REGULAR PAPER

Persistent Schedule Evaluation and Adaptive Re-planning for Maritime Search Tasks

Matthew J. Bays[1] · Thomas A. Wettergren[2] · Jaejeong Shin[3] · Shi Chang[4] · Silvia Ferrari[4]

Abstract

Search operations performed by adaptive autonomous maritime vehicles have been a topic of considerable interest for many years. Such operations require carefully scheduled coordination of multiple vehicles performing search tasks across the region of interest. Due to the inherent uncertainty of the maritime environment, however, an initially planned search schedule may not be maintained if the vehicles have significant capability to adapt their tasks to match the environment they detect in real time. We propose a multi-vehicle adaptive algorithm for dynamic evaluation and elastic re-planning of variable-length tasks commonly found in the maritime environments. In adaptive evaluation and re-planning problems, a set of tasks are initially planned for execution by adaptive, autonomous search vehicles. Tasks are allocated to search vehicles under a pre-defined schedule based on prior knowledge and desired outcome. Because of the vehicles' autonomy and reactivity to in situ conditions such as environment or target pose, the precise duration and actions required by each task are unknown a priori. We develop a hidden Markov model (HMM) for propagating task estimates, coupled with a quadratic-programming-based elastic re-scheduler. The result is an integrated estimate-and-schedule adaptation scheme that quickly and efficiently re-plans the vehicles' schedules based on in situ observations. The numerical simulation results show that this novel HMM approach decreases avoidable schedule variation by over a factor of two compared to existing methods.

Keywords Autonomous agents · Scheduling · Hidden markov model · Path planning

1 Introduction

Many planning and coordinated control methods have been recently developed for autonomous and adaptive unmanned maritime systems engaged in underwater search tasks. New methods for the search and inspection of underwater objects with varying degrees of reactivity to the environment have been proposed in [1–3]. In [1] the authors leverage information-theoretic techniques to develop bounds on the information potentially gained by a search by viewing the search problem as a communication channel. The authors then use the search channel to guide the mode switch for an individual agent from a broad search tactic to a focused search. When combined with significant advances in automated target recognition (ATR), see [4–6], these planning and control capabilities will soon allow autonomous maritime vehicles to perform a variety of complex, coordinated tasks that may vary both in expected duration and perfor-

mance over time. More recently, in [7], the authors develop a heuristic-based task allocation method for maritime patrols. This emerging need to coordinate unmanned maritime vehicles engaged in cooperative tasks is accompanied by closely-coupled constraints that necessitate minimizing changes to the overall schedule. Minimizing schedule changes reduces the risk of missing vehicle rendezvous, dramatic reconfiguration of vehicle tasking, and other performance degradations due to task changes. To date, very few methods have been developed for evaluating and re-planning the schedule of autonomous search vehicles in situ or for determining if and when higher-level re-allocation and re-scheduling is warranted. We propose a multi-vehicle adaptive algorithm for dynamic evaluation and elastic re-planning of variable-length tasks commonly found in the maritime environments. While the work herein are specifically applied to the maritime domain, the task estimation and adaptive re-planning algorithms are generalizable to other domains as well.

Prior work in vehicle scheduling and task allocation has largely involved planning based on pre-defined capabilities and estimated durations of all tasks [8, 9]. The uncertainty

✉ Matthew J. Bays
matthew.j.bays2.civ@us.navy.mil

Extended author information available on the last page of the article

in task duration has been previously handled using probabilistic models and formal risk metrics in [10]. Other studies have dealt with discrete schedule interruption events, such as vehicle inoperability or loss by minimizing deviations from the original plan [11].

However, existing approaches failed to develop scheduling and task re-allocation techniques able to persistently evaluate the performance of maritime vehicles in order to determine if and when re-allocation and re-scheduling should occur, or how to adapt current plans to gradual changes in performance. Sidoti et al. in [12] develop a method to perform multi-objective planning in a maritime domain using approximate dynamic programming with the specific application of vehicle routing to take into account weather conditions. They do so while incorporating uncertainty in environmental forecast with time and node-dependent cost scaling. The authors then extend the work to context-aware methods to assign multiple vehicles to multiple interdiction tasks in a spatio-temporal context [13]. De et al. develop a dynamic scheduling algorithm using mixed integer nonlinear programming to dynamically plan a maritime vehicle schedule across multiple time horizons [14]. Our new approach presented in this paper fills the need for holistic adaptive planning algorithms by creating a rigorous algorithmic approach to plan and re-plan task schedule and assignment for multiple vehicles with varying degrees of autonomy in the face of constantly changing conditions. We do so by leveraging a HMM for task prediction of duration mean and variance coupled with the concept of elasticity related to a variant of the flexible job-shop scheduling problem (FJSP).

Recently, there have been other metaheuristic approaches proposed for solving FJSP problems, including the tabu search algorithms [15], evolutionary approaches [16], and even using Lagrangian relaxation methods to reformulate the original MILP problem as a pure LP [17]. All of these approaches can be applied to the re-scheduling problem, but they effectively optimize the main problem again without directly solving it as an adaptation of the existing plan.

An extensive review of the literature regarding the related field multi-robot task allocation (MRTA) was recently developed by Chakraa et al. in [18]. The authors discuss at length a number of taxonomies of MRTA, including a detailed comparison of the various methods for task allocation. In the context of this review, our work is in regard to single-task - multi-robot - time-extended assignment (ST-MR-TA) with cross-schedule dependencies (XD). While the HMM is for a single-robot, our elastic scheduler takes into account dependencies between all vehicles within the network. We refer to Table 1 in Chakraa for a comparison between various works in the literature.

Recently, Dai et al. propose a number of heuristic algorithms to perform MRTA in [19], including an auction-based method, a vacancy-chain method, and a learning-based approach. Each approach has advantages and disadvantages. For example, the auction-based approach fixes the task allocation at each iteration, and thus may not be robust

Table 1 SATP Notation

Type	Symbol	Definition
Set	M	Set of maritime vehicles
	P	Set of tasks
	S	Set of service areas
	S_p	Set of subtasks for task p
	$\mathcal{K}$	Set of tasks, $k \in \mathcal{K} =$ {move, service, dock, deploy, wait}
Parameters	$T(y_s)$	Time/cost to execute task at subtask s.
	$I_{m,p}$	Task tuple such that $\mathbf{I}_{m,p} = \langle \kappa_{m,p}, \mathbf{L}_{m,p}, \mathbf{S}_{m,p}, \mathbf{E}_{m,p} \rangle$
	$\bar{\mathbf{S}}_{m,p}$	Planned start time for vehicle m executing phase p.
	$\bar{\mathbf{E}}_{m,p}$	Planned end time for vehicle m executing phase p.
Variables	A_s	A priori information for subtask s
	$\mathbf{S}_{m,p}$	Start time of phase p for vehicle m
	$\hat{\mathbf{S}}_{m,p}$	Expected start time of phase p for agent m
	$\mathbf{E}_{m,p}$	End time of phase p for agent a
	$\hat{\mathbf{E}}_{m,p}$	Expected end time of phase p for agent m
Random Variables	$Y_{m,p,s}$	Variable representing the sensed characteristic for a location sensed by vehicle m during subtask s of phase p.
	B_s	Random variable representing the actual environmental state under which the vehicle executes subtask s

to unexpected changes in the environment. The vacancy chain approach allows dynamic re-planning, but only for the limited case of a vacancy existing in the schedule. The learning-based approach mitigates these issues, but does not scale as number of robots increases. Our proposed algorithm, on the other hand, is linear with respect to robots for the HMM and allows each robot to independently compute their schedule estimate. The centralized scheduler we proposed is demonstrated to be solvable in polynomial time.

Lippi et al. develop a MILP formulation to schedule tasks, then during execution monitor for changes in parameters largely based on human variability on the team [20]. Based on the parameter change, they evaluate a modified cost for future tasks. If it deviates enough from the originally planned cost, then the execute a reallocation through re-running the MILP. Essentially their work formalizes a full replan step we tangentially discuss if the elastic scheduling fails due to constraint violation. However, our work has the additional benefit of modifying schedules without a computationally costly full re-plan.

The direct application of adaptive approaches that do not involve an adaptive solution to the original scheduling optimization problem have been limited. One recent approach is the PI-MaxAss algorithm [21], which adapts the scheduling of search-and-rescue vehicles by prioritizing the objective of maximizing the number of allocated tasks in a fixed amount of time over more direct objectives. Another recent example of re-scheduling is by Wang et al. [22], where they look at the impact of machine disruption on schedule adaption. They use a GA to solve the schedule adaptation problem and compare their results to two heuristics, a right-shifting scheduler which simply moves every task forward and a pre-scheduler that accounts for projected downtime. While the heuristics may have their advantages in speed, in many cases lack any bounding on the opimality gap. Our quadractic-programming scheduler allows optimal scheduling on polynomial time.

In our algorithm, we use a HMM to learn the delays that occur in the planned schedule. Hidden Markov models have historically been used in traditional machine learning (ML) applications for linguistics [23, 24], speech recognition [25, 26], and pose estimation [27, 28]. Schedule adaptation has been studied previously by Gabel for learning optimal policies to admit new jobs to the machine queue [29]. Buttazzo et al. first proposed the concept of using elasticity as a means to determine how to adjust schedules under changing conditions [30]. However, their work relied on the periodicity of tasks to determine the variation from one machine cycle to the next. We generalize these ideas to the scenario where tasks are not periodic, and instead use elasticity to determine a re-configuration of the timing of tasks in the job queue based on their expected HMM estimate.

Expected task end time completion statistics, the primary output of the HMM component of the presented work, has been studied previously using probabilistic and heuristic techniques. Li et al. developed an SME-based fuzzy network system for developing task completion times for a job-shop scheduling problem in [31]. While the methods provide a useful way of developing task completion time estimates with limited data, the method does not develop probabilistic estimates of the likelihood of those completion times. Barcelo et al. develop a Kalman filtering approach to travel time estimation in [32]. A related work by Hadachi et al. develop similar methods using particle-based techniques [33]. While their method provide a means to leverage collected data and a system model to create a forecasting system, the method does not provide a sufficiently generalizable method for task allocation including extraneous data. Recently, Ding et al. leverage a HMM to develop task allocation and sequencing for a set of smart machines on a factory floor in [34]. While in the same general field of task allocation as in our work, there are a few notable differences, including the focus on machine sequencing as opposed to schedule estimation and coordination. Additionally, they rely on a rather specific process flow for the sequencing compared to a generalized scheduling algorithm. To our knowledge, the proposed HMM-based approach is the first method to provide both a probabilistic technique and sufficiently generalizable method to be useful for general task allocation purposes.

We present an adaptive algorithm for dynamic evaluation of tasks within a queue that have been assigned to the unmanned maritime vehicles according to an initial schedule. A HMM is developed for estimating both the end time and relative uncertainty of the tasks as they are being carried out by the vehicles. Using real-time estimates of tasks' end times and uncertainty, the approach leverages concepts from elastic re-planning to solve a new version of the FJSP in which the start times of future tasks are modified to simultaneously maintain a valid schedule, minimize completion time, and reduce aggregate variation between the original schedule and modified schedule. The specific contributions of our work are:

- a HMM-based method to efficiently predict the expected time and duration variance of task execution that provides an improvement in flexibility and statistical refinement over other task completion time forecasting techniques.
- an FJSP-based elastic scheduling technique to maintain task precedence, while minimizing the aggregate start times between the pre-planned schedule and the executed schedule.
- numerical demonstration of the estimation and scheduling techniques with an adaptive low-level maritime survey planning algorithm in a simulation setting.

The paper is organized as follows. We formulate the estimation and scheduling problem in Section 2. We then discuss the details of the HMM in Section 3, which is then used to develop an estimate for the task end time variance. In particular, we extend the work on HMM task prediction to develop an estimate for task end time variance in Section 3.3. In Section 4, we develop the elastic re-scheduling procedure as a new form of an FJSP that is solved as an iterative quadratic programming problem. We discuss a case study and resulting application using the HMM and scheduler for an adaptive multiple aspect coverage algorithm in Section 5. Finally, we present simulation results of the entire scheduling approach demonstrated on a network of vehicles tasked with re-acquiring and identifying underwater targets of interest in Section 6.

2 Problem Formulation

Consider the problem of deploying M maritime vehicles in a region of interest to perform multiple coordinated tasks of varying type and duration. An example of this form of coordination under adaptation is shown in Fig. 1. In the figure, an unmanned surface vessel (USV) must deploy and sequentially collect unmanned underwater vehicles (UUVs) that perform adaptive survey tasks. At the same time, the UUVs adapt track spacing due to environmental characteristics. The adaptation of track spacing thereby increases the overall time to execute the individual search tasks, impacting the overall schedule. Each task has different characteristics as to their duration and variability of actual execution time when compared to planned execution schedule. We represent a task as belonging to vehicle $m \in \mathcal{M} = \{1, \ldots, M\}$, task type $\kappa \in \mathcal{K} = \{$dock, deploy, search, move$\}$ and indexed by task index $p \in \{1, \ldots, P_m\}$ using the tuple

Fig. 1 Illustration of maritime scenario involving adaptive autonomy combined with schedule constraints requiring coordination

$\mathbf{I}_{m,p} = \langle \kappa_{m,p}, \mathbf{L}_{m,p}, \mathbf{S}_{m,p}, \mathbf{E}_{m,p} \rangle$, representing the particular task type, task location set, task start time, and task end time, respectively.

Each search task is initially planned with a pre-defined start time $\mathbf{S}_{m,p} \in \mathbb{R}^+$ and end time $\mathbf{E}_{m,p} \in \mathbb{R}^+$, where $\mathbf{S}_{m,p} \leq \mathbf{E}_{m,p}$. We further assume that the executed start and end times denoted by $\bar{\mathbf{S}}_{m,p}$ and $\bar{\mathbf{E}}_{m,p}$ must have the same precedence as the planned start times. The goal of the adaptive re-planning and scheduling algorithm is to determine the optimal timing and task allocation such that the overall schedule change is minimized, while maintaining adequate makespan and schedule precedence constraints.

Each task may be divided into a finite number of subtasks such that

$$\mathbf{I}_{m,p}^{\text{search}} = \bigcup_{s=1}^{S_{m,p}} \mathbf{I}_{m,p,s}^{\text{search}}, \tag{1}$$

where $s = \{1, \ldots, S_p\}$ is an index of the subtasks of task $\mathbf{I}_{m,p}^{\text{search}}$. For search-related tasks, a subtask $\mathbf{I}_{m,p,s}^{\text{search}}$ may be executed to require one of a finite set of actions a vehicle may make based on the sensed environment. Let the environment be characterized by a finite set of exhaustive and mutually exclusive values denoted by $\mathcal{E} = \{e_1, \ldots, e_N\}$. These characteristics may be both physical and/or operational features that affect the execution of the task. In a maritime setting for bottom type, an example of $\mathcal{E}$ is $\mathcal{E} = \{$rocky, sandy, coral$\}$. It is assumed that each subtask is associated with only one type of environment. Let $Y_{m,p,s} \in \mathcal{E}$ be a random variable representing the sensed characteristic for a location sensed by vehicle m during subtask s of phase p. We represent the value a particular measurement of $Y_{m,p,s}$ as $y_{m,p,s}$. We note that we leverage the sensed state $Y_{m,p,s}$ as that for vehicle action as opposed to the hidden state. This is due to our particularly application for developing high-level schedule estimates rather than individual vehicle action decisions. We assume that there exists individual vehicle sensing and estimation algorithms onboard each vehicle, with their own unique method of determining state. Our interest is in the *aggregated* sensing and estimates, relative to overall baseline assumed state. We further note that the mathematical model for sensing and estimation of subtasks is generalizable to all vehicles m and search tasks p. Thus, we eliminate indices m and p in the references to sensing random variables $Y_{m,p,s}$ going forward when the vehicle m and phase p are clear. The executed end time $\bar{\mathbf{E}}_{m,p,s}$ of the search subtask is then dependent on both the executed start time $\bar{\mathbf{S}}_{m,p,s}$ and the time required to execute the task based on environment,

$$\bar{\mathbf{E}}_{m,p,s} = \bar{\mathbf{S}}_{m,p,s} + T(y_s). \tag{2}$$

Each subtask has a corresponding overall search task, indexed $\mathbf{I}_{m,p}^{\text{search}}$. The overall search task $\mathbf{I}_{m,p}^{\text{search}}$ has an end time
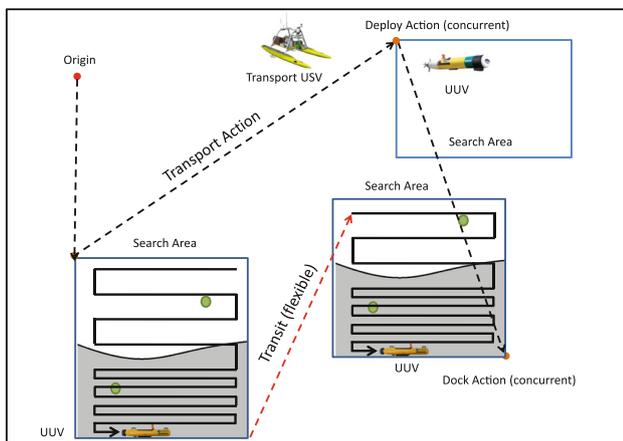
that varies depending on the particular sequence of measurements y_s, $s = 1, \ldots, S_p$ if the vehicles adapt their actions relative to the measurements. We now formulate a model for estimating and predicting the endtime for individual search tasks.

With these mathematical preliminaries, we now formulate the problem to solve:

Problem 2.1 Let there exist a set of tasks $\mathbf{I}_{1,1}, \ldots, \mathbf{I}_{M,P_m}$ for executing a multi-agent maritime search involving M agents, with multi-agent constraints as described in [35]. Let the task's planned start and end times $\{\mathbf{S}_{m,p}, \mathbf{E}_{m,p}\} \forall p \in P_m$ for each agent $m \in M$ differ from the as-executed end times dependent on the environmental variable $y_s \in Y_s$ for each subtask $s \in S_p$ such that $\mathbf{I}_{m,p} = \bigcup_{s=1}^{S_p} \mathbf{I}_{m,p,s}^{\text{search}}$. Minimize the aggregate difference between the planned $\{\mathbf{S}_{m,p}, \mathbf{E}_{m,p}\}$ and expected values of the executed start and end times $\{E(\bar{\mathbf{S}}_{m,p}), E(\bar{\mathbf{E}}_{m,p})\}$ by choice of flexible task start times

$$\bar{\mathbf{S}}_{m,p}$$

given task precedence constraints.

We refer to [35] for the full mathematical formulation of the required scheduling constraints for Problem 2.1, however we will briefly describe them herein. A set number of transport agents (unmanned surface vessels on our maritime problem) must transport a number of service agents (unmanned underwater vehicles) to service various areas, such as performing maritime surveys. A number of constraints must be satisfied such as only docking / deploying a single vehicle at a time, limiting number of vehicles carried at a given time due to capacity, and every area must be serviced. This leads to an NP-complete scheduling problem in which accurate estimation of completion time for each task is crucial to maintaining the validity and effeciency of the schedule.

3 A Hidden Markov Model for Uncertain Task Duration

Scheduling multiple coordinated tasks adaptively requires formal estimates of the mean $E\left(\bar{\mathbf{E}}_{m,p}\right) = \hat{\mathbf{E}}_{m,p}$ and variance $VAR\left(\bar{\mathbf{E}}_{m,p}\right)$ of the executed end time for each search task. We now develop an approach for calculating $\hat{\mathbf{E}}_{m,p}$ and $VAR\left(\bar{\mathbf{E}}_{m,p}\right)$ using a HMM. HMMs are dynamic Bayesian networks that typically leverage time-series data to perform state prediction [36]. In our work, we take the unique approach of developing an HMM in *task space* as opposed to *time space*. That is, we use a HMM to iteratively assess the task times for search tasks of each vehicle with respect to the sequence of subtasks. In this manner, the dynamic nature of the Bayesian network is not with respect to time, but with

respect to the subtasks $\mathbf{I}_{m,p,s}^{\text{search}}$ sequenced by the subtask series $s = 1, \ldots, S_p$. The approach allows us to develop a subtask series estimate of the expected end time and relative uncertainty for the search tasks within the overall schedule.

3.1 End Time Expectation for Varying Subtasks

To develop an expected value of $\bar{\mathbf{E}}_{m,p}$, the formal definition of expectation of the end time, in Eq. 2, is taken with respect to the environmental measurement Y_s to obtain

$$
\begin{aligned}
E\left(\bar{\mathbf{E}}_{m,p}\right) &= E\left(\bar{\mathbf{S}}_{m,p} + \sum_{s=1}^{S_p} T(Y_s)\right) \\
&= \bar{\mathbf{S}}_{m,p} + \sum_{s=1}^{S_p} E\left(T(Y_s)\right) \\
&= \hat{\mathbf{E}}_{m,p}
\end{aligned}
\tag{3}
$$

thanks to the linearity of the expectation operator. In practice, $\hat{\mathbf{E}}_{m,p}$ is conditioned on the predicted environment, as well as the previously sensed environment of the system. Although the formal conditioning is used within the development of the posterior in the derivation of the expected end time calculation, the conditioning is removed from the notation for clarity of the exposition. Now a running estimation of the value of $\hat{\mathbf{E}}_{m,p}$ while executing subtask $\mathbf{I}_{m,p,s}^{\text{search}}$, denoted $\hat{\mathbf{E}}_{m,p}(s)$, is explained by exploiting the fact that the particular mode executed at each individual subtask indexed from $s' = 1, \ldots, s - 1$ is separable from the expectation. Thus, the running mean expected endtime takes the form

$$
\begin{aligned}
\hat{\mathbf{E}}_{m,p}(s) = \bar{\mathbf{S}}_{m,p} &+ \sum_{s'=1}^{s-1} T(y_{s'}) \\
&+ \sum_{s''=s}^{S_p} \sum_{y_{s''} \in Y_{s''}} \left(T(y_{s''}) \right. \\
&\left. \times P(Y_{s''} = y_{s''} | Y_{1:s-1}, A_{1:S_p})\right),
\end{aligned}
\tag{4}
$$

where $P(Y_{s''} = y_{s''} | Y_{1:s-1}, E_{1:S_p})$ is the probability of task indexed s'' having a particular sensed environment, $Y_{1:s-1}$ is the measurement history of the vehicle, and $A_{1:S_p}$ is the a priori information on the environmental conditions. The term $A_s \in \mathcal{E}$ represents the initial environmental information which may be known from previous subsea surveys, expert knowledge, or other means to create an initial estimate of the likely environment. We now turn to the structure of the posterior probability in the development of the HMM.

3.2 Posterior Probabilities of Task Environment

We develop the posterior probability for each detection type, $P(Y_s|Y_{1:s-1}, A_{1:s})$, by first creating a model under which the detection events occur from the standpoint of the autonomous vehicle executing the subtasks. Let $B_s \in \mathcal{E}$ be a random variable representing the actual environmental state under which the vehicle executes subtask $\mathbf{I}_{m,p,s}^{\text{search}}$. Then the sensed value y_s is directly dependent on B_s. We assume the probability distribution of B_s is propagated using two factors: the previous environmental state B_{s-1}, and the prior knowledge of the overall environment for the given subtask $\mathbf{I}_{m,p,s}^{\text{search}}$, represented by A_s.

The dependencies can be represented by a Bayes network learned from data, exemplified in Fig. 2, where each in situ environmental state B_s depends on the previous in situ environmental state B_s, as well as the a priori knowledge A_s. The Bayes network in Fig. 2 contains several characteristics that are intuitive from the standpoint of standard detection theory, while additionally containing useful extensions from a vehicle schedule estimation standpoint. Firstly, sensing events are not directly dependent on each other. Additionally, the a priori random variable A_s provides a mechanism to inject beliefs about the environment from previous surveys or expert knowledge. We choose this model under the intuition that upon detecting an event at one location, we are likely to continue detecting that event at the immediate location in the future. However, the weight of the previous detection events will revert to the prior the further in the future the estimate is propagated.

From the HMM, individual priors and conditional probability density functions (CPDs) may be realized. Let $P(B_s|A_s, B_{s-1})$ be the joint conditional probability distribution based on both A_s and the previous true state B_{s-1}, and $\mathbf{A}$ be the state transition matrix form of $P(B_s|A_s, B_{s-1})$. Let $P(Y_s|B_s)$ be the probability distribution of sensed environment conditioned on the in situ environmental state. From the CPDs, the conditional probability of the next in situ environmental state observed can be obtained recursively as

$$P(Y_s|Y_{1:s-1}, A_{1:s}) = \mathbf{M}\sigma(B_s)$$

where $\sigma_s(B_s)$ is the prior belief state

$$\sigma_s(B_s) = \mathbf{A}\sigma_{s-1}(B_{s-1}). \tag{5}$$

and $\mathbf{M}$ is the state observation matrix. Within the state observation matrix, the ij^{th} position of $\mathbf{M}$ represents $P(Y_s = j|B_s = i)$. Using Eq. 5, the value of $\hat{\mathbf{E}}_{m,p}(s)$ in Eq. 4 is given by

$$\hat{\mathbf{E}}_{m,p}(s) = \bar{\mathbf{S}}_{m,p} + \sum_{s'=1}^{s} T(y_{s'}) + \sum_{s''=s+1}^{S_p} \hat{T}_{s''}, \tag{6}$$

where

$$\hat{T}_{s''} = \left(\mathbf{T}^{\text{search}}\right)^{\top} \mathbf{M}_{s''}\sigma(B_{s''}), \tag{7}$$

and $\mathbf{T}^{\text{search}}$ is the vector form of $T(y_{s''})$.

3.3 Variance Calculation for Final Endtime

We now turn to calculating $VAR(\bar{\mathbf{E}}_{m,p})$, which is also inherently conditioned on $Y_{1:s}$ and $E_{1:S_p}$. Since the terms $\bar{\mathbf{S}}_{m,p} + \sum_{s'=1}^{s} T(y_{s'})$ within $\hat{\mathbf{E}}_{m,p}$ are constant, they do not change the variance. As shown in [37], the variance of the summation is given as

$$VAR(\bar{\mathbf{E}}_{m,p}(s)) =$$
$$\sum_{s'=s+1}^{S_p} VAR(T(Y_{s'}))$$
$$+ 2 \sum_{s+1 \le s' < s''}^{S_p} COV(T(Y_{s'}), T(Y_{s''})). \tag{8}$$

From the definition of covariance, we have

$$COV(T(Y_{s'}), T(Y_{s''})) =$$
$$E(T(Y_{s'})T(Y_{s''})) - \hat{T}(Y_{s'})\hat{T}(Y_{s''}). \tag{9}$$

Fig. 2 HMM for subtask prediction

This allows the computation of the joint expectation $E\left(T(Y_{s'})T(Y_{s'})\right)$ for any two measurement variables $Y_{s'}$ and $Y_{s''}$. Specifically, the joint expectation is given by Eq. 10. Combining Eqs. 7, 9, and 10 with the properties of transient Markov chains found in [38], the covariance is then

$$
\begin{aligned}
E\left(T(Y_{s'})T(Y_{s''})\right) &= \sum_{\substack{y_{s'}\in Y_{s'} \\ y_{s''}\in Y_{s''}}} T(y_{s'})T(y_{s''}) \\
&\qquad P\left(Y_{s'}=y_{s'}, Y_{s''}=y_{s''}|A_{1:s}, Y_{1:s}\right) \\
&= \sum_{\substack{y_{s'}\in Y_{s'} \\ y_{s''}\in Y_{s''}}} T(y_{s'})T(y_{s''}) \sum_{B_{s'},B_{s''}} \\
&\qquad P\left(y_{s'}, y_{s''}|B_{s'}, B_{s''}, A_{1:s}, Y_{1:s}\right) \\
&\qquad P(B_{s'}, B_{s''}|A_{1:s}, Y_{1:s}) \\
&= \sum_{\substack{y_{s'}\in Y_{s'} \\ y_{s''}\in Y_{s''}}} T(y_{s'})T(y_{s''}) \sum_{B_{s'},B_{s''}} \\
&\qquad [P\left(y_{s'}|B_{s'}, A_{1:s}, Y_{1:s}\right)P\left(y_{s''}|B_{s''}, A_{1:s}, Y_{1:s}\right) \\
&\qquad \times P(B_{s'}, B_{s''}|A_{1:s}, Y_{1:s})] \\
&= \sum_{\substack{y_{s'}\in Y_{s'} \\ y_{s''}\in Y_{s''}}} T(y_{s'})T(y_{s''}) \sum_{B_{s'},B_{s''}} \\
&\qquad P\left(y_{s'}|B_{s'}\right)P\left(y_{s''}|B_{s''}\right) \\
&\qquad P(B_{s'}, B_{s''}|A_{1:s}, Y_{1:s}).
\end{aligned} \tag{10}
$$

$$
\begin{aligned}
COV\left(T(Y_{s'}), T(Y_{s''})\right) &= \tag{11} \\
\sum_{\substack{y_{s'}\in Y_{s'} \\ y_{s''}\in Y_{s''}}} \sum_{\substack{B_{s'} \\ B_{s''}}} & T(y_{s'})T(y_{s''})P\left(Y_{s'}=y_{s'}, Y_{s''}=y_{s''}|A_{1:s}, Y_{1:s}\right) \times \\
& \left[\sum_{\substack{B_{s'} \\ B_{s''}}} \gamma_{b_s'}^{s'} \gamma_{b_s''}^{s''-s'}\left[\sigma(B_{s'})^{\mathsf{T}}\mathbf{H}\mathbf{E}_{b_{s'}}\mathbf{H}^{-1}\right]_{b_s'}\left[\mathbf{H}\mathbf{E}_{b_{s''}}\mathbf{H}^{-1}\right]_{b_{s'}b_{s''}}\right] \\
- \sum_{\substack{y_{s'}\in Y_{s'} \\ y_{s''}\in Y_{s''}}} \sum_{\substack{B_{s'} \\ B_{s''}}} & T(y_{s'})T(y_{s''})P\left(Y_{s'}=y_{s'}, Y_{s''}=y_{s''}|A_{1:s}, Y_{1:s}\right) \times \\
& \left[\sum_{\substack{B_{s'} \\ B_{s''}}} \sigma(B_{s'})^{\mathsf{T}}\gamma_{b_s'}^{s'}\left[\mathbf{H}\mathbf{E}_{b_{s'}}\mathbf{H}^{-1}\right]_{b_s'}\sigma(B_{s''})^{\mathsf{T}}\gamma_{b_s''}^{s''}\left[\mathbf{H}\mathbf{E}_{b_{s''}}\mathbf{H}^{-1}\right]_{b_{s''}}\right],
\end{aligned}
$$

where $\gamma_{b_s'}^{s'}$ is a vector of eigenvalues of the matrix form for $P\left(B_{s'}|B_{s'-1}, A_{s'}\right)$, $\mathbf{H}$ is a matrix of the corresponding eigenvectors, and $\mathbf{E}_{b_{s'}}$ is a matrix with 1 in the $b_{s'}, b_{s'}^{th}$ place, and 0 elsewhere.

Finally, for $VAR\left(T(Y_{s'})\right)$ the variance is obtained using the standard definition, as

$$
VAR\left(T(Y_{s'})\right) = \tag{12}
$$

$$
E\left(\left(\bar{\mathbf{E}}_{m,p}(s')\right)^2\right) - \left(\hat{\mathbf{E}}_{m,p}(s')\right)^2
$$

$$
= \sum_{y_{s'}\in\mathcal{Y}_s'} \left(T(Y_{s'})\sum_{B_{s'}}P(Y_{s'}|B_{s'})\sigma(B_{s'})\right)^2 - \left(\hat{T}_{s'}\right)^2 \tag{13}
$$

In Fig. 3, we show three examples of the hidden Markov model plotting $\hat{\mathbf{E}}_{m,p}(s)$ versus task index s for a search task. We compare $\hat{\mathbf{E}}_{m,p}(s)$ to the end time calculated by simply updating the original planned end time $\mathbf{E}_{m,p}(s)$ sequentially with how long the task took at execution. The first row contains the running estimate $\hat{\mathbf{E}}_{m,p}(s)$ vs subtask index s. Blue represents the updated end time based on tasks executed and tasks that remain as initially planned. Red represents the running estimate $\mathbf{E}_{m,p}(s)$. Green represents the end time of a perfectly-predicted search effort based on ground-truth of B_s and resulting vehicle actions. As seen in the figure, the predicted end time is closer to the true search time at all points within the task execution. In the second row of Fig. 3, we show how the variance in the expected end time based on the HMM changes as the subtasks are prosecuted. The final row contains an illustration representing the actual task characteristic with which the estimator develops a real-time prediction of end time as the search proceeds through the environment. In the first two instances, the environment is favorable to easy tasks, and then changes to moderate (left-most experiment) or difficult (middle experiment). The right-most experiment has the environment randomly change to match the HMM.

4 Iterative Elastic Scheduling of Flexible Tasks

Using the online estimates given in Eqs. 6 and 12, we now turn to developing a framework to perform continuous re-planning of the schedule based on the expected end time and uncertainty of search tasks. To do so, we first develop an ontology to describe the relative interaction *between* tasks. Note that the task interaction ontology for an adaptive schedule is different from tasks types, as the interaction depends on the types as well as the relative adjacency of the tasks. For example, while flexible tasks may have different executed start and end times than initially planned, two flexible tasks immediately adjacent to each other must have a one-to-one correspondence in schedule delays of the first task.

(a)

(b)

(c)

(d)

(e)

(f)

(g)

(h)

(i)
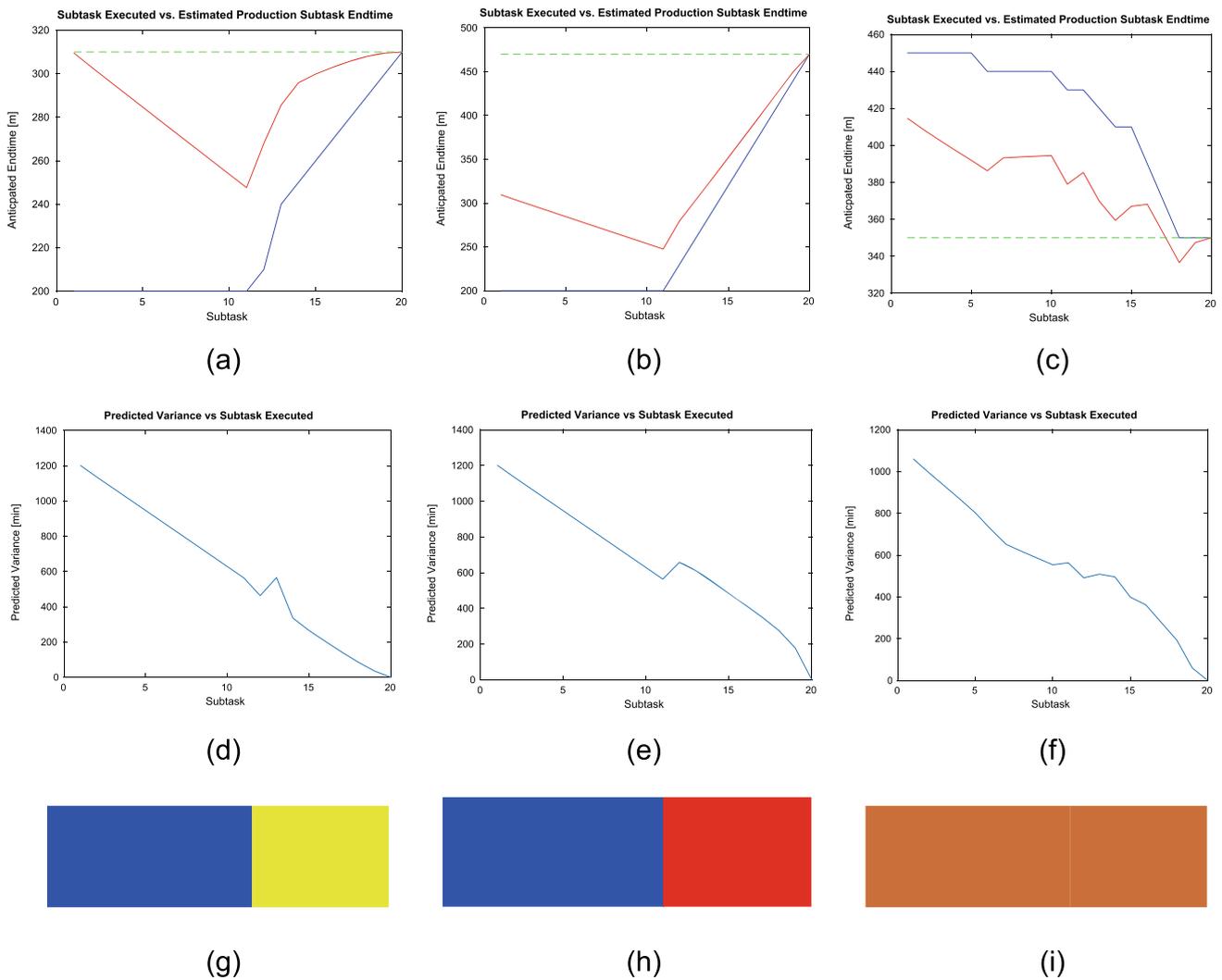
Fig. 3 (Top Row) Comparison of the prediction algorithm end time to an iteratively updated plan for a task's end time. Blue represents the updated end time based on tasks executed and tasks that remain as initially planned. Red represents the end time based on tasks executed and forward-prediction based on the belief states. (Middle Row) Variance of predicted end time based on hidden Markov model. (Bottom Row) Illustration of actual environment throughout search time proceeding left-to-right. Blue represents easy subtasks. Yellow represents moderate subtasks type. Red represents difficult. Brown represents the event when the difficulty distribution matches distribution created by state transition matrix of the HMM

Conversely, two flexible tasks that have a gap in between execution may be shifted independently.

We first define *cross-scheduled* tasks as tasks that have direct effects on different vehicles. For example, a docking task is scheduled for both a search vehicle executing search tasks, and a transport vehicle in a search vehicle - transport vehicle system. Formally, cross-scheduled tasks are defined as follows:

Definition 1 Tasks $\mathbf{I}_{m,p}$ and $\mathbf{I}_{m',p'}$ s.t. $m \neq m'$, are *cross-scheduled* ($\mathbf{I}_{m,p}, \mathbf{I}_{m',p'} \in \mathcal{I}^{\text{cross}}$) if there exists implicit

constraints that $\mathbf{S}_{m,p} = \mathbf{S}_{m',p'}$ and $\mathbf{E}_{m,p} = \mathbf{E}_{m',p'}$ within the scheduling problem.

We define *coupled* tasks as tasks whose corresponding relative start times and precedence must always remain the same within the schedule. Formally, we have

$$\bar{\mathbf{S}}_{m,p} = \mathbf{S}_{m,p} \ \forall m \in \mathcal{M} \ \text{s.t.} \ \mathbf{I}_{m,p} \in \mathcal{I}^{fixed}. \tag{14}$$

Finally, we define tasks $\mathbf{I}_{m,p}$ and $\mathbf{I}_{m',p'}$ as *adjacent* if there exists implicit constraints that $\mathbf{S}_{m,p} = \mathbf{E}_{m',p'}$.

4.1 Elasticity Parameter Calculation using Schedule Variance

We now tie the hidden Markov model developed in Section 3 to the re-planning of the linked tasks in our variant of the FSJP. We do so using an elasticity parameter for use in the cost function to minimize. The goal of the minimization is to weight the perturbation of future tasks in an optimal manner to minimize the overall difference between the new schedule and the original schedule while respecting schedule constraints. This is accomplished using the following cost function:

Minimize

$$\sum_{m \in \mathcal{M}, p \in \mathcal{P}_m} k_{m,p} \left| \left(\mathbf{S}_{m,p+1} - \mathbf{E}_{m,p} \right) - \left(\bar{\mathbf{S}}_{m,p+1} - \bar{\mathbf{E}}_{m,p} \right) \right|^2 \quad (15)$$

where

$$k_{m,p} = \sum_{p=1}^{P_m} VAR \left(\bar{\mathbf{E}}_{m,p}(s) \right), \quad (16)$$

$VAR \left(\bar{\mathbf{E}}_{m,p}(s) \right)$ is calculated in Eq. 12, and $\bar{\mathbf{S}}_{m,p}$ represent the variables of optimization.

The cost function Eq. 15 is inspired by elasticity-based path planning work originally developed by Shah [39], and represents the elastic energy of the overall schedule. The key difference is that Eq. 15 minimizes the weighted gaps between the start of one task p and the beginning of the next task $p+1$ for all vehicles $m \in \mathcal{M}$, while in [39], the authors seek to minimize the difference in elasticity between maps and sensed landmarks.

4.2 A Quadratic-programming Approach to Elastic Scheduling

We now present a quadratic programming approach to combining a number of search tasks into an optimally-delayed end-to-end schedule.

Problem 4.1 The elastic scheduling problem is written as follows

Minimize

$$\sum_{m \in \mathcal{M}, p \in \mathcal{P}_m} k_{m,p} \left| \left(\mathbf{S}_{m,p+1} - \mathbf{E}_{m,p} \right) \right.$$
$$\left. - \left(\bar{\mathbf{S}}_{m,p+1} - \bar{\mathbf{E}}_{m,p} \right) \right|^2 \quad (17a)$$

Subject to

$$\forall a \in \mathcal{M}_s, \, p \in \mathcal{P}$$
$$k_{m,p} = \sum_{p=1}^{P_m} VAR \left(\bar{\mathbf{E}}_{m,p}(s) \right) \quad (17b)$$

$$\forall a \in \mathcal{M}_s, \, p \in \mathcal{P}$$
$$\bar{\mathbf{S}}_{m,p} \leq \bar{\mathbf{S}}_{m,p+1} \quad (17c)$$

$$\forall a \in \mathcal{M}_s, \, p \in \mathcal{P} \text{ s.t.}$$
$$\mathbf{I}_{m,p} \in \mathcal{I}^{fixed} \Rightarrow \bar{\mathbf{S}}_{m,p} = \mathbf{S}_{m,p} \quad (17d)$$

$$\forall a \in \mathcal{M}_s, \, p \in \mathcal{P} \text{ s.t.}$$
$$\langle \mathbf{I}_{m,p}, \mathbf{I}_{m',p'} \rangle \in \mathcal{I}^{cross} \Rightarrow$$
$$\bar{\mathbf{S}}_{m,p} = \mathbf{S}_{m',p'} \quad (17e)$$

$$\forall a \in \mathcal{M}_s, \, p \in \mathcal{P} \text{ s.t.}$$
$$\bar{\mathbf{E}}_{m,p} = \mathbf{S}_{m,p} + \hat{T}(Y_{s''}) \quad (17f)$$

The constraints Eqs. 17a and 17b define the cost function and elasticity weights, respectively. Constraint Eq. 17c enforces precedence between task schedules. Constraint Eq. 17d enforces that fixed tasks are scheduled at their original start time. The constraint Eq. 17e enforces that if two tasks $\mathbf{I}_{m,p}$ and $\mathbf{I}_{m',p'}$ are cross-scheduled, their start times must occur concurrently. Finally, Eq. 17f enforces the end time of search tasks occur at the current expected search time based on the current timestep's estimate from Eq. 6. The above QP is implemented in software within the IBM CPLEX optimization suite. Due to the specific construction of the objective function and constraints, Problem Eq. 4.1 can be solved in polynomial time. The QP constructed in Problem Eq. 4.1 can be solved in polynomial time due to 1) the lack of binary variables, and 2) the quadratic component of

$$\begin{aligned} C = & |k_{m_n, P_n} \left(\bar{\mathbf{S}}_{M, P_M} \right)^2 - 2k_{m, P_m} \left(\bar{\mathbf{S}}_{m, P_m} \right) \left(\bar{\mathbf{E}}_{m_n, P_n - 1} \right) \\ & + k_{m, P_m} \left(\bar{\mathbf{E}}_{m, P_m - 1} \right)^2 + \ldots + \\ & x k_{1,1} \left(\bar{\mathbf{S}}_{1,2} \right)^2 - 2k_{1,1} \left(\bar{\mathbf{S}}_{1,2} \right) \left(\bar{\mathbf{E}}_{1,1} \right) + k_{1,1} \left(\bar{\mathbf{E}}_{1,1} \right)^2 + \\ & - 2k_{m, P_m} \left(\mathbf{S}_{m, P_m} - \mathbf{E}_{m, P_m - 1} \right) \bar{\mathbf{S}}_{m, P_m} - \ldots - 2k_{1,1} \left(\mathbf{S}_{1,2} - \mathbf{E}_{1,1} \right) \bar{\mathbf{S}}_{1,2} \\ & + 2k_{m, P_m} \left(\mathbf{S}_{m, P_m} - \mathbf{E}_{m_n, P_n - 1} \right) \bar{\mathbf{E}}_{m_n, P_n - 1} + \ldots + 2k_{1,1} \left(\mathbf{S}_{1,2} - \mathbf{E}_{1,1} \right) \bar{\mathbf{E}}_{1,1}| \end{aligned} \quad (18)$$

the cost function Eq. 17a is positive semi-definite when in matrix form. This quadratic property is formally proven in the following proposition.

Proposition 4.2 *The elastic scheduling quadratic program found in Problem* Eq. *4.1 is solvable in polynomial time.*

Without loss of generality, let there exist M vehicles and P_m tasks for each vehicle m. Expanding Eq. 17a, let C be the terms containing an optimization variable $(\bar{\mathbf{S}}_{m,p}, \bar{\mathbf{E}}_{m,p})$. Thus, we have the cost function $C(\cdot)$ found in Eq. 18. Rewrite C in quadratic form as

$$
\mathbf{C} = \begin{bmatrix} \bar{\mathbf{T}}_{m,P_m} \\ \vdots \\ \bar{\mathbf{T}}_{1,1} \end{bmatrix}^\top \begin{bmatrix} \mathbf{K_{m,P_m}} & & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{K_{1,1}} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{T}}_{m,P_m} \\ \vdots \\ \bar{\mathbf{T}}_{1,1} \end{bmatrix}
$$

$$
+ \begin{bmatrix} -2k_{m,P_n}\left(\mathbf{S}_{m,P_n} - \mathbf{E}_{m,P_n-1}\right) \\ 2k_{m,P_n}\left(\mathbf{S}_{m,P_n} - \mathbf{E}_{m,P_n-1}\right) \\ \vdots \\ -2k_{1,1}\left(\mathbf{S}_{1,2} - \mathbf{E}_{1,1}\right) \\ 2k_{1,1}\left(\mathbf{S}_{1,2} - \mathbf{E}_{1,1}\right) \end{bmatrix}^\top \begin{bmatrix} \bar{\mathbf{T}}_{m,Pa} \\ \vdots \\ \bar{\mathbf{T}}_{1,1} \end{bmatrix}
$$

$$
= \bar{\mathbf{T}}^\top \mathbf{Q}\bar{\mathbf{T}} + \mathbf{c}^\top \bar{\mathbf{T}} \tag{19}
$$

for all $\mathbf{K_{m,p}}$ and $\bar{\mathbf{T}}_{m,p}$, $p = 1, \ldots, P_m$ and $m = 1, \ldots, M$, where

$$
\mathbf{K_{m,p}} = \begin{bmatrix} k_{m,P_m} & -k_{m,P_m} \\ -k_{m,P_m} & k_{m,P_m} \end{bmatrix} \tag{20}
$$

and

$$
\bar{\mathbf{T}}_{m,p} = \begin{bmatrix} \bar{\mathbf{S}}_{m,p+1} \\ \bar{\mathbf{E}}_{m,p} \end{bmatrix}. \tag{21}
$$

Clearly, $\mathbf{Q}$ is a block-diagonal matrix, and thus the eigenvalues of $\mathbf{Q}$ are the eigenvalues of those of the block matrices $\mathbf{K}_{m,p}\ \forall m \in \mathcal{M}, p \in \mathcal{P}_m$. Computing the eigenvalues of $\mathbf{K}_{m,p}$, we have

$$
\lambda_{\mathbf{K}_{m,p}} = \{2k_{m,p}, 0\}. \tag{22}
$$

Due to constraint Eq. 17b, $k_{m,p} \geq 0\ \forall m \in \mathcal{M},\ p \in \mathcal{P}_m$, and thus $\mathbf{Q}$ is positive-semidefinite. Thus, Eq. 4.1 can be solved in polynomial time [40]. □

A full iterative elastic scheduler is shown in Algorithm 1. The algorithm combines the polynomial-time solvable quadratic program found in Problem Eq. 4.1 with the HMM for task completion time estimation. In the algorithm, line 5 queries the subtasks for each vehicle. Line 6 obtains the task

completion estimate for the subtasks from the vehicles returning subtasks. Line 7 executes the elastic scheduling quadratic program. Upon adjusting the schedule, the vehicles execute current tasking within line 8. Finally, line 9 checks if the elastic schedule was found to be infeasible, and if so, triggers a full replan.

Algorithm 1 Full task allocation and sequencing algorithm.

1: **procedure** ELASTICSCHEDULING($\mathcal{M}, \mathbf{I}$)
2:     **while** $\exists \mathbf{I}^{active} \in \mathbf{I}$ **do**
3:         **for** $m \in \mathcal{M}$ **do**
4:             **for** $\mathbf{I}_{m,p} \in \mathbf{I}_m^{service}$ **do**
5:                 $\mathcal{S}_m \leftarrow$ GetSubtasks$(\mathbf{I}_{m,p})$
6:                 $\hat{\mathbf{E}}_{m,p}, VAR(\bar{\mathbf{E}}_{m,p}) \leftarrow$ vehicleHMM$\left(\bar{\mathbf{I}}_{m,p}^{service}, Y_{m,p,s}, S_m\right)$
7:                 $\hat{\mathbf{S}}, IsFeasible \leftarrow$ ElasticQPSolve$\left(\mathbf{I}, \mathcal{M}, \hat{\mathbf{E}}_{m,p}, VAR(\bar{\mathbf{E}}_{m,p})\right)$
8:                 $Y_{m,p,s} \leftarrow$ ExecuteSubtasks$(\mathcal{M}, \mathcal{S})$
9:         **if** $!IsFeasible$ **then return** $ReplanRequired$
            return $TasksComplete$

---

5 A Flexible Scheduling Case Study for Adaptive Multiple Aspect Coverage

We now describe an implimentation case study for our adaptive elastic scheduling techniques for inspecting a row of discrete targets that must be identified, a common problem for maritime search and survey. In this application, the inspection of targets occurs over a set of discrete locations s that correspond to observation windows of the specific sub-areas of the domain, each one of which is associated to one of the discrete targets that must be identified. The primary goal of the scheduler is to determine the appropriate scheduling of the visitations of these sub-areas in order to maximize target identification opportunities over a limited amount of time. The ability of the system to identify a given target in location s is improved as multiple observations are obtained at that location. However, as a practical matter it is important to limit the number of observations to the minimal required to observe the target object with the required confidence. Thus, we use the HMM framework to adaptively improve the estimate of the required number of observations at each successive location and thus improve the schedule performance.

The adaptive multiple aspect coverage (AMAC) algorithm is an existing specific planning algorithm that plans these numbers of observations of each target. We utilitze the AMAC algorithm to provide a prior schedule for this case study; and also use the state transition matrix formed in this

planning for on-line schedule adaptation. An illustration of the general maritime mission using the AMAC algorithm is found in the Top of Fig. 4.

Suppose there are a line of N targets in the region of interest that are to be investigated sequentially. Let A_s represent the number of observations that are made of the target in the sub-area s in order to achieve the desired performance (i.e. identification confidence). The required number of observations will depend upon local features which may be inferred from examination of the target in the prior sub-area $(s-1)$ as well some state characteristics of the target, which we represent as G_s. The state G_s in this example can represent either the target type or target pose. Thus, both A_s and G_s are components of the previously defined multi-dimensional state E_s associated with the particular location s.

It is clear that A_s is a random variable in the range $\mathcal{A} = \{1, \ldots, A\}$ where A is the maximum number of observations that can be collected for each target location. Let a_s represent a specific realization of the random variable A_s. Furthermore, G_s is a random variable that represents the value of the state component of the target in location s, and let a specific realization of G_s be given by g_s. Now the sequence of observations $\{A_1, \ldots, A_N\}$ across all N targets can be viewed as a Markov chain with transition probability matrix $\mathbf{P}$, where the i, j element of $\mathbf{P}$ is given by

$$p_{ij} = P(A_s = j | A_{s-1} = i), \qquad i, j \in \{1, \ldots, A\} \tag{23}$$

Upon completion of the visitation of all N locations, a sample path of the Markov chain is realized. The log-likelihood of this sample path is given by

$$\log P(a_1, ..., a_N; \mathbf{P}) = \tag{24}$$
$$\log[P(a_N | a_{N-1}; \mathbf{P}) P(a_{N-1} | a_{N-2}; \mathbf{P})$$
$$\times \ldots \times P(a_2 | a_1; \mathbf{P}) P(a_1)]$$
$$= \sum_{s=2}^{N} \log P(a_s | a_{s-1}; \mathbf{P}) + \log P(a_1)$$
$$= \sum_{s=2}^{N} \sum_{i=1}^{A} \sum_{j=1}^{A} I(a_{s-1} = i, a_s = j) \log p_{ij} + \log P(a_1)$$
$$= \sum_{i=1}^{A} \sum_{j=1}^{A} J_{ij} \log p_{ij} + \log P(a_1)$$

where J_{ij} is the number of transitions from state i to state j in the sample path, and $P(a_1)$ is the initial probability of the Markov chain. The maximum likelihood estimate (MLE) of the transition matrix $\mathbf{P}$ can now be found by setting

$$\frac{d}{dp_{ij}} \log P(a_1, \ldots, a_N; \mathbf{P}) = 0 \tag{25}$$

to arrive at the estimate

$$\hat{p}_{ij} = \frac{J_{ij}}{\sum_{j=1}^{A} J_{ij}}. \tag{26}$$

In other words, the MLE of P_{ij} is equal to the number of transitions from state $A_s = i$ to state $A_s = j$ divided by the total number of appearances of state $A_s = i$ in the sample path. In a similar manner, the maximum likelihood estimate of the transition probabilities $P(A_s | A_{s-1}, G_s)$ can be computed as

$$\hat{P}(A_s = j | A_{s-1} = i, G_s = g) = \tag{27}$$
$$\frac{\sum_{s=2}^{N} I(a_{s-1} = i, a_s = j, g_s = g)}{\sum_{s=2}^{N} \sum_{j=1}^{A} I(a_{s-1} = i, a_s = j, g_s = g)}$$

where g is a specific target state such as target type or target pose.

To determine the observations for each target that is to be identified, we apply the AMAC algorithm. As autonomous target recognition performance varies depending on the viewing angle of an imaging sensor, the AMAC algorithm adapts the vehicle's path such that the next viewing angle has the highest expected information gain on target classification. The AMAC algorithm computes the most informative aspect angle based on the expected confidence level [41]. The expected confidence level is an estimate of confidence level on target classification given that the next observation is obtained at a given aspect angle.

The expected confidence level is computed based on a probabilistic sensor model, which represents the relationship between aspect angle, sensor measurements, and target features. After the next best aspect angle is computed, the AMAC algorithm plans a swath at the chosen aspect angle such that the probability of detection becomes higher than a user-chosen threshold. The target position uncertainty is represented by a 2-dimensional Gaussian distribution, and sensor field-of-view is represented by the probability of a target to be detected by the sensor as a function of distance from the sensor to a target. Considering these two conditions, the location and length of the swath is computed to obtain a satisfactory probability of detection. This swath planning method is applied from the multiple aspect coverage (MAC) algorithm, whose theoretical and experimental results are presented in [42].

As the vehicle follows the planned swath, the sensor can obtain measurements. Based on this observation, the AMAC algorithm updates the target classification confidence level. Since we are considering an application where the ground truth is unknown, the belief on target classification is modeled using a confidence level such that the vehicle is required to obtain enough observations and achieve a desired confidence level. Thus, the AMAC algorithm decides whether the

Fig. 4 (a) Illustrative cartoon of vehicle observation counts dependent on target state. (b) Example of a segment of path generated by the AMAC algorithm for five targets, where the confidence level threshold is chosen as 0.97

(a)

(b)
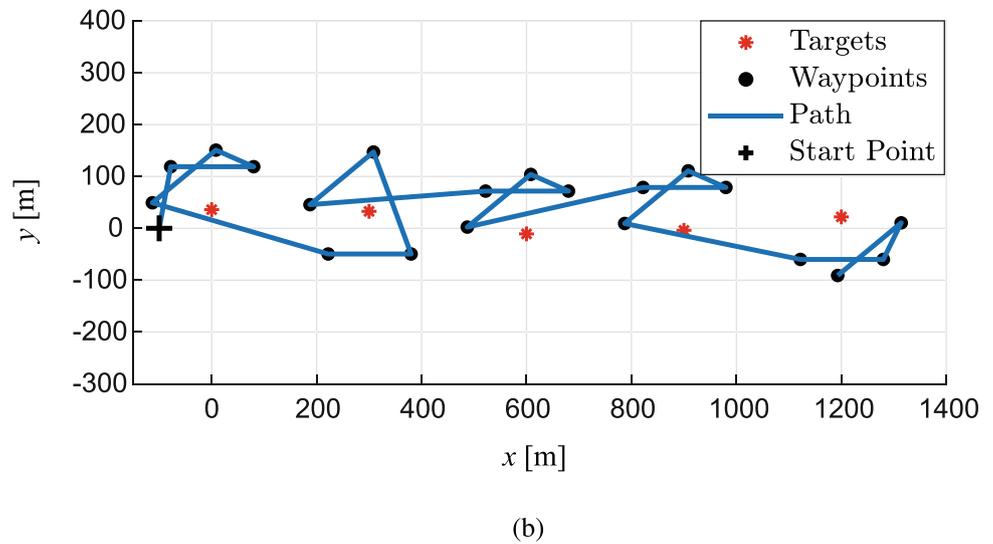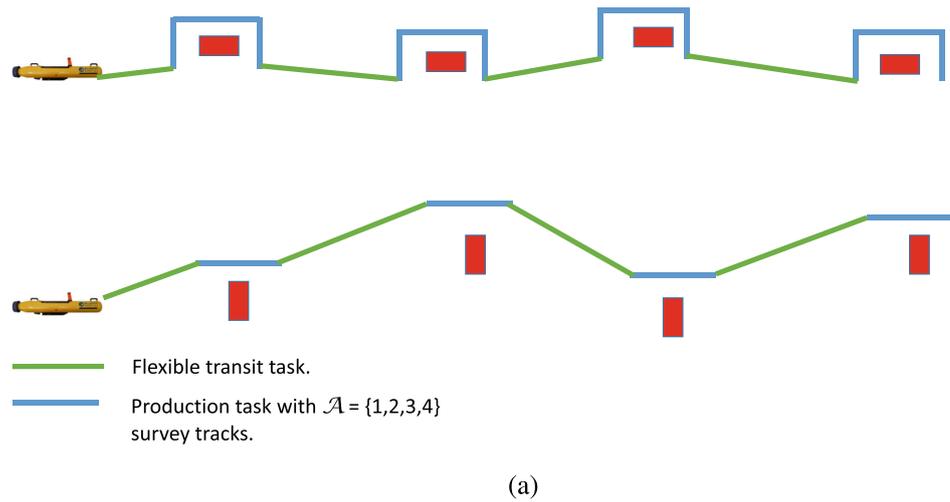
sensor will obtain more observations or move on to the next target based on the updated classification confidence level. By adapting the aspect angles and the number of observations, the AMAC algorithm can optimize the travel time while guaranteeing that a desired confidence level will be achieved.

As an example of the AMAC algorithm and the associated transition matrix, we ran a simulated scenario with $N = 500$ targets that are to be identified. In this scenario, the distance between adjacent targets are set at 300 meters. The given range of the sonar imaging sensor is 300 meters with 30 meters of dead-zone at the center of the field-of-view, and the desired confidence level is set as 0.97. Each target required a set of $a_s \in \{2, 3, 4\}$ observations of the target to achieve the required level of identification accuracy (corresponding to the 0.97 confidence level).

A visual display of a segment of the associated vehicle trajectory to achieve these passes is shown in Fig. 4. Of the $N = 500$ targets in this scenario, $N_2 = 393$ required two

Table 2 The empirical probabilities of requiring A_s observations for target s, given A_{s-1} observations are obtained for previous target $s - 1$ and the true target classification Y_s, that is, $P(A_s|A_{s-1}, Y_s)$

Target true classification	$G_s = 0$			$G_s = 1$		
Previous contact	$A_{s-1} = 2$	$A_{s-1} = 3$	$A_{s-1} = 4$	$A_{s-1} = 2$	$A_{s-1} = 3$	$A_{s-1} = 4$
$A_s = 2$	0.7987	0.3538	0.4091	0.9310	1	0.7333
$A_s = 3$	0.1447	0.6462	0	0.0216	0	0
$A_s = 4$	0.0566	0	0.5909	0.0474	0	0.2667

Table 3 Average distance and time cost vs observations

Number of observations per target	2	3	4
Average path length spent on each target (m)	428.35	718.33	972.58
Average path time cost spent on each target (min)	4.7594	7.9814	10.8064

observations, $N_3 = 70$ required three observations, and the remaining $N_4 = 37$ required four observations. This led to an empirically derived transition probability matrix **P** (as in Eq. 23) of the form (written in compact form) found in Table 2. A counterpart to the probabilities are the costs associated with each type of action, found in Table 3. Both represent the components of the HMM found in Section 3 in our case study for maritime search.

6 Simulation Results

The effectiveness of the elastic scheduling algorithm is demonstrated on an autonomous maritime search task scheduling problem. In the first study, the scheduler is applied to search tasks involving the surveying of different environments. In the second study, the scheduler is applied to a group of vehicles performing re-identification of multiple observed targets.

Fig. 5 Simulation of elastic schedule at multiple time increments of one transport vehicle and four search vehicles. (a) Initial schedule. (b) Schedule showing accumulated delays at t=1300 min. (c) Schedule showing accumulated delays at t=2600 min
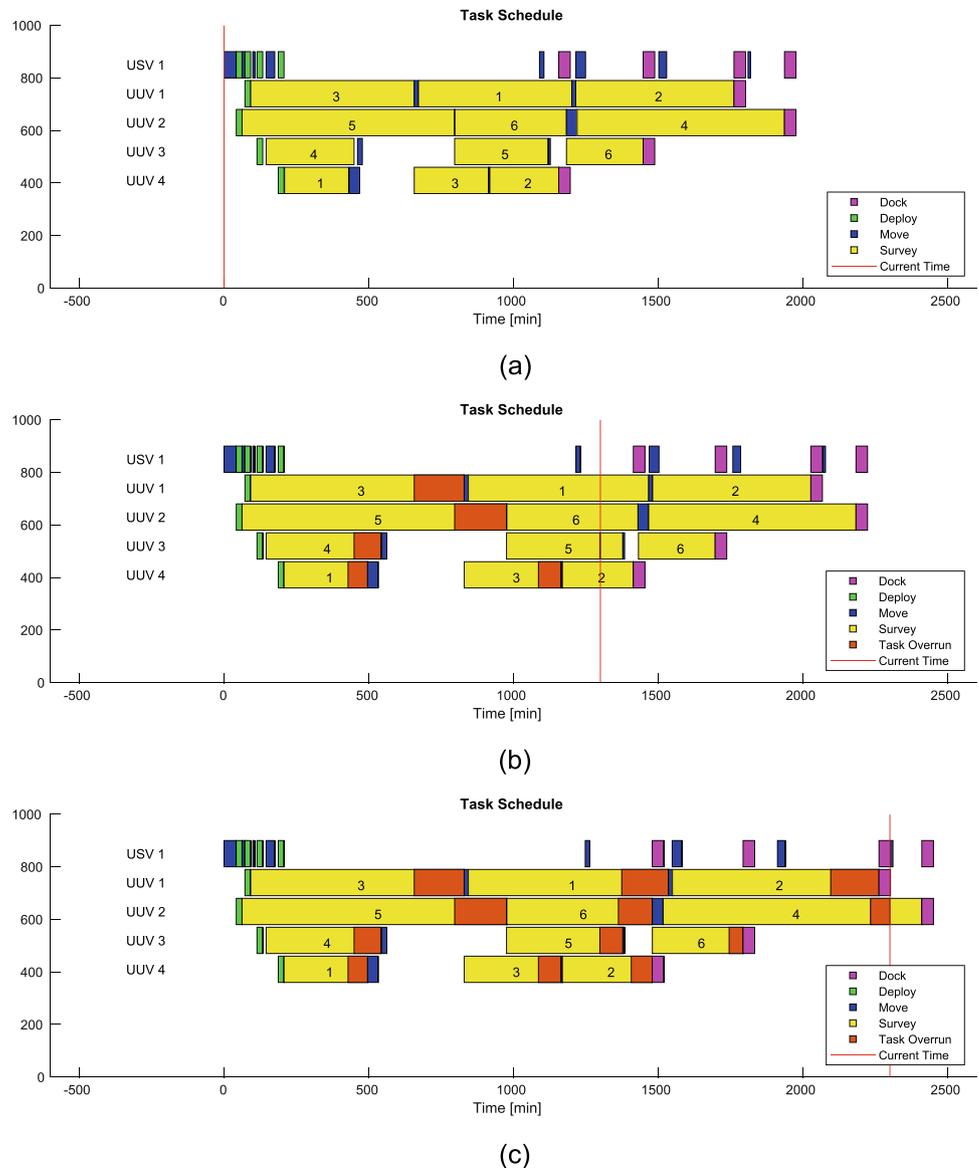
6.1 Adaptive Scheduling of Environmental Surveys

We first present a simulation of $M_t = 1$ transport vehicle and $M_s = 4$ search vehicles (USV and UUVs, respectively), for a combined total of $M = 5$ vehicles to prosecute $S = 12$ search tasks. Each search task must be visited by one of the search vehicles. The optimized schedules were pre-solved using the partially-decoupled scheduling algorithms found in [43]. In the simulations, each search task consists of $S = 10$ subareas. In each task, we assume three possible states of a search subtask of easy, moderate, and difficult. The initial belief $E_{1:S_p}$ for each task p is that the subtask will require $T = 49.16$ minutes for easy subtasks, $T = 58.99$ minutes for moderate subtasks, and $T = 98.32$ minutes for difficult subtasks. Initially, the expected state E_{s_p} for every subtask in each search task is that the subtask is easy. However, the ground-truth is that the second half of each search task consists of difficult subtasks. Interspersed between search tasks are either movement by the UUVs between nearby search areas, or docking with the USV, a transportation action, and deployment using the USV. Naturally, because of the complex nature of docking and deployment of UUVs, it is desireable to minimize the changes in schedule around tasks such as these. Our elastic scheduling algorithm successfully minimizes a formal metric for change in the objective function found in Eq. 15. In Fig. 5, we show an example of the evolution of the elastic task schedule as the schedule evolves over time. Finally, we show the minimum objective function value evolution over time in Fig. 6.
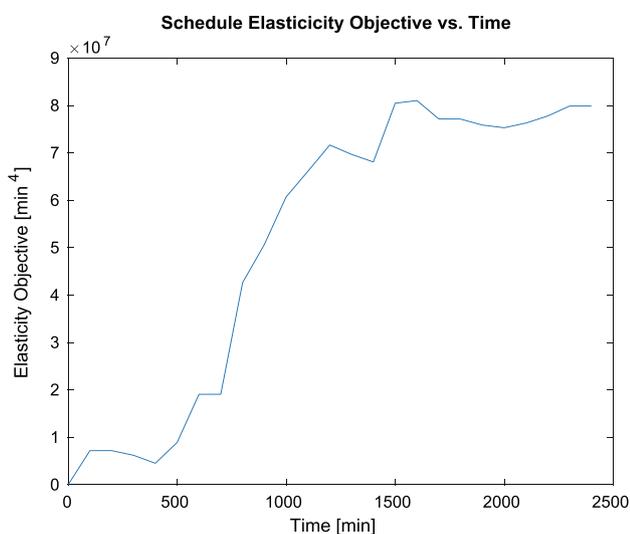
Fig. 6 Minimized objective of elastic scheduling problem vs. time increment

6.2 Adaptive Scheduling of Multiple Aspect Target Coverage

In our second set of simulations, we show a related experiment involving an adaptive multiple aspect coverage algorithm for identifying underwater targets of interest. The state transition matrix from the HMM was trained as described in Section 5. Here $M - t = 1$ transport vehicle and $M_s = 3$ identification vehicles (USV and UUVs, respectively) were simulated to identify the targets of interest, and then perform actions based on the number of passes required. We refer to this particular vehicle operation as a reacquire and identify (RI) task. An example simulation of the end-to-end vehicle schedule is shown in Fig. 7.

We next present the performance of both the HMM and elastic scheduler when compared to to a greedy scheduling algorithm combined with an incremental update of the expected duration in Table 4. The particular greedy algorithm we leverage is inspired by [44]. The comparison is made using a Monte Carlo simulation of 40 runs of randomized environments using $M_s = 3$ UUVs and $M_t = 1$ USV, with varying ground-truth state data. In the Monte Carlo simulation, we analyze the expected variation in expected duration of the tasks at the beginning, middle, and end of each set of RI tasks. When we compare the projection for each set of RI tasks midway through the set to the end, we find that the HMM coupled with elastic scheduler projects the end time within 0.71% of the final end time. The greedy schedule coupled with an incremental update to the expected final time results in an estimate within only 7.45% of the final end time of the task. Thus, the HMM - Elastic scheduler results in a 10-fold increase in the accuracy of estimated endtime for search tasks. For an iteration of the optimization sequence, the HMM requires 0.0032 seconds to develop the expected end time for all active tasks. The uncertainty model required an average of 0.28 seconds to develop the variance used in the costs. The elastic scheduler takes less than one-hundredth of a second to develop the optimal elastic solution. The overall runtime for an iteration of the scheduler and HMM combined was 0.693 seconds. The memory usage per the MATLAB Profiler was 32.132 megabytes. We believe that this is sufficiently fast for real-time operation for a task-oriented estimation scheme, so long as the tasks are sufficiently large. We also note that these performance numbers are before any code optimization for use on embedded systems.

Finally, we show the objective function evolution over time of the HMM - elastic scheduler when compared to the iterative update and greedy scheduling algorithm in Fig. 8. As seen in the figure, the HMM and elastic scheduling maintains

Fig. 7 Simulation of elastic schedule at multiple time increments of one transport vehicle and three search vehicles. (a) Initial schedule. (b) Schedule showing accumulated delays at t=160 min. (c) Schedule showing accumulated delays at t=340 min
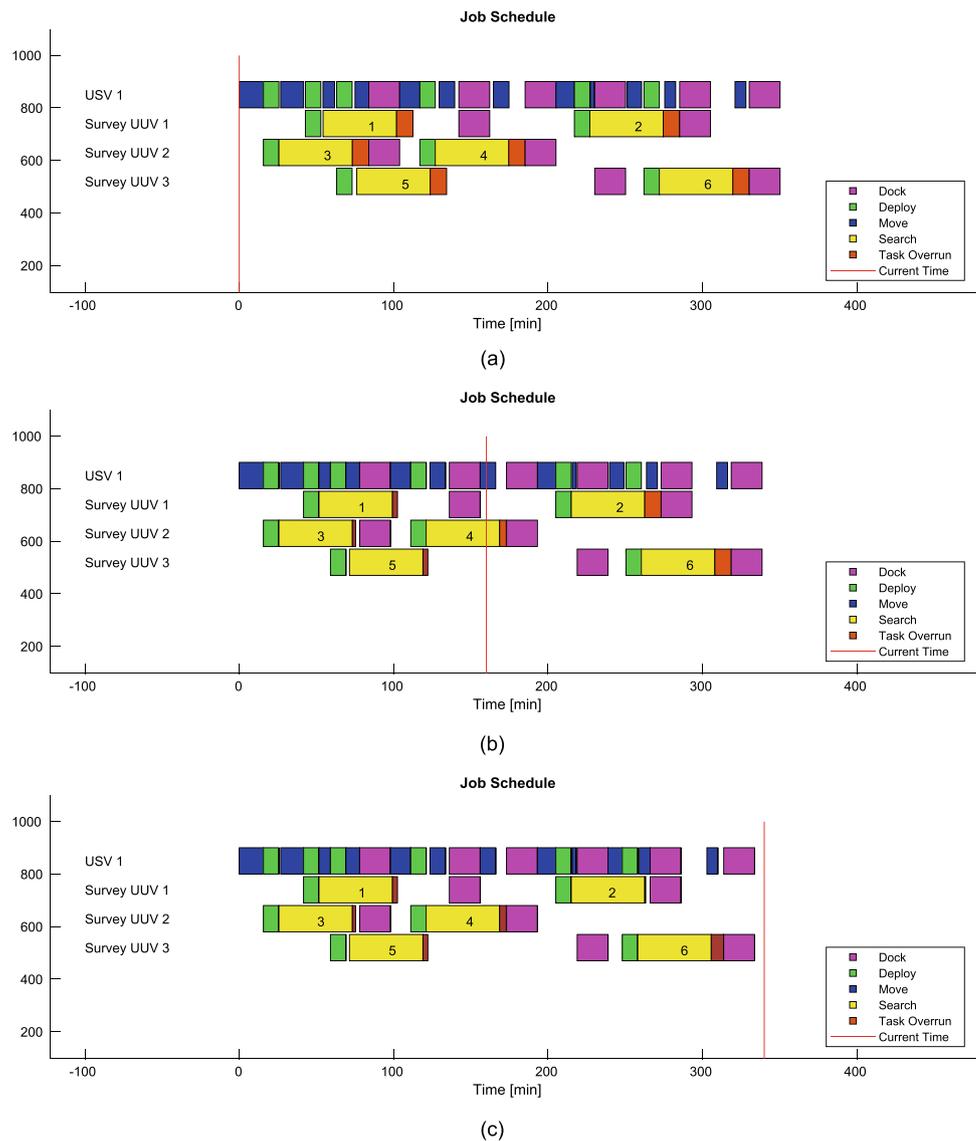
a minimized elasticity objective at around 2 min[4], while the incremental update, while sliding tasks to the right, dramatically increases the chosen metric for schedule differences from the original, as-planned schedule. Maintaining a relatively consistent schedule in lieu of adaptive changes to individual agent tasks is a primary concern in maritime operations involving multiple vehicles. Thus, our method provides

a formal metric for those changes, and minimizes that metric throughout the mission.

7 Conclusion

We have developed an elastic scheduling algorithm coupled with a HMM approach to estimate the completion time

Table 4 Table of algorithm performance compared to a greedy scheduling algorithm combined with an incremental update to the planned start time of each task

	Initial Average Duration (min)	Midway Average Duration (min)	Final Average Duration (min)
Incremental Update - Greedy	50.81	49.91	53.63
HMM - Elastic Scheduler	**58.18**	**53.27**	**53.65**

Results from 40 randomly generated environmental situations. Bold is the proposed algorithm

Schedule Elasticicity Objective vs. Time
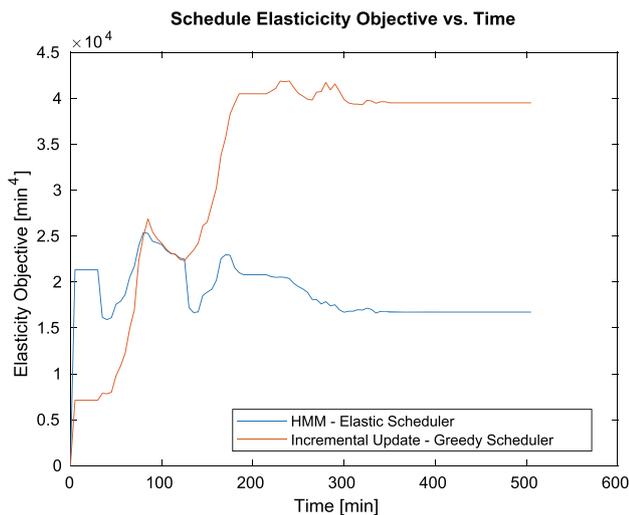
Fig. 8 Minimized objective of elastic scheduling problem vs. time increment for both the HMM - Elastic Scheduler algorithm and the Incremental Update - Greedy Scheduler

of search-level tasks, such as those commonly found in maritime search planning. The elastic scheduling technique involves a quadratic programming approach that requires no binary variables, and can be solved in polynomial time. Future work includes implementing the hidden Markov model and AMAC algorithm on maritime vehicles to provide predictive capabilities, and coordinating multiple vehicles using the elastic scheduler.

Author Contributions Matthew J. Bays contributed to the theoretical development and software implementation of the HMM algorithm and elastic scheduler. Thomas A. Wettergren contributed to the theoretical development of the HMM algorithm and elastic scheduler as well as general refinement of the manuscript. Jane Shin and Shi Chang contributed to the theoretical development and software implementation of the AMAC algorithm. Silvia Ferrari contributed to the theoretical development of the AMAC algorithm and general refinement of the manuscript.

Data Availability No experimental data or large data sets were used in this paper.

Declarations

Conflict of Interest The authors are not aware of any conflicts of interest.

Ethics and Consent This work did not involve human subjects and thus ethical considerations, consent to participate, and consent by subjects to publish do not apply. This manuscript has been designated Distribution Statement A: Approved for public release; distribution is unlimited.

References

1. Baylog, J.G., Wettergren, T.A.: Online determination of the potential benefit of path adaptation in undersea search. IEEE J. Ocean. Eng. (2014)
2. Bennett, A.A., Leonard, J.J.: A behavior-based approach to adaptive feature detection and following with autonomous underwater vehicles. IEEE J. Ocean. Eng. **25**(2), 213–226 (2000)
3. Apker, T., Liu, S.-Y., Sofge, D., Hedrick, J.K.: Application of grazing-inspired guidance laws to autononomous information gathering. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, (2014)
4. Kriminger, E., Cobb, J.T., Príncipe, J.C.: Online active learning for automatic target recognition. IEEE J. Ocean. Eng. **40**(3), 583–591 (2014)
5. Sledge, I.J., Bryner, D.W., Príncipe, J.C.: Annotating motion primitives for simplifying action search in reinforcement learning. IEEE Trans. Emerg. Top. Comput. Intell. **6**(5), 1137–1156 (2022)
6. Sledge, I.J., Emigh, M.S., King, J.L., Woods, D.L., Cobb, J.T., Principe, J.C.: Target detection and segmentation in circular-scan synthetic aperture sonar images using semisupervised convolutional encoder-decoders. IEEE J. Ocean. Eng. **47**(4), 1099–1128 (2022)
7. Zhu, T., Xiao, Y., Zhang, H.: Maritime patrol tasks assignment optimization of multiple usvs under endurance constraint. Ocean Eng. **285**, 115445 (2023)
8. Koes, M., Nourbakhsh, I., Sycara, K.: Constraint optimization coordination architecture for search and rescue robotics. In: International Conference on Robotics and Automation, (2006)
9. Gombolay, M.C., Wilcox, R.J., Shah, J.A.: Fast scheduling of multi-robot teams with temporospatial constraints. In: Robotics: Science and Systems, (2013)
10. Bays, M.J., Wettergren, T.A.: Service agent-transport agent task planning incorporating robust scheduling techniques. Robot. Auton. Syst. **89**, 15–26 (2017)
11. Molineaux, M., Auslander, B., Moore, P.G., Gupta, K.M.: Minimally disruptive schedule repair for mcm missions. In: Proc. SPIE 9454, Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XX, (2015)
12. Sidoti, D., Avvari, G.V., Mishra, M., Zhang, L., Nadella, B.K., Peak, J.E., Hansen, J.A., Pattipati, K.R.: A multiobjective path-planning algorithm with time windows for asset routing in a dynamic weather-impacted environment. IEEE Trans. Syst. Man Cybern. Syst. **47**(12), 3256–3271 (2017)
13. Sidoti, D., Han, X., Zhang, L., Avvari, G.V., Ayala, D.F.M., Mishra, M., Sankavaram, M.S., Kellmeyer, D.L., Hansen, J.A., Pattipati, K.R.: Context-aware dynamic asset allocation for maritime interdiction operations. IEEE Trans. Syst. Man Cybern. Syst. **50**(3), 1055–1073 (2020)

14. De, A., Mamanduru, V.K.R., Gunasekaran, A., Subramanian, N., Tiwari, M.K.: Composite particle algorithm for sustainable integrated dynamic ship routing and scheduling optimization. Comput. Ind. Eng. **96**, 201–215 (2016)

15. Saidi-Mehrabad, M., Fattahi, P.: Flexible job shop scheduling with tabu search algorithms. Int. J. Adv. Manuf. Tech. **32**, 563–570 (2007)

16. Arif, M.U., Haider, S.: A flexible framework for diverse multi-robot task allocation scenarios including multi-tasking. ACM Trans. Auton. Adapt. Syst. (TAAS) **16**(1), 1–23 (2022)

17. Yan, B., Bragin, M.A., Luh, P.B.: Novel formulation and resolution of job-shop scheduling problems. IEEE Robot. Autom. Lett. **3**(4), 3387–3393 (2018)

18. Chakraa, H., Guérin, F., Leclercq, E., Lefebvre, D.: Optimization techniques for multi-robot task allocation problems: Review on the state-of-the-art. Robot. Auton. Syst. 104492 (2023)

19. Dai, W., Lu, H., Xiao, J., Zeng, Z., Zheng, Z.: Multi-robot dynamic task allocation for exploration and destruction. J. Intell. Robot. Syst. **98**, 455–479 (2020)

20. Lippi, M., Marino, A.: A mixed-integer linear programming formulation for human multi-robot task allocation. In: 2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN), pp. 1017–1023. IEEE, (2021)

21. Turner, J., Meng, Q., Schaefer, G., Whitbrook, A., Soltoggio, A.: Distributed task rescheduling with time constraints for the optimization of total task allocations in a multirobot system. IEEE Trans. Cybern. **48**(9), 2583–2597 (2018)

22. Wang, Y.M., Yin, H.L., Qin, K.D.: A novel genetic algorithm for flexible job shop scheduling problems with machine disruptions. Int. J. Adv. Manuf. Tech. **68**, 1317–1326 (2013)

23. Kupiec, J.: Robust part-of-speech tagging using a hidden markov model. Comput. Speech Lang. **6**(3), 225–242 (1992)

24. Zhang, H.-P., Liu, Q., Cheng, X.-Q., Zhang, H., Yu, H.-K.: Chinese lexical analysis using hierarchical hidden markov model. In: Proceedings of the second SIGHAN workshop on Chinese language processing-Volume 17. Association for Computational Linguistics, pp. 63–70 (2003)

25. Juang, B.H., Rabiner, L.R.: Hidden markov models for speech recognition. Technometrics **33**(3), 251–272 (1991)

26. Lee, K.-F., Hon, H.-W.: Speaker-independent phone recognition using hidden markov models. IEEE Trans. Acoust. Speech Signal. Process. **37**(11), 1641–1648 (1989)

27. Evans, J., Krishnamurthy, V.: Optimal sensor scheduling for hidden markov model state estimation. Int. J. Control. **74**(18), 1737–1742 (2001)

28. Ertunc, H.M., Loparo, K.A., Ocak, H.: Tool wear condition monitoring in drilling operations using hidden markov models (hmms). Int. J. Mach. Tools Manuf. **41**(9), 1363–1384 (2001)

29. Gabel, T., Riedmiller, M.: Adaptive reactive job-shop scheduling with reinforcement learning agents. Int. J. Inf. Technol. Intell. Comput. **24**(4), (2008)

30. Buttazzo, G.C., Lipari, G., Caccamo, M., Abeni, L.: Elastic scheduling for flexible workload management. IEEE Trans. Comput. **51**(3), 289–302 (2002)

31. Yang, L., Li, J., Hackney, P., Chao, F., Flanagan, M.: Manual task completion time estimation for job shop scheduling using a fuzzy inference system. In: 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), pp. 139–146. IEEE, (2017)

32. Barceló, J., Montero, L., Marqués, L., Carmona, C.: Travel time forecasting and dynamic origin-destination estimation for freeways based on bluetooth traffic monitoring. Transp. Res. Rec. **2175**(1), 19–27 (2010)

33. Hadachi, A., Lecomte, C., Mousset, S., Bensrhair, A.: An application of the sequential monte carlo to increase the accuracy of travel time estimation in urban areas. In: 2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC), pp. 157–162. IEEE, (2011)

34. Ding, K., Lei, J., Chan, F.T., Hui, J., Zhang, F., Wang, Y.: Hidden markov model-based autonomous manufacturing task orchestration in smart shop floors. Robot. Comput. Integr. Manuf. **61**, 101845 (2020)

35. Bays, M.J., Wettergren, T.A.: A solution to the service agent transport problem. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), September pp. 6443–6450 (2015)

36. Ghahramani, Z.: An introduction to hidden Markov models and Bayesian networks. Int. J. Pattern Recognit. Artif. Intell. **15**, 9–42 (2001)

37. Casella, G., Berger, R.L.: Statistical inference. Crockett, C. (ed.) Duxbury, (2002)

38. Reynolds, J.F.: Some theorems on the transient covariance of Markov chains. J. Appl. Probab. (1972)

39. Shah, D.C., Campbell, M.A.: A robust qualitative planner for mobile robot navigation. In: IEEE International Conference on Robotics and Automation, (2011)

40. Floudas, C.A., Visweswaran, V.: Quadratic optimization. In: Handbook of global optimization, pp. 217–269. Springer, (1995)

41. Chang, S., Isaacs, J., Fu, B., Shin, J., Zhu, P., Ferrari, S.: Confidence level estimation in multi-target classification problems. In: Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XXIII, vol. 10628. International Society for Optics and Photonics, p. 1062818 (2018)

42. Bays, M.J., Shende, A., Stilwell, D.J., Redfield, S.A.: Theory and experimental results for the multiple-aspect coverage problem. Ocean Eng. **54**, (2012)

43. Bays, M.J., Wettergren, A.: Partially-decoupled service agent - transport agent task allocation & scheduling. J. Intell. Robot. Syst. **94**(2), 423–437 (2019)

44. Kahraman, C., Engin, O., Kaya, A., Ozturk, R.E.: Multiprocessor task scheduling in multistage hybrid flow-shops: A parallel greedy algorithm approach. Appl. Soft. Comput. **10**(4), 1293–1300 (2010) optimisation Methods and Applications in Decision-Making Processes

Matthew J. Bays (M '08, SM '17) is the Senior Scientist for Robotics & Optimization at the Naval Surface Warfare Center, Panama City Division (NSWC PCD). He holds a B.S. in Mechanical Engineering from Cornell University, and M.Eng. and Ph.D. degrees in Mechanical Engineering from Virginia Tech with research focusing on robotics, systems, and control. He serves as Principal Investigator on multiple Discovery & Invention programs funded by the Office of Naval Research (ONR). For his work, he is the recipient of a NSWC PCD Commanding Officer / Technical Director Young Professional Exceptional Achievement award, a Naval Sea Systems Command (NAVSEA) Team Innovation Award, and a NAVSEA Team Collaboration Award. His research interests include optimization, multi-agent scheduling, and the development of innovative software packages to ease transition of autonomy algorithms from the lab to the fleet.

Thomas A. Wettergren (SM'06) received the B.S. degree in electrical engineering and the Ph.D. degree in applied mathematics from Rensselaer Polytechnic Institute, Troy, NY, USA. He is currently a Senior Research Scientist and the US Navy Senior Technologist for Operational and Information Science with the Naval Undersea Warfare Center. He also is an adjunct professor of Industrial and System Engineering at the University of Rhode Island. His research interests revolve around the use of applied mathematics to develop new methods for the planning, control, and optimization of complex and multi-agent systems. He is a recipient of the IEEE-USA Harry Diamond Award, the NAVSEA Scientist of the Year, and the Dolores Etter Award for Top Navy Scientists and Engineers. He is an author of the forthcoming book Information-Driven Planning and Control (MIT Press, 2021). He is a senior member of the IEEE and a member of SIAM.

Jaejeong Shin received the B.S. degree in Naval Architecture and Ocean Engineering from Seoul National University in 2017. She received M.S. and Ph.D. degrees in Mechanical Engineering from Cornell University in 2021. Since 2021, she has been an Assistant Professor in the Department of Mechanical and Aerospace Engineering at the University of Florida. Her research interests include robot learning and planning, information-theoretic methods, machine learning, and underwater robotics.

Shi Chang received the B.S. degree in mechanical engineering from the Pennsylvania State University, University Park, PA, USA, in 2017. He is currently pursuing the Ph.D. degree in mechanical engineering at Cornell University, Ithaca, NY, USA. His current research interests include computer vision, machine learning, information theory and information-driven path planning of robots.

Silvia Ferrari Silvia Ferrari is John Brancaccio Professor of Mechanical and Aerospace Engineering at Cornell University. Prior to that, she was Professor of Engineering and Computer Science at Duke University, and Founder and Director of the NSF Integrative Graduate Education and Research Traineeship (IGERT) and Fellowship program on Wireless Intelligent Sensor Networks (WISeNet). Currently, she is the Director of the Laboratory for Intelligent Systems and Controls (LISC) at Cornell University and the co-Director of the Cornell-Unibo Věho Institute on Vehicle Intelligence at the Cornell Tech. Her principal research interests include active perception, robust adaptive control, learning and approximate dynamic programming, and control of multiscale dynamical systems. She is the author of the book "Information-driven Path Planning and Control," MIT Press (2020), and of the TED talk "Do robots dreams of electric sheep?". She received the B.S. degree from Embry-Riddle Aeronautical University and the M.A. and Ph.D. degrees from Princeton University. She is a Fellow of ASME, a Senior Member of the IEEE and AIAA, and a Member of SPIE and SIAM. She is the recipient of the ONR young investigator award (2004), the NSF CAREER award (2005), and the Presidential Early Career Award for Scientists and Engineers (PECASE) award (2006).

Authors and Affiliations

Matthew J. Bays[1] · **Thomas A. Wettergren[2]** · **Jaejeong Shin[3]** · **Shi Chang[4]** · **Silvia Ferrari[4]**

Thomas A. Wettergren
thomas.a.wettergren.civ@us.navy.mil

Jaejeong Shin
jane.shin@ufl.edu

Shi Chang
sc2892@cornell.edu

Silvia Ferrari
ferarri@cornell.edu

[1] Naval Surface Warfare Center, Panama City Division, 110 Vernon Ave, Panama City, FL 32407, USA

[2] Naval Undersea Warfare Center, Newport Division, 1176 Howell St, Newport, RI 02841, USA

[3] Department of Mechanical and Aerospace Engineering, University of Florida, 1064 Center Dr. Gainesville, Panama City, FL 32607, USA

[4] Sibley School of Mechanical and Aerospace Engineering, Cornell University, 616 Thurston Ave., Ithaca, NY 14853, USA