point of the HMM modelling effort was to determine if and how such models could be used to elucidate training behaviors. Using novices in a representative multiple UAV control simulation where the UAVs could be hacked, we could see that added functionality resulted in added states to an already known cognitive model. Furthermore, we could see that the participants clustered into 3 strategies, with the best performers adopting hybrid strategies.

These results strongly suggest that more work is needed to extend this method in real Army applications because not only could it identify how people are developing strategies, ether in training or in actual operations, but also where possible inefficiencies lie. Moreover, while the current models are descriptive, because they are based on user clicks, they could be turned into predictive models and could actual be used in error prediction. So, in this case, the computer could predict when an operator was executing a strategy that would likely be suboptimal and then notify a supervisor that intervention may be needed.

Overall Project Summary

This project, which broadly focused on developing better diagnostic approaches to identify training deficiencies as well as the impact of inserting futuristic forms of autonomy into a training program. To this end, the following milestones were achieved:

- A midterm report was published per the statement of work in February of 2016 (Hutchins, Clamann, Furth, & Cummings, 2016)
- A provisional patent was filed for the Autonomous System Training Model (ASTM) that incorporates training requirements, the influence of increasing autonomy, both training and operational errors, and trainee attrition (Appendix B)
- The ASTM was applied to Army UAS systems (Grey Eagles and Shadows). Validation results showed the model was more accurate for Phase 1 than Phase 2, although more data is needed from multiple years to further elucidate potential error sources in the model. The model also clearly suggested that overall training time was most affected by variability in Module C training, which focuses on UAS Ground School. Thus, one tangible result from this model is that more work is needed to improve this module both in terms of variability of training time and content.
    - A journal paper based on this is in development.
    - A poster was presented at an undergraduate research conference at Duke in 2016 highlighting this work (Appendix C)
- An experiment was designed, including all the supporting software and training material, to investigate the relationship between skill-, rule-, and knowledge-based reasoning in a multiple UAS control environment.
- Hidden Markov Models were shown to be an effective way to represent operator strategies with clear indications for potential training applications. Such a model could be used in a diagnostic manner (i.e., identifying problems in training or identify areas where autonomy is needed) or in a predictive manner (identifying a poor strategy in real time which could be addressed before resulting in a critical error.)
    - A journal paper based on this is in development.
- A Fitts's Law experimental testbed was created and is being made available for any interested researchers on the HAL website.
- RESCHU-SA is a freely available experimental testbed for any interested DoD-related researchers.
- This research led to additional funding through ONR to further explore potential dependencies on skills and rules in the development of knowledge.
- This research also partially supported the publication of this high impact paper with the Royal Institute of International Affairs (aka, Chatham House, the 2nd most influential think tank in the world).
    - https://www.chathamhouse.org/sites/files/chathamhouse/publications/research/2017-01-26-artificial-intelligence-future-warfare-cummings-final.pdf

Acknowledgments

Students that participated in this effort were Andrew Hutchins, Bhargav Chauhan, Joshua Furth, and Ted Zhu, research scientists were Michael Clamann and Alex Stimpson, and Lixiao Huang was a post-doctoral associate who assisted on this effort. We appreciate all the help from Mark Farrar, Director of Training 2-13th AVN REGT Fort Huachuca, AZ, CW4 James Harris from Ft. Bragg, TCM-UAS training developers and training leads from Ft. Rucker, AL, Colonel "Dutch" Holland and the USMC Shadow operators at VMU-1 MCAS Yuma, AZ, and COL Jasper Jeffers, Commanding Officer of the 1st Stryker Brigade, 2nd Infantry Division. All were indispensable to the success of this effort.

Appendix A: Original Statement of Work

Task 1: Adapt the joint human-autonomous system metric class framework described in the previous section to explicitly consider the evaluation of training, as well as the need for different levels of reasoning.

The current framework only considers metrics for operation and not evaluation. Moreover, it does not address the need for different levels of cognitive reasoning such as skill, rule, and knowledge based reasoning. By adapting the framework, a better diagnostic approach can be taken that can identify training concerns as they apply to the different layers of reasoning and how increased autonomy may be either hindering or promoting various training objectives.

Task 2a: Apply the adapted framework to a generalized UAV and UGV Army application to illustrate how the framework can and should be utilized for training evaluation.

Once the framework is adapted, developing use cases that illustrate how such a framework can augment training evaluation is critical for dissemination. Moreover, this task will include determining how such a framework can inform autonomy design through the training evaluation lens, which will include identifying what functions could be allocated to the automation, human, or both.

This task will also include stakeholder engagement, i.e., Army training professionals will be consulted for their expertise for specific systems to gain a deeper insight into those metrics that are considered the most important.

> Task 2b: Determine if and how machine learning algorithms could be used to gain additional insights from the training evaluation data.
>
> Recent work in training evaluation research in computer-based training settings (Stimpson & Cummings, 2014; Stimpson, Buinhas, Bezek, Boussemart, & Cummings, 2012) has determined that it is possible to leverage powerful machine learning algorithms (i.e., big data), to detect patterns of trainee behavior that would otherwise gone undetected. One sub task in this effort will be to look at the data collected in a representative setting under Task 2 and determine whether machine learning algorithms could provide insight into trainee behavior and the quality of the training program.

Option Tasks

Task 3a: Using the metric class framework and the results from Task 2, develop a feedback model of an Army UAV or UGV application that addresses how increases in autonomy across various functions could potentially affect training evaluation.

Once the generalized approach to using the joint human-autonomous system evaluation framework is established and vetted with the subject matter experts, we will determine how to take the structure and the results from metric class framework, and develop a feedback model to illustrate how increasing autonomy can affect training outcomes and how this relationship can and should be evaluated. For example, new path planning algorithms could be inserted into either UAVs or UGVs which would change the level of reasoning a human would apply for navigation tasks. Thus, the evaluation approach would commensurately change.

Task 3b: Test the validity of the feedback model in a simulated task environment.

In this task, we will work with Army training personnel to determine how this approach helps them in evaluation for both individuals as well as overall training program development. We also can continue iterative improvements on the assessment strategy and utility of the framework.

Task 4 was not funded**

Task 4a: Using the metric class framework and the results from Task 3, refine the feedback model to consider how changing the user interface as well as an increase in autonomy across a specific function could affect training evaluation.

Task 4b: Test the validity of the feedback model in a simulated task environment.

This task will also involve working with Army personnel to determine model changes as a result of changes in the interface in the presence of increasing autonomy.

Deliverables: A mid term and a final report that describe the modified metric taxonomy, the application of it to representative Army unmanned platforms, the analysis of the applicability of machine learning algorithms, and the feedback models should the option be exercised.

Appendix B: AUAS ASTM Model

Figure B.1 shows a detailed sample from the Phase 1 AUAS ASTM, which was designed using Vensim® SD modelling software. This sample from the larger AUAS ASTM has three of the five sections shown (Training Modules, Module Training Time, and Module Training Tasks). These three sections make up the core of the ASTM.

Initially, the total number of tasks that are taught in each training module are determined from the training syllabus. Then, the tasks are classified into SBT, RBT, or KBT in the Module Training Task section, using the method illustrated in Table 1 and Figure 2. After the tasks are split into SBT, RBT, and KBT in the ASTM, they are converted into training times. Each set of SBTs, RBTs, and KBTs has their total training time per module. This is calculated in the Module Training Time section of the ASTM (Figure B.1) by multiplying the number of SBT, RBT, or KBT by the average time to train a SBT, RBT, or KBT. The resulting value is the total time to train all of the tasks in each module in terms of SBT, RBT, and KBT training time.

Finally, the time to train the SBTs, RBTs, and KBTs are summed to give the total module training time. This total module training time parameter, *Phase 1 Module (A, B, or C) Required Weeks of Training* (Figure B.1), should match the prescribed time from the training syllabus, assuming the equations and average SBT, RBT, and KBT training time per module are accurate. Recall that there are three modules in FY14 Phase 1 training, and, thus, this procedure will occur three times for each module.

The *Phase 1 Module (A, B, or C) Required Weeks of Training* parameter is used to control the rate functions of trainee groups moving from one module to the next. Since there are three modules in FY14 Phase 1, there are three *Phase 1 Module Required Weeks of Training* parameters, one for each module. When batches of trainee groups arrive to a module (stock) they will stay in that module for the duration of that module's required weeks of training. Once that time has expired, the trainee groups move into the next module and the process begins again. This cycle continues until the trainees reach the Trainee Outputs section (Figure 4), where they will either graduate, be recycled, discharged, or retrain/reclassed depending on their performance.

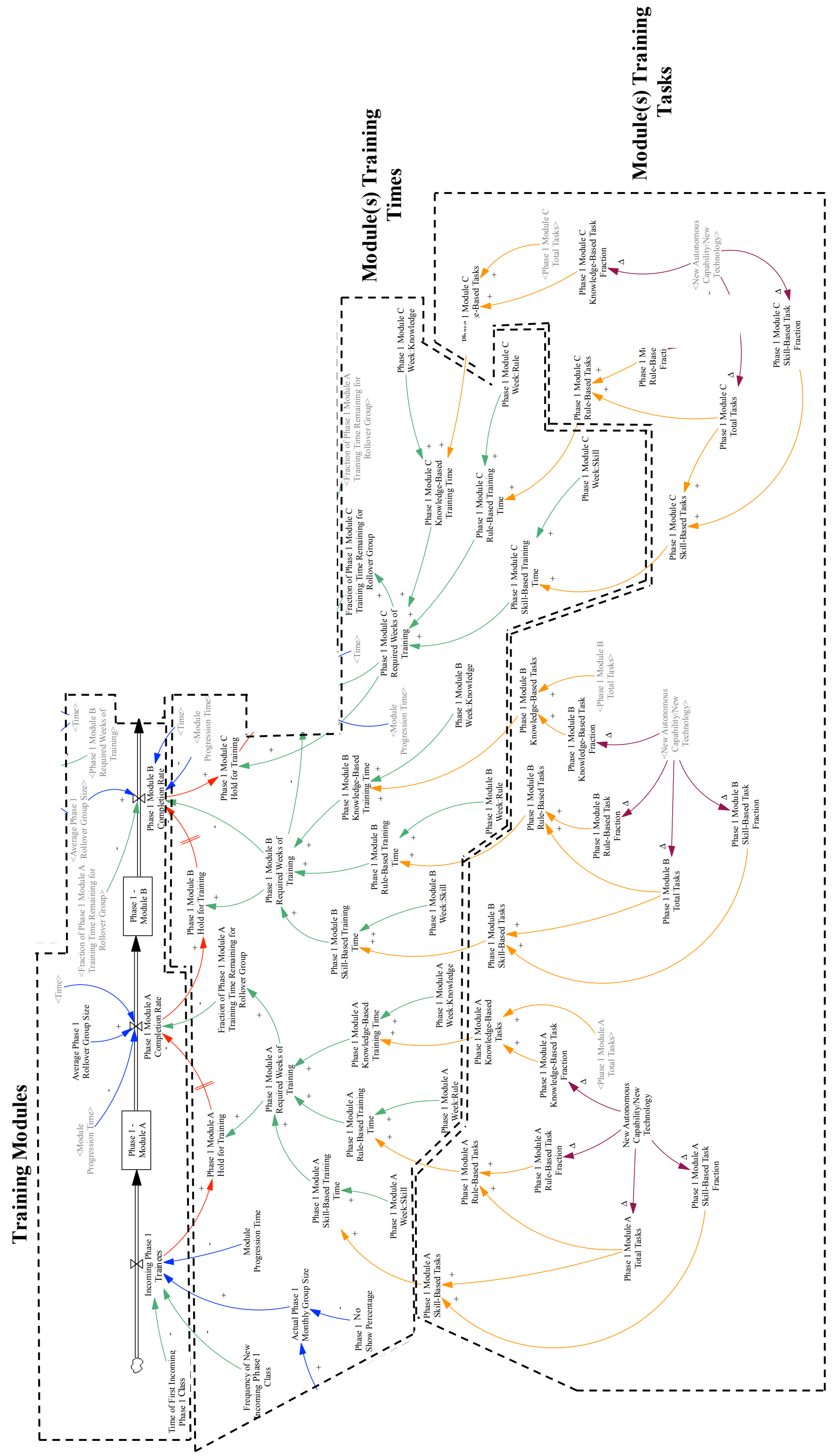
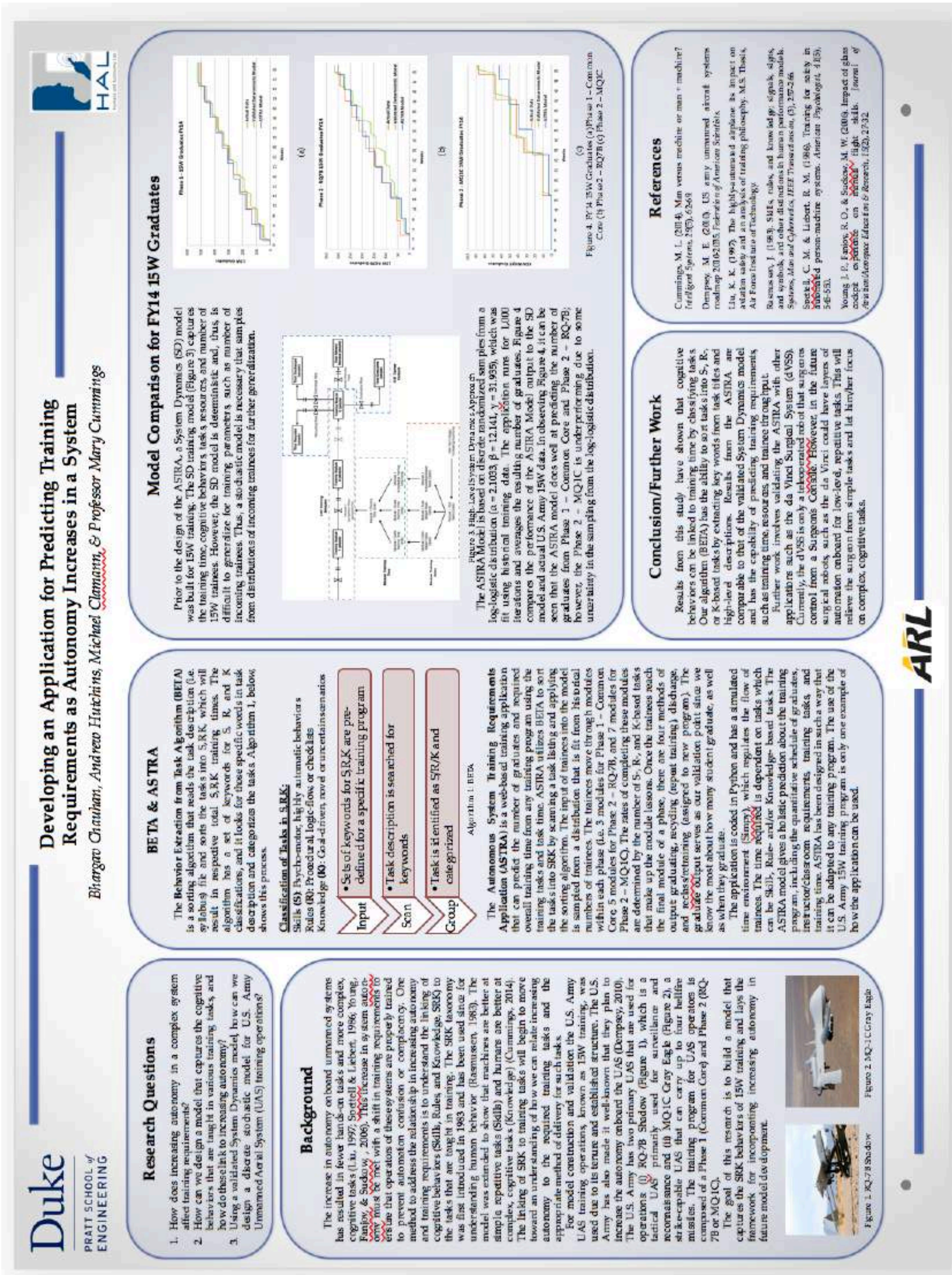


Figure B.1. Sample of AUAS ASTM for Phase 1 in FY14. Shown are the Module Training Tasks and Module Training Times sections that control the rate functions between modules in the Training Modules section

Appendix C: Developing an Application for Predicting Training Requirements as Autonomy Increases in a System



Appendix D: Provisional Patent

The U.S. Army Unmanned Aircraft Systems Roadmap 2010-2035 outlines a plan to increase autonomy on board Unmanned Aircraft Systems (UAS) in the near future. Benefits from increasing autonomy onboard UAS include reduced workload and increased safety for Soldiers and decreased operational and training costs. However, to date, there exists no methodology for predicting how increasing autonomy onboard UAS will be reflected in the operator training process or identifying what operator abilities will be required once the advanced technologies have been implemented. In response, we have proposed and are developing a modeling tool that links required core cognitive behaviors to the time needed for training these, as well as how autonomous technologies influence such training.

Modifications to training programs, such as what will be needed with increasingly autonomous aerial and ground vehicles, should account for differences in how functions are allocated to the operator and/or the automation. We propose that these functions can be expressed by both human and machines as increasingly complex reasoning behaviors, including *Skills, Rules,* and *Knowledge* (SRK). Skill-based behaviors are sensory-motor actions that are highly automatic, typically acquired after some period of training. Army UASs already automate many skill-based behaviors, e.g., the Shadow's autopilot keeps it in balanced flight. Rule-based behaviors are actions guided by subroutines, stored rules, or documented procedures. Knowledge-based behaviors incorporate the formulation and selection of plans for an explicit goal based on individual mental models.

To explicitly link training requirements (in terms of completion time and tasks) to improvements in autonomous system capabilities, we have developed the Autonomous Systems Training Model (ASTM) that incorporates typical training metrics as well as qualitative and quantitative training variables. ASTM was initially conceptualized using a *System Dynamics* framework, which allowed us to include both the qualitative (i.e., cognitive reasoning) and quantitative variables (i.e., training capacities) that influence the training process.

The current iteration of ASTM models the RQ-7B Shadow training process. The quantitative baseline for the model was developed using personnel and curriculum requirements from the 2-13th Aviation Regiment at Fort Huachuca, Arizona. The qualitative requirements of the RQ-7B training curriculum were adapted from training tasks described in the Unmanned Aircraft System Commander's Guide and Aircrew Training Manual (TC 3.04-61) and the RQ-7B Program of Instruction (POI) for FY10 and FY14. In the context of the manual, a training *task* is a defined and measurable activity performed in a simulator or live UAS platform during training, such as performing simulated aerial reconnaissance.

We divided each of the RQ-7B 23 required training tasks described in the manual into subtasks and categorized them as skill, rule, or knowledge-based tasks. By linking quantitative training time with qualitative behaviors in the model, we can make predictions as to how shifts in tasking from human manual control to the domain of automation will affect required skill- and rule- to knowledge-based behaviors, and ultimately training time. For example, when automating the landing task, several skill and rule based tasks shift to the automation, but knowledge-based tasks still remain for the human. Such a shift places the operator in a more supervisory role, which requires changes to the training curriculum. A high-level view of the ASTM appears in Figure 1.

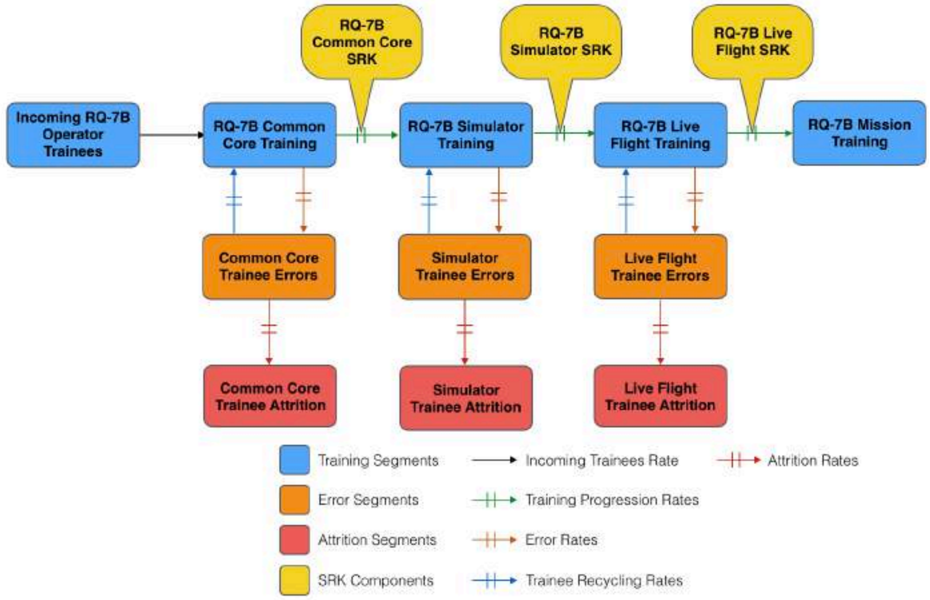


Figure 1. ASTM Overview

There are four main components to the ASTM in Figure 1: Training Segments, Error Segments, Attrition Segments, and SRK Components. The Training Segments model Soldiers progressing through Common Core (i.e., classroom), Simulator, Live Flight, and Equipment training. The Error Segments model Soldiers held back to correct errors made in a Training Segment. The Attrition Segments contain Soldiers who leave the training program due to unresolved errors or personal reasons. Finally, the SRK Components represent specific task requirements that each Soldier must complete in each Training Segment. Because SRKs are agent-agnostic, i.e., either humans or autonomy must accomplish them, they link increasing autonomy to improving training efficiency.

In addition to training efficiency in terms of time for a single platform, ASTM can be expanded to represent operators controlling multiple UASs simultaneously, or could be modified to address training programs of different types of autonomous vehicles. When fully developed and validated, ASTM could be used as a predictive model to investigate the impact of future technologies on training as well as how the insertion of different technologies and associated tasks in the training pipeline will affect overall throughputs.

Figure 2 depicts the kind of predictive information that ASTM could provide in terms of how implementations of new technologies could affect training time for UAS operators. The horizontal axis depicts a timeline of likely technological advances for UASs. The vertical axis indicates total training time as these future functionalities materialize. The blue dotted line represents progress assuming a linear function that is typical of UAS development today while the red dotted line represents a best-case scenario with substantial improvements in artificial intelligence that relieve the human from the bulk of supervisory tasks.

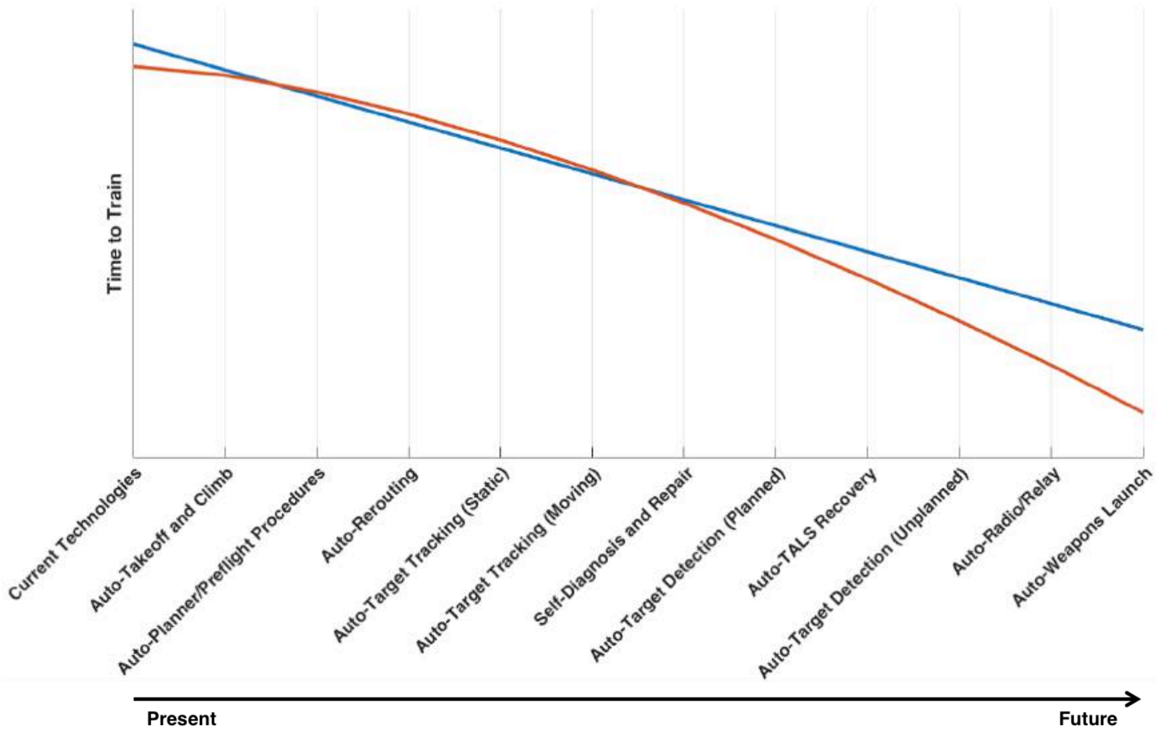


Figure 2. Training Time Projection with Increasing Autonomy. The blue line represents progress assuming evolutionary development and the red line represents progress assuming revolutionary advances in artificial intelligence.

The ASTM modeling software we are developing will provide the U.S. Army Training and Doctrine Command (TRADOC) with the ability to assess how technology advances will affect training requirements and predict the effects of new training requirements on overall training efficiency. Moreover, such a modeling tool will also have diagnostic and design elements such that current training programs can be assessed to ensure they are addressing the requisite skills, rules, and knowledge needed, as well as indicating whether different training program designs are sufficient.

The next step in ASTM development to extend this model into an agent-based simulation such that this model can be turned into usable software so Army decision makers can explore possible future training architectures. In addition, such an extension will also allow us the ability to make ASTM more generalizable across different training settings for various autonomous technologies. Given the utility of this generalizable structure, a technology disclosure agreement has been filed with Duke University to allow us to move forward with filing for a patent or licensing opportunities.

Appendix E: Pre and Post Experiment Surveys

ARL - Demographic Survey

Please provide your information in the spaces below.

Subject ID: ____________________

Age: ____________________

Gender:
❍ Male
❍ Female

What is your visual acuity?
❍ Normal Vision (20/20)
❍ Corrected Vision (20/20)
❍ Other (please specify)____________________

Do you have any type of color blindness?
❍ No
❍ Not Sure
❍ Yes (please specify what type) ____________________

Current occupation:
❍ Student
❍ Army personnel
❍ Other (please specify) ____________________

If you are a student, what program are you currently enrolled? (If non-student, skip this question)
❍ Undergraduate
❍ Masters
❍ PhD
❍ Other (please specify) ____________________

If you are a student, when is your expected graduation year? (If non-student, skip this question)
____

1. How often do you play computer games normally?
   - ❍ Rarely
   - ❍ A few times a month
   - ❍ Once a week
   - ❍ A few times a week
   - ❍ Daily

2. Types of video games played (check all that apply):
   - ❑ First person shooter
   - ❑ Sports
   - ❑ Action/Adventure
   - ❑ Third person shooter
   - ❑ Educational
   - ❑ Survival horror
   - ❑ Puzzle
   - ❑ Role playing
   - ❑ Real time strategy
   - ❑ Beat em ups
   - ❑ Other unknown type (please explain) ____________________
   - ❑ I have never played any video games

3. How confident are you using computer programs in general?

	1	2	3	4	5	
Not at all confident	○	○	○	○	○	Very confident

4. How do you like Unmanned Aerial Vehicles (Drones) in general?

	1	2	3	4	5	
Strongly dislike	○	○	○	○	○	Strongly like

5. What is your experience examining aerial imagery?

	1	2	3	4	5	
None	○	○	○	○	○	Extensive

Questions for Session 1

Subject ID: ____________________

1. How confident were you about visual inspection tasks during session 1?

	1	2	3	4	5	
Not at all confident	○	○	○	○	○	Very confident

2. How do you feel you performed?

	1	2	3	4	5	
Poor	○	○	○	○	○	Excellent

3. Do you feel some tasks are more difficult than others?
   - ❍ Yes
   - ❍ No

4. Explain why you feel this way in the question above.

5. How hard do you think you worked during this session?

	1	2	3	4	5	
Minimal effort	○	○	○	○	○	Extremely hard

Questions for Session 2

Subject ID: ____________________

1. How confident were you about visual inspection tasks during session 2?

	1	2	3	4	5	
Not at all confident	○	○	○	○	○	Very confident

2. How do you feel you performed?

	1	2	3	4	5	
Poor	○	○	○	○	○	Excellent

3. Do you feel some tasks are more difficult than others?
   - ❍ Yes
   - ❍ No

4. Explain why you feel this way in the question above.

5. How hard do you think you worked during this session?

	1	2	3	4	5	
Minimal effort	○	○	○	○	○	Extremely hard

Post Experiment Survey

Subject ID: ____________________

1. Do you think the training was sufficient to prepare you for the missions?
   - ❍ Yes
   - ❍ No

2. How well do you feel you understand how to operate the simulation?

	1	2	3	4	5	
Very poorly	○	○	○	○	○	Very well

3. Comments about the training:

Appendix F: Experiment Checklists

ATR Experiment – Procedure Checklist *v2.0.5*

Participant information		
ID:	Group: SR Without ATR	Date:
Name:		

CHECKLIST		
	Setup	
	Consent Form	
	Ask if subject wants a copy of the form	
	Demographic Survey	
	Camtasia turned ON	
	Change mouse settings	
	Fitts's Test - standard mouse	
	Save Data,	
	Ask if subject has used the Unfamiliar Mouse Input Device before	
	Change mouse settings	
	Fitts's Test – Unfamiliar Mouse Input Device – 450 clicks	
	Save Data	
	RESCHU Presentation – WITHOUT ATR	
	RESCHU Practice session – WITHOUT ATR	
	Save Data	
	Test Session 1 – WITHOUT ATR	
	Save Data	
	Post session 1 survey	
	Test Session 2 – WITHOUT ATR	
	Save Data	
	Post session 2 survey	
	Fitts's Test – Unfamiliar Mouse	
	Save Data	
	Turn Camtasia OFF	
	Reset Mouse Settings to Default	
	Post experiment survey	
	Debriefing	
	IRB Compensation form	

ATR Experiment – Procedure Checklist *v2.0.5*

Participant information		
ID:	Group: SR With ATR	Date:
Name:		

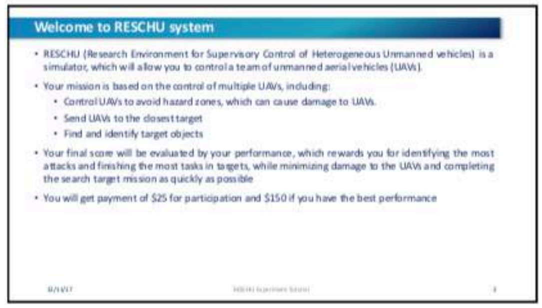
CHECKLIST		
	Setup	
	Consent Form	
	Ask if subject wants a copy of the form	
	Demographic Survey	
	Camtasia turned ON	
	Change mouse settings	
	Fitts's Test - standard mouse	
	Save Data,	
	Ask if subject has used the Unfamiliar Mouse Input Device before	
	Change mouse settings	
	Fitts's Test – Unfamiliar Mouse Input Device – 450 clicks	
	Save Data	
	RESCHU Presentation – WITH ATR	
	RESCHU Practice session – WITH ATR	
	Save Data	
	Test Session 1 – WITH ATR	
	Save Data	
	Post session 1 survey	
	Test Session 2 – WITH ATR	
	Save Data	
	Post session 2 survey	
	Fitts's Test – Unfamiliar Mouse	
	Save Data	
	Turn Camtasia OFF	
	Reset Mouse Settings to Default	
	Post experiment survey	
	Debriefing	
	IRB Compensation form	

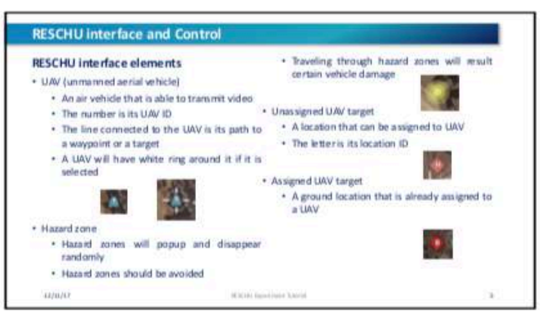
ATR Experiment – Procedure Checklist *v2.0.5*

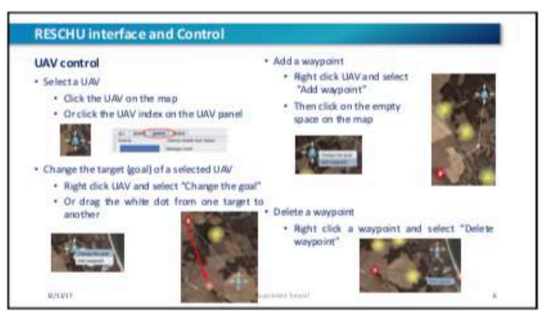
Participant information		
ID:	Group: R-only With ATR	Date:
Name:		

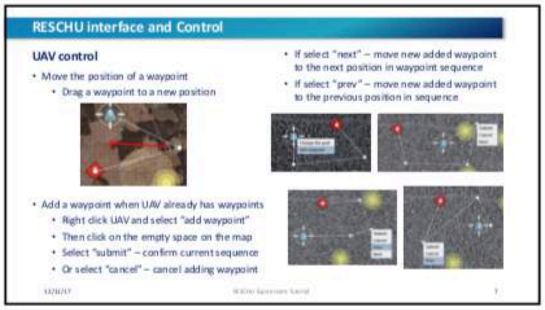
CHECKLIST		
	Setup	
	Consent Form	
	Ask if subject wants a copy of the form	
	Demographic Survey	
	Camtasia turned ON	
	Change mouse settings	
	Fitts's Test - standard mouse	
	Save Data,	
	Ask if subject has used the Unfamiliar Mouse Input Device before	
	Change mouse settings	
	Fitts's Test – Unfamiliar Mouse Input Device – ~~450 clicks~~, 30 Clicks	
	Save Data	
	RESCHU Presentation – WITH ATR	
	RESCHU Practice session – WITH ATR	
	Save Data	
	Test Session 1 – WITH ATR	
	Save Data	
	Post session 1 survey	
	Test Session 2 – WITH ATR	
	Save Data	
	Post session 2 survey	
	Fitts's Test – Unfamiliar Mouse	
	Save Data	
	Turn Camtasia OFF	
	Reset Mouse Settings to Default	
	Post experiment survey	
	Debriefing	
	IRB Compensation form	

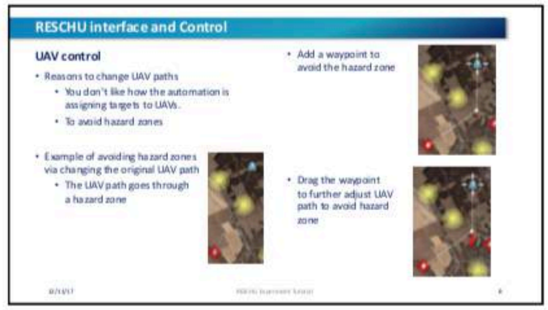
Appendix G: Tutorial Slides (With and Without ATR)



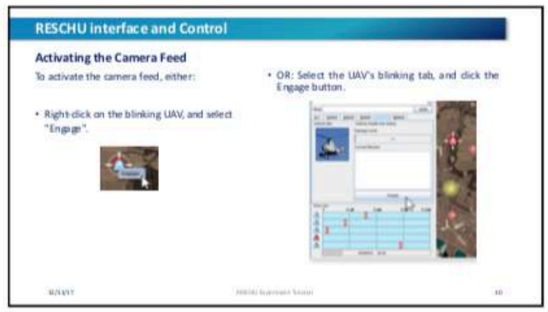





RESCHU interface and Control
UAV control
• Move the position of a waypoint
• Drag a waypoint to a new position
• Add a waypoint when UAV already has waypoints
• Right click UAV and select "add waypoint"
• Then click on the empty space on the map
• Select "submit" – confirm current sequence
• Or select "cancel" – cancel adding waypoint
• If select "next" – move new added waypoint to the next position in waypoint sequence
• If select "prev" – move new added waypoint to the previous position in sequence


RESCHU interface and Control
UAV control
• Reasons to change UAV paths
• You don't like how the automation is assigning targets to UAVs.
• To avoid hazard zones
• Example of avoiding hazard zones via changing the original UAV path
• The UAV path goes through a hazard zone
• Add a waypoint to avoid the hazard zone
• Drag the waypoint to further adjust UAV path to avoid hazard zone


RESCHU interface and Control
Activating the Camera Feed
When a UAV reaches its target...
• The UAV will blink red and blue.
• The UAV's tab in the control panel will flash blue.


RESCHU interface and Control
Activating the Camera Feed
To activate the camera feed, either:
• Right-click on the blinking UAV, and select "Engage".
• OR: Select the UAV's blinking tab, and click the Engage button.


RESCHU interface and Control
Activating the Camera Feed: ATR
• The camera will take a moment to load
CAMERA LOADING...
3... 2... 1...
• Then, it will show the video feed of the drone.
Current Mission
• Your goal is to find the target object indicated by the current mission text.


RESCHU interface and Control
Activating the Camera Feed: ATR
• ATR stands for Automated Target Recognition, and is a new technology that enable drones to detect and fly directly over their target object.
• When ATR works, the target object, as indicated in the current mission text, will be within the viewing window after the camera initializes.
• When ATR fails, the target object will not be in view when the camera initializes. You will need to use the panning and zoom controls to position the video feed over the target object.
Current Mission

RESCHU interface and Control

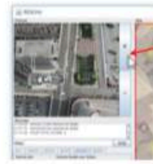
Activating the Camera Feed

- Pan around the camera feed by clicking in the left, right, top, and bottom areas of the camera feed window.

RESCHU interface and Control

Activating the Camera Feed

- Zoom the camera in and out by clicking on the blue + and – buttons found to the right of the video feed.



RESCHU interface and Control

Submitting your target

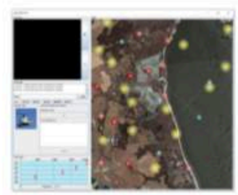
To complete the mission, you will need to submit the position of the target object.

- Make sure the target object is positioned in the viewing window, using the panning and zooming controls.
- Right-click on the target object, and choose "Submit"

RESCHU interface and Control

Returning to the Map

After you have submitted the correct target position, the drone's mission will automatically end and you will be returned control of the main map and the other drones.



RESCHU interface and Control

Finish experiment

- RESCHU system will automatically terminate when time is up
- You will see your final score with your performance statistics immediately after experiment

Conclusion and question

- Keep controlling UAVs to minimize distance between UAV and targets while avoid hazard zones until experiment time ends
- If you have any question, please check with the experimenter before you start
- You will have a chance to practice 10 minutes before real experiment

RESCHU Experiment Tutorial

Humans and Autonomy Lab
Duke University

Welcome to RESCHU system

- RESCHU (Research Environment for Supervisory Control of Heterogeneous Unmanned vehicles) is a simulator, which will allow you to control a team of unmanned aerial vehicles (UAVs).
- Your mission is based on the control of multiple UAVs, including:
  - Control UAVs to avoid hazard zones, which can cause damage to UAVs.
  - Send UAVs to the closest target
  - Find and identify target objects
- Your final score will be evaluated by your performance, which rewards you for identifying the most attacks and finishing the most tasks in targets, while minimizing damage to the UAVs and completing the search target mission as quickly as possible
- You will get payment of $25 for participation and $150 if you have the best performance

RESCHU interface and Control

RESCHU system initial window

- The following shows the start button for the main game

Figure 1. The start window

RESCHU interface and Control

RESCHU system main interface

- The main interface includes:
  - Map – contains the locations of vehicles, hazard zones and targets
  - Camera – display video stream from selected vehicle
  - Message box – displays events and collect operator inputs
  - Control panel – provides vehicles health and mission information
  - Timeline – shows the estimated time of arriving waypoints and target

Figure 2. The main interface

RESCHU interface and Control

RESCHU interface elements

- UAV (unmanned aerial vehicle)
  - An air vehicle that is able to transmit video
  - The number is its UAV ID
  - The line connected to the UAV is its path to a waypoint or a target
  - A UAV will have white ring around it if it is selected

- Hazard zone
  - Hazard zones will popup and disappear randomly
  - Hazard zones should be avoided
  - Traveling through hazard zones will result certain vehicle damage

- Unassigned UAV target
  - A location that can be assigned to UAV
  - The letter is its location ID

- Assigned UAV target
  - A ground location that is already assigned to a UAV

RESCHU interface and Control

UAV control

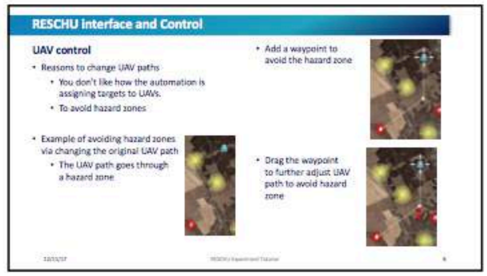
- Select a UAV
  - Click the UAV on the map
  - Or click the UAV index on the UAV panel

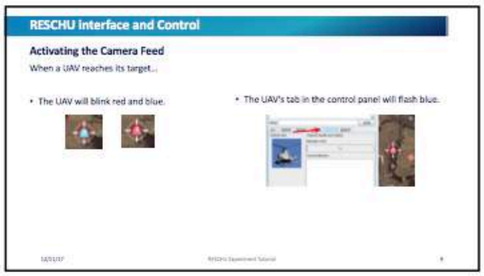
- Change the target (goal) of a selected UAV
  - Right click UAV and select "Change the goal"
  - Or drag the white dot from one target to another

- Add a waypoint
  - Right click UAV and select "Add waypoint"
  - Then click on the empty space on the map

- Delete a waypoint
  - Right click a waypoint and select "Delete waypoint"


RESCHU interface and Control
UAV control
• Move the position of a waypoint
• Drag a waypoint to a new position
• If select "next" – move new added waypoint to the next position in waypoint sequence
• If select "prev" – move new added waypoint to the previous position in sequence
• Add a waypoint when UAV already has waypoints
• Right click UAV and select "add waypoint"
• Then click on the empty space on the map
• Select "submit" – confirm current sequence
• Or select "cancel" – cancel adding waypoint
7


RESCHU interface and Control
UAV control
• Reasons to change UAV paths
• You don't like how the automation is assigning targets to UAVs.
• To avoid hazard zones
• Example of avoiding hazard zones via changing the original UAV path
• The UAV path goes through a hazard zone
• Add a waypoint to avoid the hazard zone
• Drag the waypoint to further adjust UAV path to avoid hazard zone
8


RESCHU interface and Control
Activating the Camera Feed
When a UAV reaches its target...
• The UAV will blink red and blue.
• The UAV's tab in the control panel will flash blue.
9


RESCHU interface and Control
Activating the Camera Feed
To activate the camera feed, either:
• Right-click on the blinking UAV, and select "Engage".
• OR: Select the UAV's blinking tab, and click the Engage button.
10


RESCHU interface and Control
Activating the Camera Feed
• The camera will take a moment to load
• Then, it will show the video feed of the drone.
Current Mission
• Your goal is to find the target object indicated by the current mission text.
11


RESCHU interface and Control
Activating the Camera Feed
• Pan around the camera feed by clicking in the left, right, top, and bottom areas of the camera feed window.
Click Here to Pan Left
12

RESCHU Interface and Control

Activating the Camera Feed

- Zoom the camera in and out by clicking on the blue + and – buttons found to the right of the video feed.

Click Here to Zoom In

RESCHU Interface and Control

Submitting your target

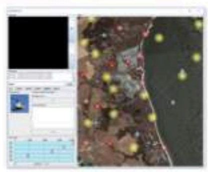
To complete the mission, you will need to submit the position of the target object.

- Make sure the target object is positioned in the viewing window, using the panning and zooming controls.
- Right-click on the target object, and choose "Submit"

RESCHU Interface and Control

Returning to the Map

After you have submitted the correct target position, the drone's mission will automatically end and you will be returned control of the main map and the other drones.



RESCHU Interface and Control

Finish experiment

- RESCHU system will automatically terminate when time is up
- You will see your final score with your performance statistics immediately after experiment

Conclusion and question

- Keep controlling UAVs to minimize distance between UAV and targets while avoid hazard zones until experiment time ends
- If you have any question, please check with the experimenter before you start
- You will have a chance to practice 10 minutes before real experiment

References

Amalberti, R. (1999). Automation in aviation: A human factors perspective. In D. Garland, J. Wise, & V. Hopkin (Eds.), *Handbook of Aviation Human Factors* (pp. 173-192). Mahwah, NJ: Erlbaum.

Asmussen, S. (2008). *Applied probability and queues* (Vol. 51): Springer Science & Business Media. .

Blitch, J. G. (2007). *Implications for Automation Assistance in Unmanned Aerial System Operator Training.* (MS), Colorado State University, Fort Collins, CO.

Boussemart, Y., Cummings, M. L., Las Fargeas, J., & Roy, N. (2011). Supervised vs. Unsupervised Learning for Operator State Modeling in Unmanned Vehicle Settings. *Journal of Aerospace Computing, Information, and Communication, 8*(11), 71-85.

Brailsford, S. C., Desai, S. M., & Viana, J. (2010). *Towards the Holy Grail: Combining System Dynamics and Discrete-Event Simulation in Healthcare*. Paper presented at the Proceedings of the 2010 Winter Simulation Conference (WSC).

Brown, C., & Campbell, B. (2000). *Technological change, training, and job tasks in a high-tech industry*. Paper presented at the 12th International Industrial Relations Research Association World Congress, Tokyo, Japan.

Clare, A. S. (2013). *Modeling Real-Time Human-Automation Collaborative Scheduling of Unmanned Vehicles.* (PhD), Massachusetts Institute of Technology, Cambridge, MA.

Cummings, M. L. (2004). *Automation Bias in Intelligent Time Critical Decision Support Systems.* Paper presented at the AIAA 3rd Intelligent Systems Conference, Chicago.

Cummings, M. L. (2014). Man vs. Machine or Man + Machine? *IEEE Intelligent Systems, 29*(5), 62-69.

Cummings, M. L., & Clare, A. S. (2015). Holistic Modelling for Human-Autonomous System Interaction. *Theoretical Issues in Ergonomics Science, 16*(3), 214-231.

Cummings, M. L., & Huang, L. (in progress,). Assessing the importance of skill-based training for supervisory control systems with increasing autonomy. *Theoretical Issues in Ergonomics Science*.

Dahai, L., Blickensderfer, E. L., Macchiarella, N. D., & Vincenzi., D. A. (2008). Transfer of Training. In D. A. Vincenzi, J. Wise, M. Mouloua, & P. A. Hancock (Eds.), *Human Factors in Simulation and Training* (pp. 49 - 60). Boca Raton, FL: CRC Press.

Ferris, T., Sarter, N., & Wickens, C. (2010). Cockpit automation: still struggling to keep up *Human Factors in Automation* (2nd ed.). Amsterdam NL. : Elsevier.

Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology, 47*, 381-391.

Fleming, E., & Pritchett, A. (2016). SRK as a Framework for the Development of Training for Effective Interaction with Multi-Level Automation. *Cognition, Technology & Work, 18*(3), 511-528.

Forrester, J. W. (1961). *Industrial Dynamics*. Cambridge, MA: MIT Press.

Forrester, J. W., & Senge, P. M. (1980). Tests for Building Confidence in System Dynamics Models. *TIMS Studies in Management Sciences, 14*(2008), 209-228.

Galata, A., Johnson, N., & Hogg, D. (2001). Learning variable-length Markov models of behavior. *Computer Vision and Image Understanding, 81*(3), 398-413.

Gao, F. (2016). *Modeling Human Attention and Performance in Automated Environments with Low Task Loading.* (PhD), Massachusetts Insitute of Technology, Cambridge, MA.

Gao, F., Clare, A. S., Macbeth, J. C., & Cummings, M. L. (2013). *Modeling the impact of operator trust on performance in multiple robot control.* Paper presented at the AAAI Spring Symposium: Trust and Autonomous Systems, Stanford, CA.

Hutchins, A. R., Clamann, M., Furth, J., & Cummings, M. L. (2016). *Determining UAV/UGV Training Effectiveness as Autonomy Increases Interim Report #1* (HAL2016-01). Retrieved from Durham, NC:

Kilgore, R., & St-Cyr, O. (2006). *The SRK Inventory: A Tool for Structuring and Capturing a Worker Competencies Analysis*. Paper presented at the Proceedings of the Human Factors and Ergonomics Society Annual Meeting, San Francisco, CA.

Lee, J., & Moray, N. (1992). Trust, control strategies and allocation of function in human-machine systems. *Ergonomics, 35*(10), 1243 - 1270.

Liu, K. K. (1997). *The Highly-Automated Airplane: Its Impact on Aviation Safety and an Analysis of Training Philosophy.* (MS), Air Force Institute of Technology, Wright-Patterson Air Force Base.

MacKenzie, I. S. (1995). Movement time prediction in human-computer interfaces. In R. M. Baecker, W. A. S. Buxton, J. Grudin, & S. Greenberg (Eds.), *Readings in human-computer interaction* (2nd ed., pp. 483-493). San Francisco: Kaufmann.

MacKenzie, I. S., Kauppinen, T., & Silfverberg, M. (2001). *Accuracy measures for evaluating computer pointing devices.* Paper presented at the ACM Conference on Human Factors in Computing Systems - CHI 2001, New York.

Mikati, N. (2010). Dependence of Lead Time on Batch Size Studied by a System Dynamics Model. *International Journal of Production Research, 48*(18), 5523-5532.

Mitchell, J. L., Yadrick, R. M., & Bennett, W. R. (1993). Estimating Training Requirements from Job and Training Models. *Military Psychology, 5*(1), 1-20.

Parasuraman, R., & Manzey, D. H. (2010). Complacency and bias in human use of automation: an attentional integration. *Human Factors, 52*(3), 381-410.

Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (2000). A Model for Types and Levels of Human Interaction with Automation. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 30*(3), 286-297.

Pentland, A., & Liu, A. (1999). Modeling and prediction of human behavior. *Neural Computation, 11*, 229-242.

Rabiner, L., & Juang, B. (1986). An introduction to hidden Markov models. *ASSP Magazine, IEEE [see also IEEE Signal Processing Magazine], 3*(1), 4-16.

Rasmussen, J. (1983). Skills, rules, and knowledge: Signals, signs, and symbols, and other distinctions in human performance models. *IEEE Transactions on Systems, Man, and Cybernetics, 13*(3), 257-266.

Rodríguez-Fernández, V., Gonzalez-Pardo, A., & Camacho, D. (2016). *Finding behavioral patterns of UAV operators using Multichannel Hidden Markov Models.* Paper presented at the IEEE Computational Intelligence Symposium Series, Athens, Greece.

Sarter, N. B., Woods, D. D., & Billings, C. E. (1997). Automation surprises. In G. Salvendy (Ed.), *Handbook of Human Factors and Ergonomics* (pp. 1926-1943). New York: Wiley.

Saxena, Y., & Wang, S. (2017). *The Effect of Training on Situation Awareness in a Simulated UAV Task*. Human-Robot Interaction class, ME 555. Duke University. Durham, NC.

Schwarz, G. (1978). Estimating the dimension of a model. *The annals of statistics, 6*(2), 461-464.

Singh, A. L., Tiwari, T., & Singh, I. L. (2009). Effects of Automation Reliability and Training on Automation-Induced Complacency and Perceived Mental Workload. *Journal of the Indian Academy of Applied Psychology, 35*, 9-22.

Sterman, J. D. (2000). *Business Dynamics: Systems Thinking and Modeling for a Complex World*. New York: Irwin/McGraw-Hill.

Stimpson, A., & Cummings, M. (2014). Assessing Intervention Timing in Computer-Based Education Using Machine Learning Algorithms. *IEEE Open Access, 2*, 78-87.

Stimpson, A. J., Buinhas, L. S., Bezek, S., Boussemart, Y., & Cummings, M. (2012). *A Model-Based Measure to Assess Operator Adherence to Procedures.* Paper presented at the HFES 56th Annual Meeting, Boston, MA.

Suzuki, K., & Jansson, H. (2003). An analysis of driver's steering behaviour during auditory or haptic warnings for the designing of lane departure warning system. *JSAE Review, 24*, 65-70.

Suzuki, T., Sekizawa, S., Inagaki, S., Hayakawa, S., Tsuchida, N., Tsuda, T., & Fujinami, H. (2005). *Modeling and Recognition of Human Driving Behavior based on Stochastic Switched ARX model.* Paper presented at the 44th IEEE Conference onDecision and Control CDC-ECC '05. , Seville, Spain.

Tsang, P. S., & Vidulich, M. A. (2003). *Principles and Practice of Aviation Psychology*. Mahwah, NJ: Lawrence Erlbaum Associates Publishers.

Tvaryanas, A. P., Thompson, William T., Constable, Stefan H. . (2006, July 2006). Human Factors in Remotely Piloted Aircraft Operations: HFACS Analysis of 221 Mishaps over 10 Years. *Aviation, Space, and Environmental Medicine, 77*.

United States Air Force Scientific Advisory Board. (2011). *Operating Next-Generation Remotely Piloted Aircraft for Irregular Warfare* (SAB-TR-10-03). Retrieved from Washington DC:

United States Government Accountability Office. (2017). *UNMANNED AERIAL SYSTEMS: Air Force and Army Should Improve Strategic Human Capital Planning for Pilot Workforces* (GAO-17-53). Washington DC.

White, A. S. (2003). *A Qualitative Systems Dynamic Model of Human Monitoring of Automated Systems.* Paper presented at the 21st International Conference of the Systems Dynamics Society, New York, NY.

Wickens, C. D., & Dixon, S. R. (2007). The benefits of imperfect diagnostic automation: a synthesis of the literature. *Theoretical Issues in Ergonomics Science, 8*(3).
Wiener, E. L., & Curry, R. E. (1980). Flight-Deck Automation: Promises and Problems. *Ergonomics, 23*(10), 995-1011.