

Predictive Planning for Heterogeneous Human-Robot Teams

Sameera S. Ponda, Han-Lim Choi, Jonathan P. How *

This paper addresses the problem of task allocation over a heterogeneous team of human operators and robotic agents with the object of improving mission efficiency and reducing costs. A distributed systems-level predictive approach is presented which simultaneously plans schedules for the human operators and robotic agents while accounting for agent availability, workload and coordination requirements. The approach is inspired by the Consensus-Based Bundle Algorithm (CBBA), a distributed task allocation framework previously developed by the authors, which is used to perform the task coordination for the team in a dynamic environment. Results show that predictive systems-level planning improves mission performance, distributes workload efficiently among agents, reduces operator over-utilization and leads to coordinated agent behavior.

I. Introduction

Modern day network centric operations involve large teams of agents, with heterogeneous capabilities, interacting together to perform missions. These missions involve executing several tasks, such as conducting intelligence, reconnaissance and surveillance, with the possibility of follow-up tasks like target classification or rescue operations.^{1,2} Within the heterogeneous team some specialized agents are better suited to handle certain types of tasks than others. For example, UAVs equipped with video can be used to perform search and surveillance, human operators can be used for classification tasks, ground teams can be deployed to perform rescue operations, etc. Ensuring proper coordination and collaboration between agents in the team is crucial to efficient and successful mission execution. To this effect it is of interest to develop autonomous task allocation methods to improve mission coordination. This paper addresses the problem of developing a decentralized task allocation algorithm for a large network of human operators and robotic agents. The objective is to distribute the tasks across the entire network of heterogeneous agents to accomplish the overall mission efficiently and successfully. We are implementing a predictive systems-level approach that simultaneously plans schedules for all agents in the heterogeneous team. This predictive systems-level algorithm should ensure efficient coordination amongst all agents to reduce mission execution time, expected mission cost, and operator over-utilization, while maintaining robustness and flexibility of the system to handle uncertain and dynamic environments.

Prior research in human robot interaction (HRI) has primarily focused on low-level details and specific interactions between robotic agents and human operators (small teams), but the directions of HRI research are widespread. For example, one area of research focuses on improving the performance of human operators given specific robotic agents, involving problems such as predicting the utilization of operators controlling multiple unmanned vehicles, designing efficient user interfaces which improve the operators' ability to effectively control robotic agents, or exploring workload mitigation strategies for supervisors overseeing unmanned vehicles.^{10-12, 19, 20} Other HRI research involves exploring the effects of changing different human-robot team parameters, such as varying vehicle autonomy levels or varying the number of vehicles controlled by each operator.^{15, 23} These previous approaches attempt to improve the performance of one type of agent (either human or robotic) given expected plans for the other agents they are interacting with, but do not consider optimization over both human and robotic agents simultaneously. In addition, this previous work focuses predominantly on the case of a single operator controlling multiple vehicles but does not consider coordinated tasks involving multiple operators and multiple unmanned vehicles. Furthermore, the scope of this previous work has typically restricted the problem size to small-scale teams of operators and robots.

For large heterogeneous teams involving multiple human and robotic agents, it is better to consider a distributed systems-level approach that *simultaneously* optimizes assignments (or schedules) for all agents.

*S. Ponda, H.-L. Choi, and J. P. How are with the Dept. of Aeronautics and Astronautics, MIT, Cambridge, MA, {sponda, hanlimc, jhow}@mit.edu

This systems-level planner should account for coordination requirements between the heterogeneous agents and should capture the temporal relationships and probabilistic dependencies between tasks, thus enabling coordinated task execution and smooth transitions between tasks. By modeling the interaction between agents in the team and directly accounting for agent availability and workload up front, a systems-level approach would improve mission efficiency. In realistic ISR missions, several unmanned vehicle tasks are known to require human input and support, for example, UAVs are used for collecting imagery and surveillance data, but a human operator is required to manually perform target identification and classification on the imagery obtained. Previous research has addressed the problem of tasking human operators in charge of classifying targets encountered by surveillance unmanned vehicles. For example, Savla et. al.²² present an approach that uses dynamic models of operator workload to determine which operators to task in order to minimize mean task execution time. While the approach in²² involves planning for operators and UAVs at the systems-level, it is a *reactive* approach, in the sense that a human agent is assigned a task in real-time upon the arrival of that particular task. For the types of missions considered in this paper, the rules of engagement and mission time-lines are typically known ahead of time and that information can be exploited by adopting a *predictive* planning approach. For example, in the above mentioned scenario, once a UAV is tasked to survey a target it is expected that a human classification task will arrive once the UAV reaches its destination, and therefore a human classification task can be created with a time-window of validity which can be predicted in advance. Performing predictive rather than reactive planning over a team of operators enables the system to distribute the workload effectively among the agents thus reducing operator over-utilization and enhancing the overall mission performance. In addition, predictive planning also reduces disruptions and unpredictability in operator schedules, thus leading to improved operator situational awareness and reduced stress levels.

This work aims at designing a systems-level predictive task allocation algorithm that distributes tasks with time-windows of validity over a large heterogeneous team involving both human operators and robotic agents, while accounting for expected workload and availability of both types of agents over the scheduling window. This predictive systems-level approach enables both spatial and temporal coordination of agents in the heterogeneous team in order to effectively perform a set of tasks defined by deterministic and/or stochastic mission rules and expectations. Furthermore, it is crucial that this predictive planner be computationally efficient to enable real-time re-planning to handle dynamic environments and modeling uncertainties. In this research, the low-level detailed results and trends from previous work in HRI are leveraged to create human operator and robotic agent models which are then used in the large-scale scheduling optimization. The next sections describe the approach, the associated challenges and proposed solutions to this systems-level predictive task allocation problem. Results are presented showing that predictive systems-level planning improves mission performance, distributes workload efficiently among agents, and leads to coordinated agent behavior.

II. Coordinated Predictive Task Allocation

The goal of the systems-level predictive task planner is to simultaneously allocate a set of tasks among a team of heterogeneous agents. These tasks have different time-windows of validity within the scheduling window, may require coordinated execution between several agents, and may also involve coupling with other tasks.^{5,13} The agents have varying capabilities and constraints, and the network structure may be dynamic given agent mobility and communication constraints. Figure 1 depicts the overall system structure. The inputs to the task allocation planner are a set of tasks that have specifications on when they must be executed and on the team configuration required to complete them, a set of agent models for behavior prediction and the overall network configuration. Given the input list of tasks over some planning horizon, the planner allocates the tasks among the agents, thus creating schedules for each of the heterogeneous agents. This allocation is determined by taking into account the availability and capabilities of the agents over the course of the mission, with the object of minimizing task execution times and incurred costs, and maximizing mission efficiency. As the mission progresses, the mission control center manages the task list by creating new tasks or pruning completed and expired ones, and the agent models and network configuration are periodically updated. The task allocation algorithm runs in real-time and this replanning architecture allows the system to remain robust and flexible to dynamic environments and modeling uncertainties.

Within the task planning algorithm, the agent models are used to predict the respective agents' capabilities, specifically in regard to their expected task execution time and the expected cost incurred. The planner

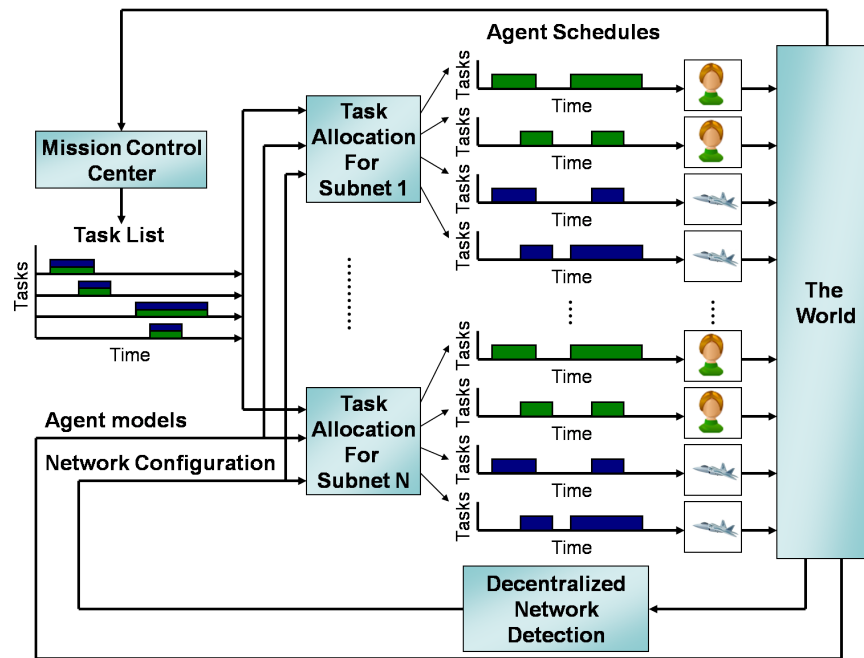


Figure 1. Real-time task allocation architecture for a heterogeneous team: Inputs to the task planner are tasks with time-windows of validity and coordination requirements, agent models and network configuration. Outputs of the planner are the particular schedules for each agent.

can use these metrics to determine the expected score an agent would obtain for doing a particular task, as well as for handling timing considerations such as ensuring that tasks are not assigned to busy agents. The robotic agents typically have well known expected service times and mission costs. The human agents, on the other hand, have complex nonlinear, time-varying, and stochastic capabilities, and their service rates and task performance are a function of workload, situational awareness, experience, and skill-level among other factors.^{12, 18, 20} For example, it is likely that an operator under stress will make more errors than when he is not stressed, or that an operator switching between different types of tasks will have poorer situational awareness than an operator performing the same type of task, and therefore his expected service time will be higher. The stochastic nature of this problem and the uncertainty produced by the inaccuracies in the agent models suggest that planning over an excessively long scheduling window is not useful, and that periodic revisions to the plan are required to accommodate the uncertainties in the system. It is important, therefore, to analyze and select an appropriate planning horizon length and an algorithm re-plan rate which optimize the trade-off between the benefits of predictive planning and the added computational complexity associated with executing such an algorithm.

II.A. Challenges

The coordinated predictive planning problem for a heterogeneous human-robot team presents several key challenges which are described in this section. The main issues involve: adequately modeling the human operator, dealing with the stochastic nature of this problem and the impact of uncertainty on task coordination, planning over a large, not fully connected network in a distributed fashion, and addressing the computational requirements resulting from increasing algorithm complexity.

MODELING HUMAN AGENTS Developing appropriate models of human agents is a difficult task. The models need to be sophisticated enough to account for the temporal variations in the operator dynamics (e.g., accuracy and speed of execution as a function of workload, situational awareness, experience, etc.), and differences in the capabilities between operators. These models are typically stochastic, nonlinear, and time-varying (see Cummings et. al.^{12, 18, 20}). However, models that are too complex cannot be effectively embedded into the task allocation framework. Therefore, the research challenge is to identify the key features of complex human behavior that are essential to executing typical operator tasks and describe those features in a mathematical form that can be used within the predictive planner.

STOCHASTIC NATURE OF COORDINATION REQUIREMENTS The synchronization of tasks by the planner is based on some prior knowledge of rules of engagement or mission operation specifications. However, these rules and specifications are not necessarily deterministic. For example, discovery of a certain target by a surveillance UAV would typically be followed by a target classification task for a human operator, however, if the UAV is late in performing its task, the human classification task time-window will also be impacted. Furthermore, the next set of tasks are dependent on the classification result. For example, the target could be a civilian, which requires no action, or it could be an enemy vehicle, which requires a follow-up defensive task. A probabilistic representation would be a natural choice to describe this stochastic aspect of mission specifications, as then the plans and schedules could be made based on some maximum-likelihood optimization and then updated as more information becomes available. The main challenges involve determining how to embed these probabilistic descriptions in the task allocation algorithm and how to update and replan without causing major disruptions to the agents' schedules.

COMPLEX COMBINATORIAL DECISION The problem of optimal task allocation over a network of agents is an NP-hard problem.⁸ Several approximation algorithms have been developed that address this problem complexity, balancing between solution sub-optimality and computational tractability. Previous work on multi-agent multi-task allocation includes many variants of the Traveling Salesman Problem (TSP) and the Dynamic Vehicle Routing Problem (DVRP), where different optimization techniques such as MILP solvers, genetic algorithms, and dynamic programming are used.^{13,16,25} The coordinated task allocation problem presented above has additional layers of complexity, involving predictive planning for tasks with future time-windows, coupling amongst agents (for example, operator team dynamics), and probabilistic dependencies between tasks, requiring both spatial coordination and temporal synchronization of the plans for the heterogeneous agents in the network. This additional layer of complexity expands the decision space, motivating the development of combinatorial approximation solutions that can maintain computational tractability while still ensuring algorithm convergence and solution feasibility.

NETWORK STRUCTURE AND DECENTRALIZED PLANNING Solutions to the coordinated task allocation problem typically fall into two main categories: centralized planning or decentralized approaches. For increasingly large networks, the decision space of the combinatorial problem quickly becomes intractable and a centralized planning approach is likely to be infeasible, promoting the use of decentralized algorithms (such as decentralized auction algorithms^{3,6,14,17,21}). The main challenges associated with decentralized planning involve ensuring efficient communication and consensus between agents in a network which may not be fully connected and dealing with different situational awareness between the agents.

II.B. Predictive Planning Algorithm Description

II.B.1. Consensus-Based Bundle Algorithm

Summarizing the aforementioned challenges, the planning problem is a stochastic combinatorial optimization problem for which the objective function is non-stationary and nonlinear. In addition, this work considers a large-scale network of agents, and therefore, it is essential for the solution technique to exhibit good scalability (such as being a polynomial-time algorithm).

Our approach to this complex combinatorial optimization planning problem is inspired by the Consensus-Based Bundle Algorithm (CBBA).⁹ CBBA is a distributed auction protocol that provides provably good approximate solutions for multi-agent multi-task allocation problems over networks of agents. CBBA consists of iterations between two phases: a bundle building phase where each vehicle greedily generates an ordered bundle of tasks, and a consensus phase where conflicting assignments are identified and resolved through local communication between neighboring agents. There are several core features of CBBA that can be exploited to develop an efficient planning mechanism for heterogeneous teams of human operators and autonomous agents. First, CBBA is a decentralized decision architecture. For a large team of autonomous agents, it would be too restrictive to assume the presence of a central planner (or server) with which every agent communicates. Instead, it is more natural for each agent to share information via local communication with its neighbors. This is also true for a team of human operators who typically use local communication interfaces (e.g., instant messenger) to share information. Second, CBBA is a polynomial-time algorithm. The worst-case complexity of the bundle construction is $\mathcal{O}(N_t L_t)$, where N_t and L_t are the number of tasks and the length of the bundle respectively, and CBBA converges within $\max\{N_t, L_t N_a\}D$ iterations, where D is the network diameter (always less than N_a). Thus, the CBBA framework scales well with the size of the network

and/or the number of tasks (or equivalently, the length of the planning horizon). Third, various design objectives, agent models, and constraints can be incorporated by defining appropriate scoring functions. If the resulting scoring scheme satisfies a certain property called *diminishing marginal gain* (DMG),⁹ a provably good feasible solution is guaranteed. Therefore, the problem of incorporating complicated models of human behavior boils down to defining appropriate score functions based on which bids for the human agents can be placed. The next sections describe the problem framework, the extensions to the CBBA planning algorithm to explicitly account for time-windows of validity for tasks, and the modeling process used for incorporating agent models within the planner framework.

II.B.2. Planning Problem Framework

This section describes the framework used to model the combinatorial decision planning problem. Given a list of N_t tasks and N_a agents, the goal of the task allocation algorithm is to find a conflict-free matching of tasks to agents that maximizes some global reward. An assignment is said to be free of conflicts if each task is assigned to no more than one agent. The global objective function is assumed to be a sum of local reward values, while each local reward is determined as a function of the tasks assigned to that particular agent.

The task assignment problem described above can be written as the following integer (possibly nonlinear) program:

$$\begin{aligned}
& \max && \sum_{i=1}^{N_a} \left(\sum_{j=1}^{N_t} c_{ij}(\tau_{ij}(\mathbf{p}_i(\mathbf{x}_i))) x_{ij} \right) \\
& \text{subject to:} && \sum_{j=1}^{N_t} x_{ij} \leq L_t, \quad \forall i \in \mathcal{I} \\
& && \sum_{i=1}^{N_a} x_{ij} \leq 1, \quad \forall j \in \mathcal{J} \\
& && x_{ij} \in \{0, 1\}, \quad \forall (i, j) \in \mathcal{I} \times \mathcal{J}
\end{aligned} \tag{1}$$

where the binary decision variable x_{ij} is 1 if agent i is assigned to task j , and $\mathbf{x}_i \in \{0, 1\}^{N_t}$ is a vector whose j -th element is x_{ij} . The index sets are defined as $\mathcal{I} \triangleq \{1, \dots, N_a\}$ and $\mathcal{J} \triangleq \{1, \dots, N_t\}$. The vector $\mathbf{p}_i \in (\mathcal{J} \cup \{\emptyset\})^{L_t}$ represents an ordered sequence of tasks for agent i ; its k -th element is $j \in \mathcal{J}$ if agent i conducts j at the k -th point along the path, and becomes \emptyset (denoting an empty task) at the k -th point if agent i conducts less than k tasks. L_t is a limit on the maximum amount of tasks that can be assigned to an agent. The summation term in brackets in the objective function represents the local reward for agent i .

Key assumptions underlying the above problem formulation are:

1. The score c_{ij} that agent i obtains by performing task j is defined as a function of the arrival time τ_{ij} at which the agent reaches the task (or possibly the expected arrival time in a probabilistic setting).
2. The arrival time τ_{ij} is *uniquely* defined as a function of the path \mathbf{p}_i that agent i takes.
3. The path \mathbf{p}_i is *uniquely* defined by the assignment vector of agent i , \mathbf{x}_i .

Many interesting design objectives for multi-agent decision making problems feature scoring functions that satisfy the above set of assumptions. The time-discounted value of targets^{4,7} is one such example, in which the sooner an agent arrives at the target, the higher the reward it obtains. However, in more complex missions scenarios, it may not be desirable to visit the target as soon as possible. For example, if the task is to re-investigate a previously observed target at some scheduled time in the future, a more reasonable choice of score function would have its maximum at the desired re-visiting time and lower values at re-visit times around the optimal time. This work develops methodologies to address these types of complicated scoring structures.

II.B.3. Scoring Functions with Time Windows

In order to execute predictive planning, it is necessary for the scoring function to reflect the temporal nature of the task, such that an agent receives a positive reward only if the task is started within its time-window of validity. To begin, this work defines the following entities:

Definition 1 1. *Score Profile ($s_j(t)$):* The score profile $s_j(t)$ represents the reward an agent gets from task j when it arrives at the task at time t , and is based on the value of the task, R_j , and any time penalty associated with the task. An example score profile is $s_j(t) = e^{-\lambda_j(t-t_{j_start})}R_j$, where $(t - t_{j_start})$ is the difference between the task start time and the agent arrival time, and $\lambda_j > 0$ is a discount parameter to penalize late arrivals. Without time discounting the score profile is $s_j(t) = R_j$.

2. *Time Window ($u_j(t)$):* The time window of validity for a task represents the time in which the task is allowed to be started. For task j this window is defined as

$$u_j(t) = \begin{cases} 1, & t_{j_start} \leq t \leq t_{j_end} \\ 0, & \text{otherwise.} \end{cases}$$

Using time windows for tasks provides a framework to penalize early arrivals as well as late arrivals.

In the CBBA planning algorithm,⁹ scoring profiles with time windows of validity can be incorporated as follows. The score an agent receives for a task is a function of his arrival time at the task location, τ_{ij} , and can be computed as $c_j(\tau_{ij}) = s_j(\tau_{ij})u_j(\tau_{ij})$. The arrival time, τ_{ij} , is in turn a function of the path the agent has taken before reaching task j . Given a path \mathbf{p}_i which is composed of tasks, and a corresponding set of best times $\tau_{ik}^*(\mathbf{p}_i)$ for all $k \in \mathbf{p}_i$, the bidding process can be described as follows. For each task $j \notin \mathbf{p}_i$, the best time to do task j can be found by solving the following problem,

$$\begin{aligned} \tau_{ij}^*(\mathbf{p}_i) &= \operatorname{argmax}_{\tau_{ij} \in [0, \infty)} c_j(\tau_{ij}(\mathbf{p}_i \oplus j)) \\ \text{subject to: } \tau_{ik}(\mathbf{p}_i \oplus j) &= \tau_{ik}^*(\mathbf{p}_i), \quad \forall k \in \mathbf{p}_i \end{aligned} \quad (2)$$

where \oplus signifies inserting task j into path \mathbf{p}_i without shuffling the order of tasks already in \mathbf{p}_i . The constraint states that the insertion of the new task j into path \mathbf{p}_i cannot impact the current arrival times for the tasks already in the path. The path is updated by inserting j in the best location, $\mathbf{p}_i \leftarrow (\mathbf{p}_i \oplus j)$. The best time and score for task j are then saved as $\tau_{ij}(\mathbf{p}_i) = \tau_{ij}^*$ and $c_{ij}(\tau_{ij}(\mathbf{p}_i)) = c_j(\tau_{ij}^*)$.

As mentioned before, an important property for convergence is the *diminishing marginal gain* property (DMG). In words, DMG means that the score for a task not in the path cannot increase as more tasks are added to the path, i.e., $\forall j \notin \mathbf{p}_i$

$$c_{ij}(\tau_{ij}(\mathbf{p}_i \oplus j)) \geq c_{ij}(\tau_{ij}(\mathbf{p}'_i \oplus j))$$

where $\mathbf{p}'_i = \{\mathbf{p}_i \oplus m\}$.

Consider the calculation of the best arrival time for task j when the current path is \mathbf{p}'_i instead of \mathbf{p}_i . Then, the following optimization needs to be solved:

$$\begin{aligned} \tau_{ij}^*(\mathbf{p}_i) &= \operatorname{argmax}_{\tau_{ij} \in [0, \infty)} c_j(\tau_{ij}(\mathbf{p}'_i \oplus j)) \\ \text{subject to: } \tau_{ik}(\mathbf{p}'_i \oplus j) &= \tau_{ik}^*(\mathbf{p}'_i), \quad \forall k \in \mathbf{p}'_i \end{aligned} \quad (3)$$

The constraint can be rewritten recursively as the following set of constraints,

$$\tau_{ik}(\mathbf{p}_i \oplus m \oplus j) = \tau_{ik}^*(\mathbf{p}_i \oplus m) = \tau_{ik}^*(\mathbf{p}_i), \quad \forall k \in \mathbf{p}_i \quad (4)$$

$$\tau_{im}(\mathbf{p}_i \oplus m \oplus j) = \tau_{im}^*(\mathbf{p}_i \oplus m). \quad (5)$$

Therefore, calculation of $\tau_{ij}^*(\mathbf{p}'_i)$ involves solving an optimization with the same objective function but an additional constraint (5). Thus, the optimal objective value for (3) cannot be greater than that for (2); i.e. $c_j(\tau_{ij}^*(\mathbf{p}_i)) \geq c_j(\tau_{ij}^*(\mathbf{p}'_i))$, which means the DMG property is satisfied. In other words, with the arrival time defined by the optimization in (2), the score function satisfies DMG regardless of the details of the score profiles.

II.B.4. Agent Modeling

In order to use CBBA as a planner for this heterogeneous task allocation problem it is necessary to incorporate scoring functions that accurately represent the agents. For the robotic agents, the score function should reflect the reward received for arriving at a task in a timely fashion and the cost incurred during travel. Within the CBBA framework, the scoring function used to represent the score a robotic agent i receives for doing task j is composed of two parts: a task reward with an exponential discount based on delay time and a fuel penalty due to travel distance,

$$c_j(\tau_{ij}^*) = e^{-\lambda_j(\tau_{ij}^* - t_{jstart})} R_j u_j(\tau_{ij}^*) - F_i \Delta D'_{ij}$$

where λ_j is a discount parameter associated with the task, R_j is the reward for executing task j , F_i is the cost of fuel per meter incurred by agent i and $\Delta D'_{ij}$ is a heuristic distance for every task, based on the distance from the vehicle's initial position to the particular task location. This heuristic $\Delta D'_{ij}$ is used in place of the actual travel distance (from one task to the next, $\Delta D_{ij}(\mathbf{p}_i)$) to ensure that the DMG property required for algorithm convergence is satisfied. The first term in the score function represents the time critical nature of tasks (such as target tracking tasks for which the likelihood of the target moving increases as longer delays progress) and the second term penalizes the agent for traveling to farther task locations.

As mentioned before, modeling human operators in the planning process is a complex task. For this work, time-varying server models are used for representing human operators, where the service rate varies as a function of operator workload and situational awareness. This type of model is described by Nehme¹⁸ and is presented in Figure 2. The curve describes a penalty due to loss of situational awareness incurred

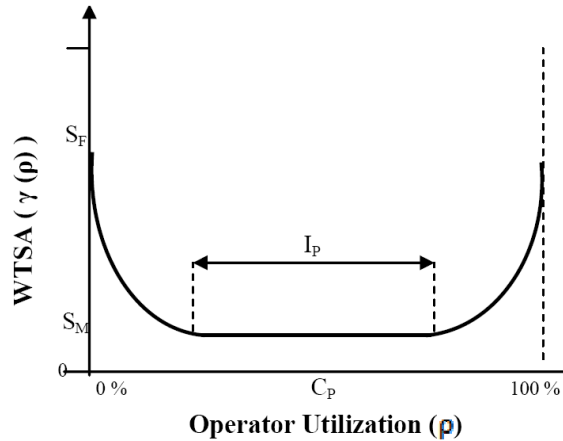


Figure 2. Wait-time due to loss of situational awareness as a function of operator utilization (diagram from Nehme,¹⁸ page 65).

by the operator (WTSA) as function of operator utilization. This curve is the inverse of the well-known Yerkes-Dodson law,²⁴ and indicates that due to low arousal levels and high stress levels, under-utilized and over-utilized operators respectively will incur longer delays due to lack of situational awareness. This model was incorporated in the CBBA planning framework as an exponential discounting penalty on the reward human operators receive for tasks. The score operator i receives for doing task j is defined by

$$c_j(\tau_{ij}^*) = e^{-\lambda_j(\tau_{ij}^* - t_{jstart} + \gamma(\rho_i))} R_j u_j(\tau_{ij}^*)$$

where λ_j is a discount parameter penalizing late arrivals and $\gamma(\rho_i)$ is the wait time due to loss of situational awareness (WTSA) as a function of the operator utilization level ρ_i (see Figure 2). In order to ensure algorithm convergence (satisfy DMG), the model used in CBBA focused on operator over-utilization and ignored the effects of operator under-utilization (such that the score function remained monotonic). This assumption was justified by the fact that in typical ISR missions involving small UAVs, the human operators usually remain busy enough to be over the 50% utilization level on average (performing several tasks such as target classification and continual vehicle status monitoring). One important limitation to consider when using human agent models is that the models are developed to represent an average operator and are not particularly well suited to represent any particular operator. It is beneficial, therefore, to update the

individual model parameters in real-time to represent the particular operators in the team. Work on adaptive modeling to address this issue is currently underway. In addition, development is also underway to extend the model to account for operator experience and interaction with other operators in the team.

III. Simulation Results and Discussion

To test the performance of the planning algorithm in a dynamic environment, a simulation of the system was implemented. The mission scenario involved a heterogeneous team of five UAVs and three human operators. The UAV tasks of interest consisted of surveillance and target tracking, where small UAVs equipped with imaging sensors would be sent to prescribed locations to collect imagery. These tasks were simulated by requiring the vehicle to travel to a particular location, arriving during a prescribed time-window, and to loiter at that location for a predefined task duration. For the human operators two types of tasks were considered. The first consisted of target classification and involved the operator analyzing imagery received from the UAVs. These tasks were simulated by requiring the human operator to be busy for a predefined task duration corresponding to the average task length of typical target classification tasks. The UAV surveillance tasks and the operator classification tasks were created with the same time-windows and with a high discount penalty on arriving late (high λ_j) to represent the time-critical nature of such tasks and to ensure coordination between the UAVs and the human operators. The second type of operator task involved vehicle status monitoring. These tasks were longer, occurred at a uniform rate, and were considered secondary and non time-critical (low λ_j value and low reward compared to the classification tasks). The planning algorithm computed the UAV and operator schedules over a prescribed planning horizon, and replanned periodically to account for uncertainties in operator state and to accommodate new tasks in the system. Figure 3 shows the agent schedules and operator utilization levels for a sample simulation run. The plot of agents schedules shows that the tasks for UAVs and operators were coordinated in time. It is interesting to note that this system does not require a UAV to be matched to the same operator every time, making this system structure more flexible than the typical system considered in HRI literature. The plot on the right, depicting the utilization level for the three operators, shows that the planner distributes the workload evenly across the operators and attempts to minimize the amount of time a particular operator spends above the 50% utilization threshold.

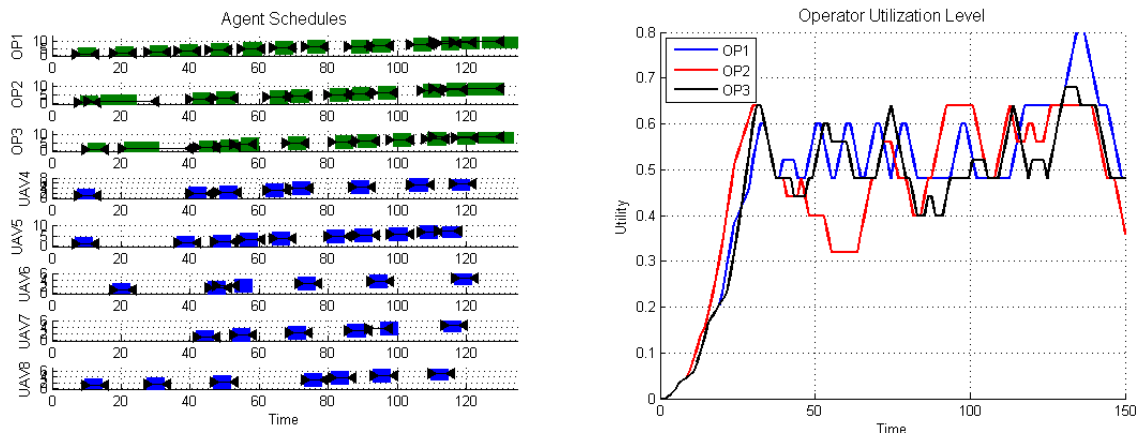


Figure 3. Agent schedules (left) and operator utilization levels (right) over the course of the simulation

The planning objectives for this system were to increase mission performance, increase operator efficiency (reduce task execution time and lower WTSA), and to reduce the amount of time that operators were over-utilized. By executing predictive planning over a planning horizon, as opposed to reactively planning as tasks arrive, the mission performance can be enhanced. Longer planning horizons allow agents to bid on future tasks that are more valuable, and planning agent schedules in advance helps coordinate the behavior of agents to ensure that tasks are reached in a timely fashion. The main tradeoff to executing predictive planning is that the computational complexity increases as the planning horizon increases, making real-time replanning difficult. It is therefore desirable to select a planning horizon that achieves good mission performance while maintaining computational tractability of the algorithm. In order to explore how mission performance was

affected by the length of the planning horizon, a Monte Carlo simulation was implemented and the results are displayed in Figure 4. The left plot shows the overall mission score as a function of planning horizon. The

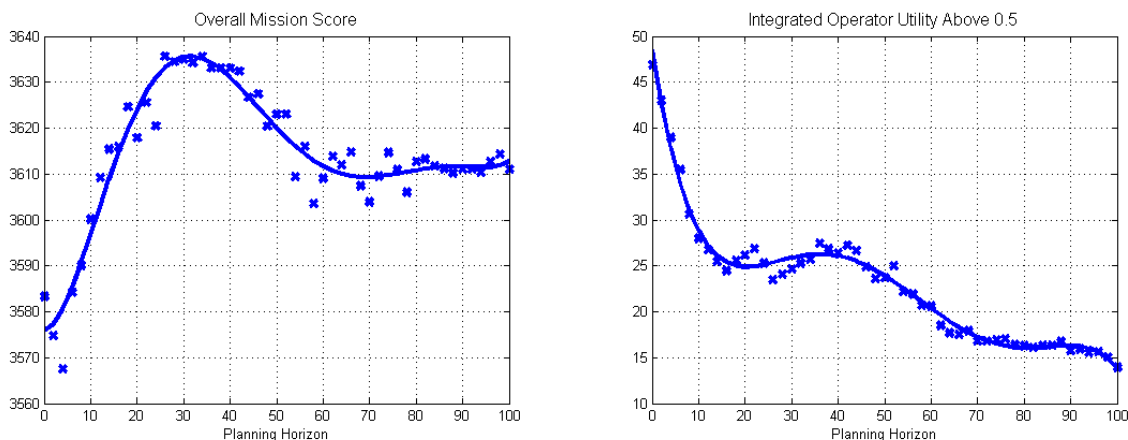


Figure 4. Comparison of mission scores and operator over-utilization as a function of planning horizon

plot on the right displays the integral of the operator utilization curve above the 50% utilization threshold accumulated over all the operators (represents the total operator over-utilization). As the planning horizon increases the mission performance is shown to increase up to a certain threshold. The decrease in mission performance for long planning horizons occurs because of the suboptimality in the planning algorithm caused by the sequential greedy assignment process, which is required to ensure polynomial-time convergence and computational tractability. This suboptimality becomes more severe as the ratio of the number of tasks to the number of agents increases. The mission score plot suggests that it is possible to optimize the planning horizon length to mitigate the effect of the planner suboptimality. The plot on the right shows that the amount of time the human operators spend above the optimal utilization level decreases as planning horizon increases. The results show that predictive planning improves mission performance and reduces operator over-utilization, and that the length of the planning horizon can be chosen to achieve good mission performance while maintaining computational tractability.

IV. Conclusions and Future Work

This paper presents a distributed systems-level predictive task allocation algorithm for a network of heterogeneous agents including human operators and robotic agents. This real-time planning architecture simultaneously allocates tasks with known time-windows of validity to all agents, while accounting for expected workload and availability of both types of agents over the planning horizon. The algorithm presented is an extension of the CBBA planning algorithm, a polynomial-time decentralized auction protocol that provides provably good approximate solutions for multi-agent multi-task allocation problems. This predictive systems-level approach enables both spatial and temporal coordination of agents in the heterogeneous team in order to effectively perform a set of tasks defined by deterministic or stochastic mission rules and expectations. Furthermore, this predictive planning architecture is computationally efficient and allows for real-time re-planning to handle dynamic environments and modeling uncertainties. The results show that predictive systems-level planning improves mission performance, distributes workload efficiently among agents, and leads to coordinated agent behavior.

There are several future extensions of this work. The first involves adaptive modeling within the planning architecture, where the performance of the agents can be recorded and used to tune the specific model parameters representing particular agents. It is also of interest to perform *integrated planning* over the system. This involves choosing the time-windows for the coordinated tasks such that the surveillance tasks assigned to the UAVs take into consideration the optimal times for the corresponding operator classification tasks in order to reduce operator utilization levels. Integrated planning would require the scoring functions of the agents to be coupled such that the bidding structure reflects the combined optimization problem. Further extensions of this work include incorporating error probabilities for the agents as an additional metric on mission success, as well as incorporating probabilistic dependencies between tasks (such that the

result of one task affects the existence of subsequent tasks). These extensions, which are currently under development, will result in a planning framework which increases system performance while maintaining robustness to dynamic environments and modeling uncertainties.

Acknowledgments

This research was supported in part by AFOSR (FA9550-08-1-0086) and MURI (FA9550-08-1-0356).

References

- ¹Air force doctrine document 2-5.2: Intelligence, surveillance, and reconnaissance operations. Technical report, United States Air Force, April 1999.
- ²Unmanned aircraft systems roadmap, 2005-2030. Technical report, Office of the Secretary of Defense, August 2005.
- ³A. Ahmed, A. Patel, T. Brown, M. Ham, M. Jang, and G. Agha. Task assignment for a physical agent team via a dynamic forward/reverse auction mechanism. In *International Conference on Integration of Knowledge Intensive Multi-Agent Systems*, 2005.
- ⁴M. Alighanbari and J. P. How. Decentralized task assignment for unmanned aerial vehicles. In *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference*, 2005.
- ⁵M. Alighanbari, Y. Kuwata, and J.P. How. Coordination and control of multiple uavs with timing constraints and loitering. In *American Control Conference, 2003. Proceedings of the 2003*, volume 6, pages 5311–5316 vol.6, June 2003.
- ⁶M. L. Atkinson. Results Analysis of Using Free Market Auctions to Distribute Control of UAVs. In *AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, 2004.
- ⁷J. Bellingham, M. Tillerson, A. Richards, and J. How. Multi-Task Allocation and Path Planning for Cooperating UAVs. In *Proceedings of Conference of Cooperative Control and Optimization*, Nov. 2001.
- ⁸D. Bertsimas and R. Weismantel. *Optimization over integers*. Dynamic Ideas Belmont, MA, 2005.
- ⁹H.-L. Choi, L. Brunet, and J. P. How. Consensus-based decentralized auctions for robust task allocation. *IEEE Trans. on Robotics*, 25 (4):912 – 926, 2009.
- ¹⁰J. W. Crandall and M. L. Cummings. Developing performance metrics for the supervisory control of multiple robots. In *Proceedings of the 2nd Annual Conference on Human-Robot Interaction*, Washington, D.C., 2007.
- ¹¹M. L. Cummings and P. Mitchell. Operator scheduling strategies in supervisory control of multiple UAVs. *Aerospace Science and Technology*, 11:339–348, 2007.
- ¹²M.L. Cummings and P.J. Mitchell. Predicting controller capacity in remote supervision of multiple unmanned vehicles. *IEEE Systems, Man, and Cybernetics, Part A Systems and Humans*, 38(2):451–460, 2008.
- ¹³R. Dondo and J. Cerdá. An MILP framework for dynamic vehicle routing problems with time windows. *Latin American Applied Research*, 36(4):255–261, 2006.
- ¹⁴B. Gerkey and M. Mataric. Sold!: Auction methods for multirobot coordination. *IEEE Transactions on Robotics and Automation*, 18(5):758–768, 2002.
- ¹⁵Benjamin Hardin and Michael A. Goodrich. On using mixed-initiative control: a perspective for managing large-scale robotic teams. In *HRI '09: Proceedings of the 4th ACM/IEEE international conference on Human robot interaction*, pages 165–172, New York, NY, USA, 2009. ACM.
- ¹⁶Q. Jun, J. Wang, and B. Zheng. A Hybrid Multi-objective Algorithm for Dynamic Vehicle Routing Problems. *Lecture Notes in Computer Science*, 5103:674–681, 2008.
- ¹⁷T. Lemaire, R. Alami, and S. Lacroix. A Distributed Task Allocation Scheme in Multi-UAV Context. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2004.
- ¹⁸C. Nehme. *Modeling Human Supervisory Control in Heterogeneous Unmanned Vehicle Systems*. PhD thesis, Massachusetts Institute of Technology, Dept. of Aeronautics and Astronautics, 2009.
- ¹⁹C. E. Nehme, B. Mekdeci, J. W. Crandall, and M. L. Cummings. The impact of heterogeneity on operator performance in future unmanned vehicle systems. *The International Command and Control Journal*, 2, 2008.
- ²⁰C.E. Nehme, R.M. Kilgore, and M.L. Cummings. Predicting the impact of heterogeneity on unmanned-vehicle team performance. In *Proceedings of HFES 2008: 52nd Annual Meeting of the Human Factors and Ergonomic Society*, New York, NY, USA, September 2008.
- ²¹S. Sariel and T. Balch. Real time auction based allocation of tasks for multi-robot exploration problem in dynamic environments. In *Proceedings of the AIAA Workshop on "Integrating Planning Into Scheduling"*, 2005.
- ²²K. Savla, T. Temple, and E. Frazzoli. Human-in-the-loop vehicle routing policies for dynamic environments. *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pages 1145–1150, Dec. 2008.
- ²³Huadong Wang, Michael Lewis, Prasanna Velagapudi, Paul Scerri, and Katia Sycara. How search and its subtasks scale in n robots. In *HRI '09: Proceedings of the 4th ACM/IEEE international conference on Human robot interaction*, pages 141–148, New York, NY, USA, 2009. ACM.
- ²⁴R. M. Yerkes and J. D. Dodson. The relation of strength and stimulus to rapidity of habit-formation. *Journal of Comparative Neurology and Psychology*, 1908.
- ²⁵K. Zhu. A new genetic algorithm for VRPTW. In *Proceedings of the International Conference on Artificial Intelligence*, page 311264. Citeseer, 2000.