Modeling Human Workload in Unmanned Aerial Systems

J. J. Moore, R. Ivie, T.J. Gledhill, E. Mercer and M. A. Goodrich
Computer Science Department
Brigham Young University
Provo, UT

Abstract

Unmanned aerial systems (UASs) often require multiple human operators fulfilling diverse roles for safe correct operation. Although some persuasively dispute the utility of minimizing the number of humans needed to administer a UAS (Murphy and Burke 2010), it is a long-standing objective for many designers. This paper presents work toward understanding how workload is distributed between multiple human operators and multiple autonomous system elements in a UAS across time, with an ultimate goal to reduce the number of humans in the system. The approach formally models the *actors* in a UAS as a set of communicating finite state machines, modified to include a simple form of external memory. The interactions among actors are then modeled as a directed graph. The individual machines, one for each actor in the UAS, and the directed graph are augmented with workload metrics derived from a review of the relevant literature. The model is implemented as a Java program, which is analyzed by the Java Pathfinder (JPF) model checker. JPF generates workload profiles showing how workload changes through time. To demonstrate the utility of the approach, this paper presents a case study on a wilderness search and rescue (WiSAR) UAS analyzing two different mission outcomes. The generated workload profiles are shown to be consistent with known features of actual workload events in the WiSAR system.

Introduction

Unmanned aerial systems (UASs), ranging from large military-style Predators to small civilian-use hovercraft, usually require more than one human to operate. It is perhaps ironic that a so-called “unmanned” system requires multiple human operators, but when a UAS is part of a mission that requires more than moving from point A to point B, there are many different tasks rely on human input including: operating the UAS, managing a payload (i.e., camera), managing mission objectives. Some persuasively argue that this is desirable because different aspects of a mission are handled by humans trained for those aspects (Murphy and Burke 2010), but since human resources are expensive many others argue that it is desirable to reduce the number of humans involved.

However, the question of *how* to reduce the number of humans while maintaining a high level of robustness is an open question. Some progress has been made by improving autonomy using, for example, automatic path-planning (Wong, Bourgault, and Furukawa 2005; Bortoff 2000; Pettersson and Doherty 2006; Quigley et al. 2006; Nelson et al. 2006), and automated target recognition (Morse, Engh, and Goodrich 2010; Dasgupta 2008; Barber et al. 2006). However, careful human factors suggest that the impact of changes in autonomy are often subtle and difficult to predict, and this decreases confidence that the combined human-machine system will be robust across a wide range of mission parameters (Kaber and Endsley 2004; Chen, Barnes, and Harper-Sciarini 2011; Chen, Haas, and Barnes 2007).

We argue that one reason for the limitations of prior work in measuring workload is that the level of resolution is too low. For example, although the NASA TLX dimensions include various contributing factors to workload (e.g., physical effort and mental effort), the temporal distribution of workload tends to be “chunked” across a period of time. Secondary task measures can provide a more detailed albeit indirect breakdown of available cognitive resources as a function of time (Kaber and Riley 1999), but with insufficient explanatory power for what in the task causes workload peaks and abatement. Cognitive workload measures, including those that derive from Wickens’ multiple resource theory (Wickens 2002), provide useful information about the causes of workload spikes, but these measures have not been widely adopted; one way to interpret this paper is as a step toward robust implementations of elements of these measures. Finally, measures derived from cognitive models such as ACT-R are providing more low-level descriptions of workload which potentially include a temporal history (Lebiere, Jentsch, and Ososky 2013), but these approaches may require a modeling effort that is too time-consuming to be practical for some systems.

This paper presents a model of four human roles for a UAS-enabled wilderness search and rescue (WiSAR) task, and is based on prior work on designing systems through field work and cognitive task analyses (Adams et al. 2009; Goodrich et al. 2008). We first identify a suite of possible workload measures based on a review of the literature. We then consider seven *actors* in the team: the UAV, the operator and the operator’s GUI, the video analyst and the analyst’s GUI, the mission manager, and a role we call the “parent search” which serves to connect the UAS technical search team to the other components of the search enterprise. We present the formal model of each of these actors using finite state machines, and then discuss how the connections between these state machines defines what we call a *Directed Team Graph* (DiTG) that describes who communicates with whom and under what conditions. We then augment the model to be able to encode specific metrics based on a subset of the measures identified in the literature. Using the Java PathFinder model checker, we then create temporal profiles for each of the workload metrics and check consistency of the temporal profiles by associating workload peaks and abatements with likely causes.

Workload Categories

In this section, we present a brief review of the set of workload categories from which we distill a set of workload metrics. We restrict attention to three general categories of workload metrics: cognitive, temporal, and algorithmic. A fourth relevant workload category is the cost of maintaining team constructs like shared situation awareness (Elias and Fiore 2011), but we leave a discussion of team-related workload to future work.

Cognitive

Cognitive workload describes the difficulties associated with managing various signals, decisions, and actions relevant to a particular task or goal (Moray et al. 1991; Lebiere, Jentsch, and Ososky 2013; Goodrich 2004; Chadwick et al. 2004). We adopt a simple form of Wickens' multiple resource theory (Wickens 2002), and make the simplifying assumption that cognitive workload can be divided into two categories: parallel sensing and sequential decision making. We further restrict the sensing channels to visual and auditory modalities, ignoring haptic. Parallel sensing means that it possible for a human to perceive complementary stimuli over different channels. An example of this would be an individual hearing their call sign on the radio while analyzing video. However, when multiple signals may occur over the same channel at the same time, this induces attentional workload for the human (i.e., two audio signals or two visual items needing attention). Sequential decision making occurs when a decision must be made, where we have adopted the assumption made by Wickens and supported by work in the psychology of attention (Pashler 1998) that a "bottle-neck" occurs when multiple channels (a) require a either a decision to be generated or (b) exceed the limits of working memory.

Algorithmic

Algorithmic workload results from the difficulty of bringing a task to completion. Adopting a common model from artificial intelligence (Murphy 2000), and consistent with Wickens' three stage multiple resource model, we assume that this is comprised of three phases: sense, plan, and act. During the sensing and perception phase, the actor takes all active inputs, interprets them, and generates a set of relevant decision-making parameters. In the planning phase the actor reviews the breadth of choices available and selects one, possibly using search or a more naturalistic decision-making process like recognition-primed decision-making (Zsambok and Klein 1997) or a cognitive heuristic (Todd 1999). We assume that the workload in these two phases is related to the number of choices the actor has, allowing us to use big-O analysis from computational complexity theory to describe the workload associated with sensing and planning; in this paper we make the unrealistic assumption that workload from computational complexity is $O(n)$, but include an explicit temporal component that allows us to detect when multiple decisions "pile up" at the same time. During the acting phase, the actor either follows through with the decision or disregards it. The workload in this section is entirely dependent on the length and difficulty of executing the plan. Before concluding, we note that workload is highly dependent on the experience of the actor (Zsambok and Klein 1997), but we leave a careful treatment of this to future work.

Temporal

Temporal workload deals with the scheduling of prioritized, infrequent, and/or repetitious tasks (Dessouky, Moray, and Kijowski 1995; Moray et al. 1991). Various measures have been proposed, but we are most interested in those related to so-called "fan-out", meaning the number of tasks that a single actor can manage (Goodrich 2010; Olsen Jr. and Wood 2004; Crandall et al. 2005; Cummings et al. 2007a). There are two aspects of temporal workload that particularly important. The first aspect consist of both timing deadlines and the ordering of when tasks are addressed. When a task is constrained by (a) the time by which the task must be completed or (b)the need to complete other tasks before or after the given task then (c) it causes scheduling pressure and workload (Mau and Dolan 2006). The second aspect is operational tempo, which represents how frequently new tasks arrive. High tempo is especially relevant to workload because it may result either in insufficient time to complete all required tasks or in the neglect of essential tasks. From a scheduling or queuing theory perspective, operational tempo impacts workload by causing pressure to manage the rate of arrival and the response time of the decision.

Actor Model

In previous work (Gledhill, Mercer, and Goodrich 2013), we represented each *actor*, both human and autonomous component, of the WiSAR search team as a Mealy state machines as has been done in other models (Bolton, Bass, and Siminiceanu 2013). However the Mealy state machine model does not lend itself well to workload analysis so this paper uses Moore machines[1] Moore machines confine model non-determinism to state changes, which minimizes the possible sources of non-determinism and simplifies the validation process. We augment these Moore machines with a simple form of memory. This memory increases the computational power of the machine, but the simple form we use keeps the complexity of validation manageable.

[1]The outputs or signals from Mealy machines are based both on the inputs into the machine and the state itself, and Moore machines restrict changes to the input to state changes.

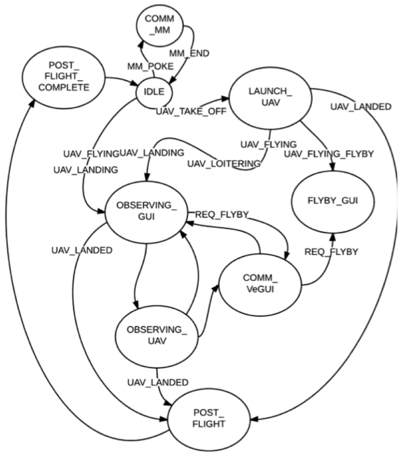

Figure 1: Example actor model from prior work.

We now present a formal description of each component of the model

Actors

Actors represent the human decision-makers and autonomous elements of the WiSAR team. An actor is composed of a set of states S, an initial state s_0, a set of inputs from the environment Σ_{env}, a set of inputs from other actors on the team Σ_{team}, a set of outputs Λ_{out}, a simple form of memory Ω_{men}, and a transition function that determines the next state from the inputs and memory δ. Formally, we denote an actor as:

$$Actor = (S, s_0, \Sigma_{\text{env}}, \Sigma_{\text{team}}, \Lambda_{\text{out}}, \Lambda_{\text{out}}\Omega_{\text{mem}}, \delta) \quad (1)$$

Outputs

An actor's output has two components: signals to other actors on a team and a *duration* parameter that represents the time required for the actor to complete its transition to the next state. Thus, $\Lambda_{\text{out}} = \Sigma_{\text{team}} \times \Re^{+}$. The duration represents the relative difficulty of the task(s) associated with the transition. We justify this by assuming that all tasks are performed at a constant rate, thus more difficult tasks take longer, but note that future work should address this restrictive assumption.

Transitions

A transition is a relation on the cross product of inputs (a state, environment signal, signals from other actors, and element of memory) with outputs (a next state, an output, and a modification to memory). Thus,

$$\delta : [S \times \Sigma_{\text{env}} \times \Sigma_{\text{team}} \times \Omega_{\text{mem}}] \times [S \times \Sigma_{\text{team}} \times \Omega_{\text{mem}}], \quad (2)$$

where we have used brackets to separate inputs from outputs.

Given an actor's current state, set of input signals, and memory, it is possible for multiple transitions to be possible. This occurs because we assume that multiple environment signals or inter-actor signals may be occurring at the same time, which are all perceived by the actor since we assume perception is a parallel operation. Thus, it is useful to explicitly note the number of possible transitions that are possible from a given state. A transition is considered *enabled* when all of its input requirements are met and *disabled* otherwise.

Current State

When an actor is in a given state, it is useful to explicitly denote the set of enabled and disable transitions. This allows a model-checker to use this information to create an estimate of algorithmic workload, where we assume that algorithmic workload is a function of the number of choices available to the actor. Thus, we allow the current state to give an workload signal

$$s_0^{\text{work sig}} = (T_{enabled}, T_{disabled}) : T_{enabled} \cap T_{disabled} = \emptyset \quad (3)$$

Declarative Memory

This memory represents internal facts stored by an actor and used in decision making. This memory takes the form of internal variables within an actor SUCH AS ???

Directed Team Graph (DiTG)

As described in the previous section, we modeled a WiSAR team as a collection of Mealy machines, called *actors*, augmented with a simple form of memory. A key element of these actors is that inputs to one actor can be outputs from another actor. We can therefore create a directed graph from actor to actor, with an edge from actor A to actor B existing if the output from actor A is a possible input to actor B. We call this graph a *Directed Team Graph* (DiTG); Figure 2 illustrates the DiTG for the WiSAR team used in this paper.

Labeled Edges

Using the fact that multiple resource theory (Wickens 2002) indicates that visual and auditory channels can be perceived in parallel, it is useful to label the edges in the graph with the a *channel type* as in Figure 3). These labels allow the model-checker to identify when multiple signals are given to a single actor over the same channel. When this occurs, we expect actor workload to be high.

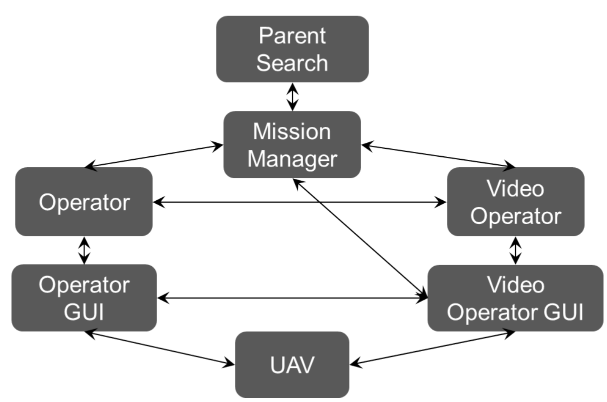

Figure 2: High Level DiTG

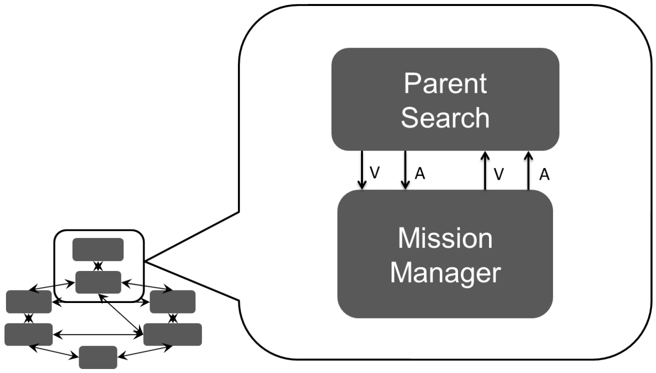

Figure 3: Detail view of DiTG: V is a Visual channel and A is an Audio channel

Overview of Implementation

We represent a system as a DiTG, a collection of actors connected to one another by a set of channels. Whenever the state of the system changes, an actor will petition from its current state, a list of enabled transitions thus defining what decisions can be made. The actor may then activate one of these transitions.

Because transitions from one state to another take time, as encoded in the *Duration* element of the output, it is useful to label transitions as either *active* or *fired*. Note that when we presented the model of an actor, we labeled transitions as *enabled* and *disabled*, which was sufficient when we considered the actor in isolation. When we consider the workload of an actor as part of the overall team, workload depends on what is going on with other team members, so we have added the active and fired labels to so that we can determine when an enabled transition (meaning a possible choice available to an actor) is chosen by an actor (making it active) and when the actor completes the work required to produce the next state (the transition fires).

From an implementation perspective, when a transition becomes active it creates temporary output values for declarative memory and channels. These temporary values are then applied to the actual declarative memory and channel values once the transition fires.

Actor vs Tasks

For our model we never explicitly define a single task. Instead we define actors, states, and transitions. Each transition defines its own perceptual, cognitive, response, and declarative resources (Salvucci and Taatgen 2008), allowing the model to represent multiple possible tasks. In this way, an actor's state determines what task(s) are being performed, achieving multi-tasking without explicitly defining tasks.

¡¡¡¡¡¡¡ HEAD =======

Model Creation

To simplify the modeling process and ensure rigorous model creation we developed a transition language, similar to a Kripke structure, which allows models to be expressed as a list of Actor transitions. A parser then automatically generates the classes required to run the model simulation. The transition language uses the following structure.

$$(s_{current}, [\phi_{input} = value, \ldots], [\omega_{input} = value, \ldots], duration) \times (s_{next}, [\phi_{output} = value, \ldots], [\omega_{output} = value, \ldots]) \quad (4)$$

The language is compiled to a Java program suitable to run standalone as a simulation or analyzed by the JPF model checker to create workload profiles. ¿¿¿¿¿¿¿ refs/remotes/origin/master

Workload Metrics

We are now in a position to combine the three categories of workload (cognitive, algorithmic, and temporal) with the formal model of the actors and team to generate a set of workload metrics. Because the categories include many possible measurements that are beyond the scope of the paper, we use labels for the workload metrics that are slightly different from the workload categories. As shown in Figure 5, cognitive workload is measured using metrics under the *resource* workload label, algorithmic under *decision*, and temporal workload is labeled the same.

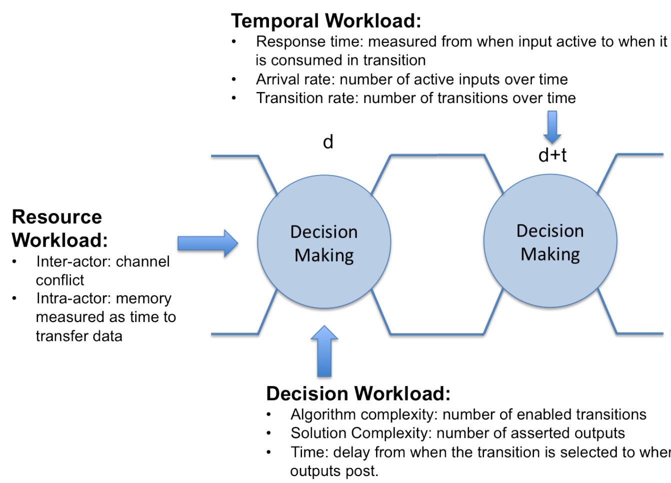

Figure 4: Workload in the model.

Resource Metrics

Cognitive workload is separated into both inter-actor communication and actor memory load. JPF listens to the output of the state transitions and records outputs to Ω_{men}. JPF also listens to all channel reads, noting how many communications are on the visual and auditory input channel, identified by the labeled edges of the DiTG, of each actor.

Decision Metrics

Algorithmic workload can be broken down into the timing, the complexity of the algorithm, and the complexity of the solution. JPF is instrumented to measure timing as the time between a transition becomes active and when it fires. JPF is also instrumented to count the number of enabled transitions in each state as it is visited, giving us an $O(n)$ estimate of how the number of choices affects workload. Finally, JPF is instrumented to count the number of output signals generated (Σ_{team}), yielding an $O(n)$ estimate for the complexity of computing the required outputs of a state given its inputs.

Temporal Metrics

Temporal workload includes three metrics: operations tempo, arrival rate, and response time. JPF measures the average op-tempo counting how many transitions occur over a course of the simulation. JPF measures arrival rate by tracking the rate at which inputs become active. Finally, JPF measures response time by measuring the time from when an input goes active to the point when it is read by the actor. =======

Metric Classes

Conversion of the Workload theory into quantifiable measurements necessitated the creation of workload metric classes (see Figure 5). In the interest of finding areas to consolidate actors we left off measuring team workload. Given that cognitive workload describes the difficulties presented by managing resources such as memory and inputs we named that metric class resource. Algorithmic workload analyzes the difficulties presented by decisions giving rise to the decision metrics.

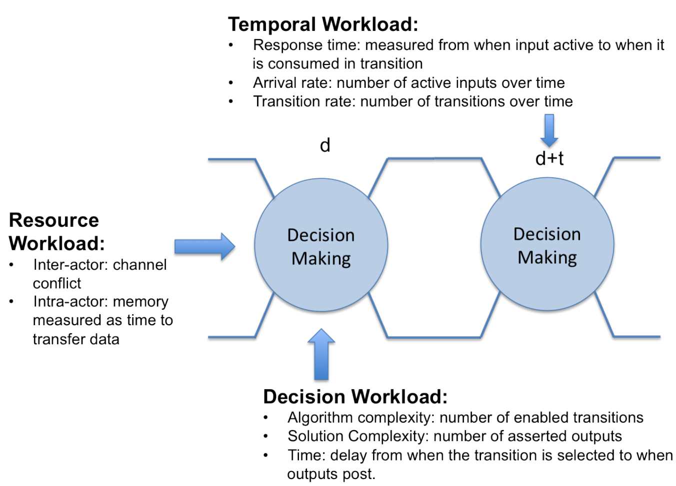

Figure 5: Workload in the model

Resource Metrics

Cognitive workload is separated into both inter-actor communication and intra-actor memory analysis. We instrumented JPF to listen for recorded memory accesses to handle the intra-actor workload generated. JPF is then further instrumented to listen to all channel reads. By using these two listeners we gained the ability to maintain an accurate view of how the cognitive workload fluctuates over time.

Temporal Metrics

The DiTG lent itself to measuring temporal workload using three separate metrics. The op-tempo is measured by recording how many transitions occur over a course of the simulation. Tracking the rate at which inputs become active gives an accurate reflection for the arrival rate of data. The response time is calculated by measuring the time from when an input goes active to the point when it is read by the actor.

Decision Metrics

Algorithmic Workload can be broken down into the timing, the complexity of the algorithm, and the complexity of the solution. The timing is calculated by measuring the time when a transition is chosen till the outputs are posted. The point when the outputs go high represents the time when that task is complete and the actor has moved on to her next task. In this fashion we measure the time it takes to execute a given task. By counting the number of possible transitions we can track the number of choices the actor has and therefore measure the algorithmic complexity. The solution complexity is analyzed by counting the number of outputs a transition activates. This last measurement is not very precise since some tasks are more complex than others despite requiring the same number of actions. The level of abstraction does however give us sufficient accuracy to aid in our workload predictions while keeping our metrics at a manageable simplicity. ‘

Results

In the interest of consolidating operators it is critical to find an accurate measurement that detects situations that exceed the capacity of a given human. One way to detect this is by building a map of each actor's workload as a function of time. JPF explores all possible paths the model can take and returns the ones that violate the model's criteria. By augmenting our model with the metrics described above we can identify all possible areas of high workload.

We propose three levels of increasing validity for evaluating the approach in the paper. The first level, the one used in this paper, is to check for *consistency*. We say that the approach is *consistent* if the workload peaks, valleys, and trends match what we know about a small set of given situations; in other words, the approach is consistent if it matches our expectations on tasks that we know a lot about.

The second level, which is an area for future work, is to check for *sensitivity*. We say that the approach is sufficiently *sensitive* if we can use JPF to create new scenarios that have very high or very low workloads, and we can then generate a satisfactory explanation for the levels of workload by evaluating the new scenarios. The third level, which is also an area of future work, is to *substantiate* workload levels using experiments with human participants by comparing the perceived workload of humans with those predicted by the model.

In this paper, we restrict attention to finding areas of high and low workload, and then checking these areas for consistency. We evaluated consistency using two scenarios. The first is when the Video Operator was able to identify the target during a flight without any complications occurring (see Figure 6). For the first 40 time steps everything behaves as expected with low to moderate workload. An initial workload bump occurs as actors exchange information necessary to start a search. At time step forty we see a dramatic deviation from the norm. This is a result of constant information passing between the GUIs and the operators. Since there is a constant passing of data between the machinery and the operators, it is only logical that the workload would increase substantially. However the results indicate that a weighting should be instigated to balance the three workload measures rather than allowing the single category to dominate the system. In our sensitivity study we will verify whether this is a fault in our metrics or if this one scenario lends itself to the distortion. WHAT HAPPENS TO CAUSE THIS SPIKE, AND WHY DOES IT ABATE?

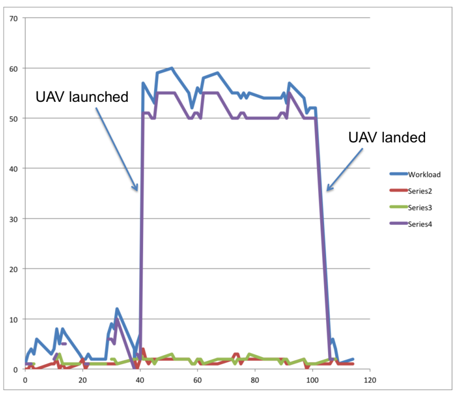

Figure 6: Workload over an uneventful flight.

The second simulation includes a situation where, after a short period of flight, the battery rapidly fails. In this particular situation the operator was unable to respond quickly enough to land the UAV before it crashed (see Figure 7). There is an immediate spike in the temporal workload but, surprisingly,the workload then decreases back to normal levels in just five time steps. WHY? The second spike indicates an unexpected fluctuation in options among one of the actors, which will have to be investigated further to verify if this is an accurate response or if a flaw in the model had slipped past the verification stage of development. Finally, as would be expected, when the UAV crashed there was a small spike in the workload before everything came to a halt.

These two simulations demonstrate that workload rises and falls as expected and provide evidence that the approach is consistent with expectations. However, there is also one event that was surprising and that needs further study.

Related Work

This work is an extension of previous work which focused on modeling human machine systems, specifically WiSAR. This work extends this model to incorporate the measurement of workload (Gledhill, Mercer, and Goodrich 2013).

Multiple resource theory plays a key role in how we are measuring workload (Wickens 2002). The multiple resource

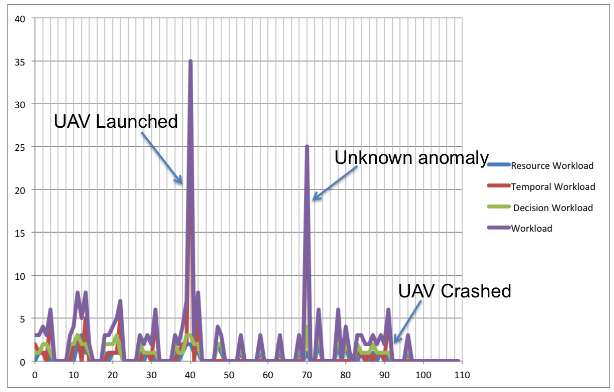

Figure 7: Emergency battery failure simulation

model defines four categorical dimensions that account for variations in human task performance. A task can be represented as a vector of these dimensions. Tasks interfere when they share resource dimensions. Using these vectors, Wickens defined a basic workload measure consisting of the task difficulty (0,1,2) and the number of shared dimensions. Using this metric it is possible to predict task interference by looking at tasks which use the same resource dimensions. Our model differs in that we do not explicitly define tasks, instead we use Actor state transitions which may imply any number of concurrent tasks. The transition then informs us of which resources are being used and for how long.

Threaded cognition theory states that humans can perform multiple concurrent tasks that do not require executive processes (Salvucci and Taatgen 2008). By making a broad list of resource assumptions about humans, threaded cognition is able to detect the resource conflicts of multiple concurrent tasks. Our model differs from threaded cognition theory in that it does not allow learning nor does our model distinguish between perceptual and motor resources. In almost all other aspects our model behaves in a similar fashion.

Related work on temporal workload has attempted to predict the number of UAVs an operator can control, otherwise known as *fan-out* (Cummings et al. 2007b; Olsen Jr. and Wood 2004; Crandall et al. 2005). This work used queuing theory to model how a human responds in a time sensitive multi-task environment. Queuing theory is helpful in determining the temporal effects of task performance by measuring the difference between when a task was received and when it was executed. Actors can only perform a single transition at a time, similar to queuing theory, but it is possible for each state to take input from multiple concurrent tasks which differs from standard models of queuing theory.

ACT-R is a cognitive architecture which attempts to model human cognition and has been successful in human-computer interaction applications (Anderson et al. 2004; Lebiere, Jentsch, and Ososky 2013). The framework for this architecture consists of modules, buffers, and a pattern-matcher which in many ways are very similar to our own framework. The major difference is that ACT-R includes higher levels of modeling detail, such as memory access time, task learning, and motor vs perceptual resource differences. Our model exists at a higher level of abstraction.

Summary and Future Work

This paper proposed a model-checking approach to analyzing human workload in an Unmanned Aerial System. Humans and other autonomous actors were modeled as modified Mealy machines, yielding a directed graph representing team communication. Workload categories were distilled from the literature, and the models of the actors and team were augmented so that specific workload metrics could be obtained using model-checking. Preliminary analysis demonstrated a weak level of validity, namely, that the temporal workload profile was consistent with expected behavior for a set of well-understood situations. For these scenarios, inter-actor communication was a primary cause of spikes in workload.

For future work, we plan to add an actor to represent a human performing the role of integrating the UAS into the National Air Space. operations. A sensitivity study, followed by an experiment with real human users is needed to understand and justify the workload measures. One we have verified that our system analyzes workload correctly, it will be useful to design a GUI optimized to managing workload and to formulate a generalized model that will has application to other human-machine systems.

Acknowledgment

The authors would like to thank the NSF IUCRC Center for Unmanned Aerial Systems, and the participating industries and labs, for funding the work.

References

Adams, J. A.; Humphrey, C. M.; Goodrich, M. A.; Cooper, J. L.; Morse, B. S.; Engh, C.; and Rasmussen, N. 2009. Cognitive task analysis for developing unmanned aerial vehicle wilderness search support. *Journal of cognitive engineering and decision making* 3(1):1–26.

Anderson, J. R.; Bothell, D.; Byrne, M. D.; Douglass, S.; Lebiere, C.; and Qin, Y. 2004. An integrated theory of the mind. *Psychological review* 111(4):1036.

Barber, D. B.; Redding, J. D.; McLain, T. W.; Beard, R. W.; and Taylor, C. N. 2006. Vision-based target geo-location using a fixed-wing miniature air vehicle. *Journal of Intelligent and Robotic Systems* 47(4):361–382.

Bolton, M.; Bass, E.; and Siminiceanu, R. 2013. Using formal verification to evaluate human-automation interaction: A review. *IEEE Transactions on Systems, Man, and Cybernetics* 43(3):488–503.

Bortoff, S. 2000. Path planning for uavs. In *American Control Conference, 2000. Proceedings of the 2000*, volume 1, 364–368 vol.1.

Chadwick, R. A.; Gillan, D. J.; Simon, D.; and Pazuchanics, S. 2004. Cognitive analysis methods for control of multiple robots: Robotics on $5 a day. In *Proceedings of HFES*.

Chen, J. Y.; Barnes, M. J.; and Harper-Sciarini, M. 2011. Supervisory control of multiple robots: Human-performance issues and user-interface design. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 41(4):435–454.

Chen, J. Y.; Haas, E. C.; and Barnes, M. J. 2007. Human performance issues and user interface design for teleoperated robots. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 37(6):1231–1245.

Crandall, J. W.; Goodrich, M. A.; Jr., D. R. O.; and Nielsen, C. W. 2005. Validating human-robot interaction schemes in multi-tasking environments. *IEEE Transactions on Systems, Man and Cybernetics — Part A: Systems and Humans* 35(4).

Cummings, M. L.; Nehme, C. E.; Crandall, J.; and Mitchell, P. 2007a. *Developing Operator Capacity Estimates for Supervisory Control of Autonomous Vehicles*, volume 70 of *Studies in Computational Intelligence*. Springer. 11–37.

Cummings, M. L.; Nehme, C. E.; Crandall, J.; and Mitchell, P. 2007b. Predicting operator capacity for supervisory control of multiple uavs. In *Innovations in Intelligent Machines-1*. Springer. 11–37.

Dasgupta, P. 2008. A multiagent swarming system for distributed automatic target recognition using unmanned aerial vehicles. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on* 38(3):549–563.

Dessouky, M. I.; Moray, N.; and Kijowski, B. 1995. Taxonomy of scheduling systems as a basis for the study of strategic behavior. *Human Factors* 37(3):443–472.

Elias, J., and Fiore, S. M. 2011. Commentary on the coordinates of coordination and collaboration. In Salas, E.; Fiore, S. M.; and Letsky, M. P., eds., *Theories of Team Cognition: Cross-Disciplinary Perspectives*, Series in Applied Psychology. New York: Routledge. In Press.

Gledhill, T. J.; Mercer, E.; and Goodrich, M. A. 2013. Modeling uass for role fusion and human machine interface optimization. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*.

Goodrich, M. A.; Morse, B. S.; Gerhardt, D.; Cooper, J. L.; Quigley, M.; Adams, J. A.; and Humphrey, C. 2008. Supporting wilderness search and rescue using a camera-equipped mini UAV. *Journal of Field Robotics* 25(1-2):89–110.

Goodrich, M. A. 2004. Using models of cognition in HRI evaluation and design. In *Proceedings of the AAAI 2004 Fall Symposium Series: The Intersection of Cognitive Science and Robotics: From Interfaces to Intelligence*.

Goodrich, M. A. 2010. On maximizing fan-out: Towards controlling multiple unmanned vehicles. In Barnes, M., and Jentsch, F., eds., *Human-Robot Interactions in Future Military Operations*. Surrey, England: Ashgate Publishing.

Kaber, D. B., and Endsley, M. R. 2004. The effects of level of automation and adaptive automation on human performance, situation awareness and workload in a dynamic control task. *Theoretical Issues in Ergonomics Science* 5(2):113–153.

Kaber, D. B., and Riley, J. M. 1999. Adaptive automation of a dynamic control task based on secondary task workload measurement. *International journal of cognitive ergonomics* 3(3):169–187.

Lebiere, C.; Jentsch, F.; and Ososky, S. 2013. Cognitive models of decision making processes for human-robot interaction. In *Virtual Augmented and Mixed Reality. Designing and Developing Augmented and Virtual Environments*. Springer. 285–294.

Mau, S., and Dolan, J. 2006. Scheduling to minimize downtime in human-multirobot supervisory control. In *Workshop on Planning and Scheduling for Space*.

Moray, N.; Dessouky, M. J.; Kijowski, B. A.; ; and Adapathya, R. 1991. Strategic behavior, workload, and performance in task scheduling. *Human Factors* 33(6):607–629.

Morse, B. S.; Engh, C. H.; and Goodrich, M. A. 2010. Uav video coverage quality maps and prioritized indexing for wilderness search and rescue. In *Proceedings of the 5th ACM/IEEE RAS International Conference on Human-Robot Interaction*.

Murphy, R. R., and Burke, J. L. 2010. The safe human-robot ratio. In Barnes, M., and Jentsch, F., eds., *Human-Robot Interaction in Future Military Operations*. Ashgate Publishing. chapter 3, 31–49.

Murphy, R. R. 2000. *Introduction to AI Robotics*. MIT Press.

Nelson, D. R.; Barber, D. B.; McLain, T. W.; and Beard, R. 2006. Vector field path following for small unmanned air vehicles. In *Proceedings of the American Control Conference*.

Olsen Jr., D. R., and Wood, S. B. 2004. Fan-out: Measuring human control of multiple robots. In *Proceedings of Human Factors in Computing systems*.

Pashler, H. 1998. *The Psychology of Attention*. Cambridge, Massachusetts, USA: MIT Press.

Pettersson, P. O., and Doherty, P. 2006. Probabilistic roadmap based path planning for an autonomous unmanned helicopter. *Journal of Intelligent and Fuzzy Systems* 17(4):395–405.

Quigley, M.; Barber, B.; Griffiths, S.; and Goodrich, M. A. 2006. Towards real-world searching with fixed-wing mini-UAVs. In *Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*.

Salvucci, D. D., and Taatgen, N. A. 2008. Threaded cognition: an integrated theory of concurrent multitasking. *Psychological review* 115(1):101.

Todd, G. G. P. M. 1999. *Simple Heuristice That Make Us Smart*. New York: Oxford University Press.

Wickens, C. D. 2002. Multiple resources and performance prediction. *Theoretical issues in ergonomics science* 3(2):159–177.

Wong, E.-M.; Bourgault, F.; and Furukawa, T. 2005. Multi-vehicle Bayesian search for multiple lost targets. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 3169–3174.

Zsambok, C. E., and Klein, G., eds. 1997. *Naturalistic Decision Making*. Hillsdale, N.J.: Lawrence Erlbaum Associates.