Proceedings of the 2019 Winter Simulation Conference
N. Mustafee, K.-H.G. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y.-J. Son, eds.

WEAKLY COUPLED MARKOV DECISION PROCESSES WITH IMPERFECT INFORMATION

Mahshid Salemi Parizi
Archis Ghate

Department of Industrial and Systems Engineering
University of Washington
Box 352650, 3900 E Stevens Way NE
Seattle, WA 98195, USA

ABSTRACT

Weakly coupled Markov decision processes (MDPs) are stochastic dynamic programs where decisions in independent sub-MDPs are linked via constraints. Their exact solution is computationally intractable. Numerical experiments have shown that Lagrangian relaxation can be an effective approximation technique. This paper considers two classes of weakly coupled MDPs with imperfect information. In the first case, the transition probabilities for each sub-MDP are characterized by parameters whose values are unknown. This yields a Bayes-adaptive weakly coupled MDP. In the second case, the decision-maker cannot observe the actual state and instead receives a noisy signal. This yields a weakly coupled partially observable MDP. Computationally tractable approximate dynamic programming methods combining semi-stochastic certainty equivalent control or Thompson sampling with Lagrangian relaxation are proposed. These methods are applied to a class of stochastic dynamic resource allocation problems and to restless multi-armed bandit problems with partially observable states. Insights are drawn from numerical experiments.

1 BACKGROUND ON WEAKLY COUPLED MDPs

Weakly coupled MDPs arise in applications such as scheduling; resource allocation; queuing; supply chain; and mutli-armed and restless bandit problems (Adelman and Mersereau 2008; Gocgun and Ghate 2010; Hawkins 2003; Parizi and Ghate 2016b; Whittle 1988). In a weakly coupled MDP, $\mathscr{I} = \{1, 2, \ldots, I\}$ denotes the set of sub-MDPs, indexed by i. The finite state-space for the i^{th} MDP is denoted by the set S_i. The state-space of the weakly coupled MDP is then $S = \bigtimes_{i=1}^{I} S_i$ actions in state $s = (s_1, \ldots, s_I)$ of the weakly coupled MDP is $A(s) = \bigtimes_{i=1}^{I} A_i(s_i)$. The set of feasible actions is given by $\bar{A}(s) = \{a \in A(s) : \sum_{i=1}^{I} D_i(s_i, a_i) \le b\}$, where, for each i, $D_i(s_i, a_i)$ are m-dimensional (column) vector functions of (s_i, a_i), and b is an m-dimensional (column) vector. The sub-MDPs are joined through these linking constraints. Upon choosing an action $a \in A(s)$ in state $s \in S$, the decision-maker receives a reward $r(s,a) = \sum_{i=1}^{I} r_i(s_i, a_i)$, and the system transitions to state $s' = (s'_1, \ldots, s'_I)$ with probability $p(s'|s,a) = \prod_{i=1}^{I} p_i(s'_i|s_i, a_i)$. That is, the rewards are additively separable and the transition probabilities are multiplicatively separable. The decision-maker's goal is to maximize the expected total discounted reward over an infinite horizon, with discount factor $0 < \alpha < 1$.

Bellman equations for such a weakly coupled MDP are given by

$$J(s) = \max_{a \in \bar{A}(s)} \left\{ \sum_{i \in \mathscr{I}} r_i(s_i, a_i) + \alpha \sum_{s' \in S} p(s'|s,a) J(s') \right\}, \; s \in S. \tag{1}$$

978-1-7281-3283-9/19/$31.00 ©2019 IEEE

Exact solution of these Bellman equations is computationally intractable. One approximate solution technique based on Lagrangian relaxation is recalled here briefly ((Adelman and Mersereau 2008),(Hawkins 2003)).

The first step in the Lagrangian relaxation approach is to relax the linking constraints and add a corresponding penalty to the objective function using nonnegative Lagrange multipliers $\lambda \in \Re^m_+$ (row vector). This yields the relaxed Bellman equations

$$J^\lambda(s) = \max_{a \in A(s)} \left\{ \sum_{i \in \mathscr{I}} r_i(s_i,a_i) + \lambda(b - \sum_{i=1}^{I} D_i(s_i,a_i)) + \alpha \sum_{s' \in S} p(s'|s,a) J^\lambda(s') \right\}. \tag{2}$$

Adelman and Mersereau (2008) showed that

$$J^\lambda(s) = \frac{1}{1-\alpha}\lambda b + \sum_{i=1}^{I} V_i^\lambda(s_i), \tag{3}$$

where $V_i^\lambda(s_i) = \max\limits_{a_i \in A_i(s_i)} \{r_i(s_i,a_i) - \lambda D_i(s_i,a_i) + \alpha \sum\limits_{s'_i \in S_i} p_i(s'_i|s_i,a_i) V_i^\lambda(s'_i)\}$, for $s_i \in S_i$. The aggregate value function is defined as $H^\lambda(\beta) = \sum\limits_{s \in S} \beta(s) J^\lambda(s)$, where $\beta(s) > 0$ for all s and $\sum\limits_{s \in S} \beta(s) = 1$. Then, applying the linear programming approach (see Puterman (1994)) to Bellman equations (2), we can obtain the best λ and $V_i(\cdot)$, for $i = 1, \ldots, I$. Given $\mathscr{A}_i$ as the set of all state-action pairs for sub-MDP i, we have,

$$\begin{aligned} H^{\lambda^*}(\beta) &= \min_{V(.),\lambda} \frac{\lambda b}{1-\alpha} + \sum_{i=1}^{I} \sum_{s_i \in S_i} \beta_i(s_i) V_i(s_i) \\ V_i(s_i) &\geq r_i(s_i,a_i) - \lambda D_i(s_i,a_i) + \alpha \sum_{s'_i \in S_i} p_i(s'_i|s_i,a_i) V_i(s'_i), (s_i,a_i) \in \mathscr{A}_i, i \in \{1, \ldots, I\}, \lambda \geq 0. \end{aligned} \tag{4}$$

In problem (4), the number of constraints grows linearly in I, and solving this linear program (LP) is computationally tractable in some applications (Adelman and Mersereau 2008; Gocgun and Ghate 2010; Gocgun and Ghate 2012). After an optimal (or sub-optimal) λ^* and $V_i^{\lambda^*}(\cdot)$, for $i = 1, 2, \ldots, I$, are available by solving problem (4) exactly or approximately, these can be substituted back in the right hand side of (1) in place of $J(s')$ via the formula (3). This enables the decision-maker to retrieve decisions in every visited state during runtime by solving (1). These decisions constitute a so-called Lagrangian policy.

Section 2 considers an extension of the above weakly coupled MDP wherein the transition probabilities $p_i(s'_i|s_i,a_i)$ are unknown to the decision-maker. Section 3 studies an extension wherein the decision-maker cannot observe states s_i but instead observes a noisy signal o_i that is probabilistically related to s_i. A general framework for solving such weakly coupled MDPs with imperfect information is currently not available.

A majority of the existing literature on weakly coupled MDPs with imperfect information focuses on a very special case of the problems studied in Section 2. Crucially, the sub-MDPs in these existing papers do not have a physical state. Rather, the states in the sub-MDPs simply correspond to components of the information state. Perhaps the most famous example is the multi-armed bandit problem (Gittins 1979), where sub-MDP i corresponds to the i^{th} arm and an index policy for choosing arms is optimal.

Whittle (1988) developed a sub-optimal index policy for multi-armed *restless* bandit problems (with perfect information). These problems are a special case of weakly coupled MDPs with perfect information described in Section 1. Liu et al. (2013) studied a regret minimization approach for a restless multi-armed bandit problem with unknown transition probabilities. Variations of this approach were pursued for other multi-armed bandit problems in Kalathil et al. (2014). Analytical results in these papers exploited the structure of their problems, and do generalize to our Bayes-adaptive weakly coupled MDPs in Section 2.

Meshram et al. (2016) provided an index policy for two-state restless multi-armed bandit problems with partial observations. They outlined applications to multi-channel communication systems and to advertisement placement systems. Their work is related to the Whittle index (Whittle 1988), but does not seem to generalize to our weakly coupled POMDPs in Section 3. The recent book by Krishnamurthy (2016) also defined a partially observable MDP for such multi-armed bandit problems (even with more than two states). The book mentioned that index policies are not tractable for such high-dimensional problems.

We concisely describe a formal modeling framework for the aforementioned classes of weakly coupled MDPs with imperfect information, and proposes intuitive heuristic solution methods that are easy to implement. The reader is referred to the first author's doctoral dissertation for a more elaborate discussion. We acknowledge that given the complex structure and general nature of the problems and algorithms considered here, it does not appear possible to derive theoretical performance guarantees.

2 BAYES-ADAPTIVE WEAKLY COUPLED MDP

Suppose that the transition probabilities for the i^{th} sub-MDP are characterized by a parameter θ_i. To emphasize this, we denote these transition probabilities by $\psi^i_{\theta_i}(s'_i|s_i,a_i)$. This parameter can take K_i possible values from the set $\Theta_i = \{\theta_i^1, \theta_i^2, \ldots, \theta_i^{K_i}\}$. The decision-maker does not know the actual value of this parameter, but begins with a prior probability mass function on its possible values. This prior is updated as state observations are made. In particular, suppose x_i is a vector of size $K_i - 1$ such that x_i^k is the probability that $\theta_i = \theta_i^k$, for $k \in \{1,2,\ldots,K_i-1\}$. Note that the probability that $\theta_i = \theta_i^{K_i}$ can be calculated simply as $1 - \sum\limits_{k=1}^{K_i-1} x_i^k$ and hence need not be stored. This x_i is called the information state of the problem. If the decision-maker chooses action a_i in physical state s_i and observes physical state s'_i, the information state is updated according to the formula

$$x_i'^k = \frac{x_i^k \psi^i_{\theta_i^k}(s'_i|s_i,a_i)}{\sum\limits_{k=1}^{K_i-1} x_i^k \psi^i_{\theta_i^k}(s'_i|s_i,a_i) + (1 - \sum\limits_{k=1}^{K_i-1} x_i^k)\psi^i_{\theta_i^{K_i}}(s'_i|s_i,a_i)},\ k \in \{1,\ldots,K_i-1\}. \tag{5}$$

The physical-state, information-state pair (s_i,x_i) encodes enough knowledge about the past to make optimal decisions; that is, it is a sufficient statistic (Bertsekas 2007). This allows us to formulate a Bayes-adaptive MDP with states (s_i,x_i) and transition probabilities

$$p_i(s'_i|s_i,x_i,a_i) = x_i^1 \psi^i_{\theta_i^1}(s'_i|s_i,a_i) + \ldots + x_i^{K_i-1}\psi^i_{\theta_i^{K_i-1}}(s'_i|s_i,a_i) + (1 - \sum_{k=1}^{K_i-1} x_i^k)\psi^i_{\theta_i^{K_i}}(s'_i|s_i,a_i). \tag{6}$$

The expected reward is $r_i(s_i,x_i,a_i) = x_i^1 r_{i,\theta_i^1}(s_i,a_i) + x_i^2 r_{i,\theta_i^2}(s_i,a_i) + \ldots + x_i^{K_i-1} r_{i,\theta_i^{K_i-1}}(s_i,a_i) + (1 - \sum\limits_{k=1}^{K_i-1} x_i^k) r_{i,\theta_i^{K_i}}(s_i,a_i)$. The expected rewards $r_{i,\theta_i^k}(s_i,a_i)$ are allowed to depend on the true value of parameter θ_i because the transition probabilities depend on this parameter and generally reward in current state can depend on next state. Bellman equations for this Bayes-adaptive MDP are given by

$$J(s,x) = \max_{a \in \bar{A}(s)} \left\{ \sum_{i=1}^{I} r_i(a_i,s_i,x_i) + \alpha \sum_{s'} \prod_{i=1}^{I} p_i(s'_i|s_i,x_i,a_i) J(s',x') \right\}, s \in S, x \in \bigtimes_{i \in \mathscr{I}} [0,1]^{K_i-1}.$$

This Bayes-adaptive MDP is itself weakly coupled across state components (s_i,x_i). Its exact solution is intractable. The standard approximation based on Lagrangian relaxation as discussed in Section 1 is not applicable because the information state x_i is continuous.

2.1 Methods for Approximate Solution

2.1.1 Semi-Stochastic Certainty Equivalent Control (CEC)

In semi-stochastic CEC, (some of) the random variables in a decision problem are replaced by their nominal (often expected) values (Bertsekas 2007). This is often helpful when the problem includes two (hierarchical) sources of uncertainty as in our framework — one from the decision maker's imperfect knowledge about the probability distributions that describe the system-dynamics and the other from the stochastic evolution of system-state even if these probability distributions were known. Semi-stochastic CEC was applied to optimize lot-sizes in a sequential auction problem with unknown bidder demand in Parizi and Ghate (2016a). For our model, at every time-step, the parameter of the transition probability for problem i is set to equal the expectation of the corresponding prior. That is, in state (s,x), the decision-maker wishes to find an action that solves $\max_{a \in \bar{A}(s)} \left\{ \sum_{i=1}^{I} r_{i,\bar{\theta}_i(x_i)}(a_i,s_i) + \alpha \sum_{s'} \prod_{i=1}^{I} \psi^i_{\bar{\theta}_i(x_i)}(s'_i|s_i,a_i) J_{\bar{\theta}(x)}(s') \right\}$. Here, $\bar{\theta}_i(x_i)$ is the expectation of the prior distribution x_i (rounded to belong to the set Θ_i), and $\bar{\theta}(x) = (\bar{\theta}_1(x_1),\ldots,\bar{\theta}_I(x_I))$. Moreover, $J_{\bar{\theta}(x)}$ is the optimal value function of the (perfect information) weakly coupled MDP whose transition probabilities equal $\psi^i_{\bar{\theta}_i(x_i)}(s'_i|s_i,a_i)$, for $i = 1,2,\ldots,I$. The decision-maker solves this problem approximately by substituting a Lagrangian approximation to $J_{\bar{\theta}(x)}$ as described in formula (3). After implementing the resulting action, the physical state evolves, the information state is updated via (5), and the process continues.

2.1.2 Thompson Sampling

Thompson sampling is a classic algorithm (Strens 2000; Thompson 1933), which was applied to the auction problem in Parizi and Ghate (2016a). It can be implemented in our context as follows. The decision-maker in state (s,x) samples the parameters of the transition probabilities according to the prior distribution x. These sampled parameters are denoted by $\hat{\theta}_i(x_i)$, for $i = 1,2,\ldots,I$. Similar to semi-stochastic CEC, the decision-maker then uses Lagrangian relaxation to approximately solve $\max_{a \in \bar{A}(s)} \left\{ \sum_{i=1}^{I} r_{i,\hat{\theta}_i(x_i)}(a_i,s_i) + \alpha \sum_{s'} \prod_{i=1}^{I} \psi^i_{\hat{\theta}_i(x_i)}(s'_i|s_i,a_i) J_{\hat{\theta}(x)}(s') \right\}$. The decision-maker implements the resulting action, the system evolves to a new physical state, the information state is updated, and the process continues.

2.2 Application to Dynamic Resource Allocation

We consider an imperfect information extension of a dynamic resource allocation problem from Gocgun and Ghate (2010). The case of perfect information is first recalled here.

The set of distinct job-types is denoted by $\mathscr{I} = \{1,2,\ldots,I\}$. The maximum possible number of new arrivals in one period for job-type i is $0 < N^i < \infty$. The probability that $1 \le m \le N^i$ type-i jobs arrive in one period is $p^i(m)$. Arrivals are independent across job-types and also across periods. The service-time for type-i jobs is geometrically distributed with success probability $0 < q^i \le 1$. Service-times are independent across job-types. An ongoing job can be interrupted and resumed later. The set of non-perishable resources is denoted by $\mathscr{J} = \{1,\ldots,J\}$. Moreover, $0 < b^j < \infty$ denotes the quantity of resource $j \in \mathscr{J}$ available in each period. Each type-i job requires $a^{ij} \ge 0$ units of resource j in each period receiving service. For each job-type i, there is at least one resource j such that $a^{ij} > 0$. A separate queue holds incomplete jobs of type-i. The queue capacity is $0 < W^i < \infty$. The following rewards/costs are accumulated in every period. Reward R^i collected at the end of the period per completed type-i job. Holding cost H^i per type-i job in queue, at beginning of time period, after selecting jobs to serve. Cost G^i per rejected type-i job at the end of time period due to it's queue reaches capacity W^i. The decision-maker needs to decide how many jobs of each type to serve in each time-period. The goal is to maximize the expected total discounted profit over an infinite horizon. The discount factor is $0 < \alpha < 1$.

A weakly coupled MDP formulation and an approximate solution procedure based on Lagrangian relaxation for this problem was developed in Gocgun and Ghate (2010). Here, we consider an imperfect information variation, where the decision-maker does not know the success probabilities q^i. The decision-maker knows that these probabilities take one value each from the sets $\Theta_i = \{\theta^i_1,\ldots,\theta^i_{K^i}\}$, for $i = 1,2,\ldots,I$. The definition of the physical state and its evolution, actions, and rewards and costs are identical to the model in Gocgun and Ghate (2010). It is recalled below for completeness. For our imperfect information case, we also need an information state, which stores the probability mass function of the decision-maker's belief about the values of the success probabilities q^i. Specifically, we use information states $x^i_k = P(q^i = \theta^i_k)$, for $k = 1,2,\ldots,K_i - 1$, and for $i = 1,2,\ldots,I$; and the $K_i - 1$-dimensional information state is written simply as x^i. Moreover, $X^i = [0,1]^{K_i-1}$ denotes the set of all possible information states for type-i. The set of all information states is denoted by $X = \bigtimes_{i\in\mathscr{I}} X^i$.

Following Gocgun and Ghate (2010), the physical state of our MDP model is defined as $s = (s^1, s^2, \ldots, s^I)$, where s^i, for $i \in \mathscr{I}$, is the number of incomplete type-i jobs in queue at the beginning of a time period. Let $S^i = \{0,1,\ldots,W^i\}$ for $i \in \mathscr{I}$. The set S of all possible physical states is then defined as the Cartesian product $S = S^1 \times S^2 \times \ldots \times S^I$. The set $\mathscr{S}$ stores all possible physical and information states and is defined as $\mathscr{S} = S \bigtimes X$. The decision vector is represented by $u = (u^1, u^2, ..., u^I)$, where u^i, for $i \in \mathscr{I}$, is the number of type-i jobs that we choose to serve in a time-period after observing the state (s,x). Let $U^i(s^i) = \{0,1,\ldots,s^i\}$ and $U(s) = \bigtimes_{i\in\mathscr{I}} U^i(s^i)$. The set $\bar{U}(s) \subset U(s)$ of all feasible decision vectors in state s is defined by the resource constraints $\bar{U}(s) = \{(u^1,u^2,...,u^I) \in U(s) : \sum_{i\in\mathscr{I}} a^{ij}u^i \le b^j, j = 1,\ldots,J\}$.

Also define $f(s,x,u) = \sum_{i\in I} f_i(s^i,x^i,u^i)$, where $f_i(s^i,x^i,u^i)$ is the expected reward given state (s^i,x^i) and action u^i. It is defined as $f_i(s^i,x^i,u^i) = \Big\{\alpha \sum_{k=1}^{K^i} x^i_k\theta^i_k u^i R^i - H^i(s^i - u^i) - \sum_{k=1}^{K^i}\sum_{n_i=0}^{N^i}\sum_{\eta^i=0}^{u^i} x^i_k p^i(n_i)\binom{u^i}{\eta^i}(\theta^i_k)^{\eta^i}(1 - \theta^i_k)^{u^i-\eta^i} G^i(\max\{(s^i - \eta^i) + n_i - W^i, 0\})\Big\}$. Thus, Bellman equations for all $(s,x) \in \mathscr{S}$ are given by

$$V(s,x) = \max_{u\in\bar{U}(s,x)} \Bigg\{ \sum_{i\in I} f_i(s^i,x^i,u^i) +$$

$$\alpha \sum_{k_1=1}^{K^1} \cdots \sum_{k_I=1}^{K^I} \sum_{n_1=0}^{N^1} \cdots \sum_{n_I=0}^{N^I} \sum_{\eta^1=0}^{u^1} \cdots \sum_{\eta^I=0}^{u^I} \Big(\prod_{i\in I} x^i_{k_i} p^i(n_i) \binom{u^i}{\eta^i} (\theta^i_{k_i})^{\eta^i}(1-\theta^i_{k_i})^{u^i-\eta^i}\Big) V(s',x') \Bigg\}.$$

Here, $s^{i'} = \min\{(s^i - \eta^i) + n^i, W^i\}$, for $i \in \mathscr{I}$. Also, for all $i \in \mathscr{I}$ and $k = \{1,\ldots,K^i - 1\}$, the information state is updated according to $x'^i_k = \frac{x^i_k\binom{u^i}{\eta^i}(\theta^i_k)^{\eta^i}(1-\theta^i_k)^{u^i-\eta^i}}{\sum_{l=1}^{K^i-1} x^i_l\binom{u^i}{\eta^i}(\theta^i_l)^{\eta^i}(1-\theta^i_l)^{u^i-\eta^i} + (1-\sum_{l=1}^{K^i-1} x^i_l)\binom{u^i}{\eta^i}(\theta^i_{K^i})^{\eta^i}(1-\theta^i_{K^i})^{u^i-\eta^i}}$. Suppose either semi-stochastic CEC or Thompson sampling calculates the success probabilities as $\tilde{q}^i$, for $i \in \mathscr{I}$, in physical state s. Then the decision-maker approximately solves Bellman equation

$$V(s) = \max_{u\in\bar{U}(s)} \Bigg\{ \sum_{i\in I} f_i(s^i,u^i) + \alpha \sum_{n_1=0}^{N^1} \cdots \sum_{n_I=0}^{N^I} \sum_{\eta^1=0}^{u^1} \cdots \sum_{\eta^I=0}^{u^I} \Big(\prod_{i\in I} p^i(n_i) \binom{u^i}{\eta^i} (\tilde{q}^i)^{\eta^i}(1-\tilde{q}^i)^{u^i-\eta^i}\Big) V(s') \Bigg\}$$

using Lagrangian relaxation. Here, $s^{i'} = \min\{(s^i - \eta^i) + n^i, W^i\}$, for $i \in \mathscr{I}$ and, $f_i(s^i,u^i) = \big\{\alpha u^i\tilde{q}^i R^i - H^i(s^i - u^i) - \sum_{n_i=0}^{N^i}\sum_{\eta^i=0}^{u^i} p^i(n_i)\binom{u^i}{\eta^i}(\tilde{q}^i)^{\eta^i}(1-\tilde{q}^i)^{u^i-\eta^i} G^i(\max\{(s^i - \eta^i) + n_i - W^i, 0\})\big\}$. The Lagrangian relaxation approach for this problem is described in Gocgun and Ghate (2010). Numerical results are presented next.

2.2.1 Computational Results for Dynamic Resource Allocation

Following Gocgun and Ghate (2012), Parizi and Ghate (2016b), and Parizi et al. (2017), we generated problem sets randomly and ran simulations to examine the performance of semi-stochastic CEC and Thompson sampling. The simulations were performed on a 3.1 GHz iMac desktop with 16 GB RAM and an Intel Core i7 chip running Mac OS X 10.9.3. All linear and integer programs were solved using CPLEX 12 from IBM via a MATLAB R2015b front-end.

Problem instances were generated as follows. The number of job-types for the problem sets in Table 1 is set to $I = 6$ and $I = 12$, respectively. The maximum number of new arrivals in each period, which is denoted N^i for every i, was set to a random uniformly distributed integer in the interval $[1,4]$. The probability $p^i(m)$ for $1 \leq m \leq N^i$ was generated by normalizing N^i uniform $(0,1)$ random variables. We assumed for simplicity that the set of possible success probability values, $\Theta_i = \{\theta_1^i, \dots \theta_{K^i}^i\}$, is identical for all i. We set it to $\Theta_i = \{0.1, 0.2, \dots, 0.9\}$. Thus, $K^i = 9$ for all i. The number of shared resources was set to $J = 1$. The resource availability and resource consumption were generated via an approach similar to Parizi and Ghate (2016b). The resource consumption for each job type is a random uniformly distributed integer value from set $[1,2]$. The resource availability is the summation of resource consumptions of all job types multiplied by the total number of job types I and the tightness ratio denoted by ρ in the first two columns of Tables 1. That is, after generating the resource consumption values, the resource availability is derived as $I \times \rho \times \sum_{i=1}^{I} a_{ij}$. Similar to Parizi and Ghate (2016b), we can alter the level of resource scarcity by changing the tightness ratio. The maximum capacity of the queue denoted by W^i for each i is set to a random uniformly distributed integer value from the interval $[1,5]$. Similarly, for the reward R^i, rejection cost G^i, and holding cost H^i, a random uniformly distributed integer value from the intervals $[1,100]$, $[1,4]$, and $[1,10]$ was sampled, respectively. The discount factor α was fixed at 0.99.

For every problem instance, we estimated the performance of different methods by averaging total discounted profit over 200 sample paths with 50 time-steps each. To examine the benefit of learning with semi-stochastic CEC and Thompson sampling, we also implemented a third policy in our numerical experiments. We call this the "no-learning" policy, wherein the decision-maker does not update the initial belief and assumes that the transition probabilities are given by the expectation with respect to the initial belief as in Equation (6). This yields a perfect information weakly coupled MDP with physical states only. Lagrangian relaxation is then employed to obtain an approximate value function.

For each row in Table 1 and for a fix number of job which is either 6 or 12, we generated 30 test instances randomly. For each test instance we run the simulation using Thompson sampling, semi-stochastic CEC and no-learning. For each of the 200 independent sample path replications for each problem instance, we started with a uniform belief over Θ_i, and a random physical state. The initial physical and belief states were identical for all three methods. Note that for every replication we have a discounted profit generated with three different policies over 50 time steps. So for every problem generated in every row we have a vector of size 200 which we can use to see whether semi-stochastic CEC or Thompson sampling performs statistically different as compared to no-learning. In Table 1, the "significant Semi-CEC" and "significant Thompson" columns report the number of test instances out of 30 wherein semi-stochastic CEC and Thompson sampling are statistically different from no-learning, respectively. To check this, we used t-test at the significance threshold of 0.05. For test instances where learning policies were statistically different from no-learning, we calculated the average discounted profit over 200 replications for the no-learning policy and divided that with the corresponding value using the learning policy. The average ratio over all test instances where the no-learning and learning policies were statistically different, is reported in columns "No-learn over Semi-CEC" and "No-learn over Thompson" of Table 1.

The tables show that there is an improvement in the profit generated by using semi-stochastic CEC and Thompson sampling as compared to no-learning. For this dynamic resource allocation problem, there is no significant difference between semi-stochastic CEC and Thompson sampling. The improvement achieved by the learning methods is less significant when resource availability is very low or very high. In the middle range of resource availability, the performance of learning policies improves when less resource is available. That is, the ratio of no-learning over learning decreases as tightness ratio decreases in the middle range. This is intuitive because a simple heuristic could perform well when resources are plentiful.

Table 1: Results for 12 problem sets, each with 30 instances.

ρ		significant Semi-CEC		significant Thompson		No-learn over Semi-CEC		No-learn over Thompson	
6 jobs	12 jobs	6 jobs	12 jobs	6 jobs	12 jobs	6 jobs	12 jobs	6 jobs	12 jobs
0.1	0.05	5	5	7	6	0.66	0.89	0.79	0.92
0.15	0.08	22	28	26	27	0.72	0.83	0.80	0.85
0.2	0.1	25	29	24	29	0.84	0.82	0.85	0.84
0.25	0.13	23	26	24	24	0.91	0.92	0.92	0.93
0.35	0.15	15	21	16	18	0.97	0.97	0.97	0.96
0.45	0.2	7	7	6	6	0.95	0.98	0.95	0.98

3 WEAKLY COUPLED PARTIALLY OBSERVABLE MDP

Standard terminology for POMDPs is available in Krishnamurthy (2016). We describe a weakly coupled POMDP as follows. We have a set $\mathscr{I} = \{1,2,\ldots,I\}$ of POMDPs. The state-space for the i^{th} POMDP is $S_i = \{1,2,\ldots,n_i\}$. Let $S = \bigtimes_{i\in\mathscr{I}} S_i$. Define $\vec{x}_i = \{x_i(1), x_i(2), \ldots, x_i(n_i)\}$, where $x_i(j)$ is the decision maker's probabilistic belief that the i^{th} POMDP is in state $j \in S_i$. The belief state-space for the i^{th} POMDP is $\Delta^i = \{\vec{x}^i \in \mathscr{R}_+^{n_i} | \sum_{j=1}^{n_i} x_i(j) = 1\}$. The action-space for the i^{th} POMDP is $A_i = \{1,2,\ldots,m_i\}$. Let $A = \bigtimes_{i\in\mathscr{I}} A_i$. Observation-space for the i^{th} POMDP is $\mathscr{O}_i = \{1,2,\ldots,O_i\}$. Let $\mathscr{O} = \bigtimes_{i\in\mathscr{I}} \mathscr{O}_i$. Transition probabilities for POMDP i are $p_i(s'_i|s_i,a_i)$. Let $p(s'|s,a) = \prod_{i\in\mathscr{I}} p_i(s'_i|s_i,a_i)$. Observation probabilities for POMDP i are $f_i(o_i|s_i,a_i)$. Let $f(o|s,a) = \prod_{i\in\mathscr{I}} f_i(o_i|s_i,a_i)$. Rewards are $r(s'|a,s) = \sum_{i\in\mathscr{I}} r_i(s'_i|a_i,s_i)$. The coupling constraints are $\sum_{i\in\mathscr{I}} D_i(a_i) \leq K$. Here, $D_i(a_i)$ is a vector function of decisions a_i. We cannot allow the constraints to depend on s_i, because this state is not observable. Discount factor is $0 < \alpha < 1$ and the goal is to maximize the expected total discounted reward over an infinite horizon.

Define $\phi_i(o'_i|\vec{x}_i,a_i)$ as the probability of observing o'_i given current belief state $\vec{x}_i$ and action a_i for sub-problem i. We have, $\phi_i(o'_i|\vec{x}_i,a_i) = \sum_{s'_i\in S_i} f_i(o'_i|s'_i,a^i) \sum_{s_i\in S_i} p_i(s'_i|s_i,a_i)x_i(s_i)$. Let $\phi(o'|\vec{x},a) = \prod_{i\in\mathscr{I}} \phi_i(o'_i|\vec{x}_i,a_i)$. The decision-maker updates the belief vector for POMDP i in a Bayesian fashion, given current belief $\vec{x}_i$, action a_i, and new observation o'_i. This update is given by

$$x'_i(s'_i) = \frac{f_i(o'_i|s'_i,a_i) \sum_{s_i\in S_i} p_i(s'_i|s_i,a_i)x_i(s_i)}{\sum_{s'_i\in S_i} f_i(o'_i|s'_i,a_i) \sum_{s_i\in S_i} p_i(s'_i|s_i,a_i)x_i(s_i)}, \ \forall s'_i \in S_i. \tag{7}$$

Bellman equations for this weakly coupled POMDP and $\forall \vec{x}_i \in \Delta^i$ are given by

$$V(\vec{x}_1,\vec{x}_2,\ldots,\vec{x}_I) = \max_{\vec{a}\in A, \sum_{i\in\mathscr{I}} D_i(a_i)\leq K} \left\{ \sum_{i\in\mathscr{I}} \sum_{s_i\in S_i} x_i(s_i) r_i(s_i,a_i) + \alpha \sum_{\vec{o}'\in\mathscr{O}} \prod_{i\in\mathscr{I}} \phi_i(o'_i|\vec{x}_i,a_i) V(\vec{x'}_1,\vec{x'}_2,\ldots,\vec{x'}_I) \right\}. \tag{8}$$

POMDPs are computationally difficult even for "moderate-sized" state-spaces. Here, the state-space is high-dimensional, rendering exact solution intractable.

3.1 Methods for Approximate Solution

3.1.1 Semi-Stochastic CEC

In semi-stochastic CEC, using the belief state $\vec{x}_i$ on the state space S_i, we define $\bar{s}_i$ as the expected state. The decision-maker then finds an action that solves $\max\limits_{\vec{a}\in A,\ \sum\limits_{i\in\mathscr{I}} D_i(a_i)\leq K} \left\{ \sum\limits_{i\in\mathscr{I}} r_i(\bar{s}_i,a_i) + \alpha \sum\limits_{s'\in\mathscr{S}} \prod\limits_{i\in\mathscr{I}} p_i(s'_i|\bar{s}_i,a_i)V(s') \right\}$. Here, s' is the next state after $\bar{s}$, and $V(s')$ is the optimal value function of a weakly coupled MDP with observable states. To solve this problem, the decision-maker uses a Lagrangian approximation of $V(s')$, as in formula (3) of Section 1. After the resulting decision is implemented, the physical state evolves, and the information state is updated based on formula (7) after making a new observation. This process continues.

3.1.2 Thompson Sampling

In Thompson sampling, the decision-maker samples an estimated physical state $\hat{s}_i$ from the belief distribution $\vec{x}_i$, for each i. Similar to semi-stochastic CEC, Lagrangian relaxation is employed to approximately solve $\max\limits_{\vec{a}\in A,\ \sum\limits_{i\in\mathscr{I}} D_i(a_i)\leq K} \left\{ \sum\limits_{i\in\mathscr{I}} r_i(\hat{s}_i,a_i) + \alpha \sum\limits_{s'\in\mathscr{S}} \prod\limits_{i\in\mathscr{I}} p_i(s'_i|\hat{s}_i,a_i)V(s') \right\}$. After the resulting decision is implemented, the physical state evolves to a new state, the information state is updated, and the process continues.

3.1.3 Discretization

Discretization simply means that we solve (8) by using a finite discretization $\vec{\sigma}'_i$ of the continuous future belief state $\vec{x'}_i$, for each $i \in \mathscr{I}$. That is, in state x, the decision-maker solves

$$\max_{\vec{a}\in A,\ \sum\limits_{i\in\mathscr{I}} D_i(a_i)\leq K} \left\{ \sum_{i\in\mathscr{I}} \sum_{s_i\in S_i} x_i(s_i) r_i(s_i,a_i) + \alpha \sum_{\vec{o}'\in\mathscr{O}} \prod_{i\in\mathscr{I}} \phi_i(o'_i|\vec{x}_i,a_i) V(\vec{\sigma}'_1,\vec{\sigma}'_2,\ldots,\vec{\sigma}'_I) \right\}, \tag{9}$$

where $\vec{\sigma}'_i$ is the rounded future belief state inside set $\mathscr{X}^{n_i}$ and $\mathscr{X}$ is a finite set of values between 0 and 1. Also note that $V(\vec{\sigma}'_1,\vec{\sigma}'_2,\ldots,\vec{\sigma}'_I)$ is the optimal value function of the discretized state version of the Bellman equation (8). Problem (8) is itself weakly coupled with continuous state space, so the value of $V(\vec{\sigma}'_1,\vec{\sigma}'_2,\ldots,\vec{\sigma}'_I)$ which is the relaxed discretized version of $V(\vec{x}_1,\vec{x}_2,\ldots,\vec{x}_I)$ is the optimal value function of a weakly couple MDP with a finite state-space; which can be approximated using Lagrangian relaxation via formula (3). A detailed implementation of discretization is discussed in section 3.2.2.

3.2 Application to Partially Observable Restless Multi-Armed Bandits

In a restless multi-armed bandit problem (Hawkins 2003; Krishnamurthy 2016), there are I types of projects. Projects of type-i have a state-space $S_i = \{1,2,\ldots,n_i\}$, however, the decision-maker cannot observe project-states. At every time-step, the decision-maker chooses K out of these I projects to work on. These chosen projects are called active. A reward $r_i(j)$ is collected for type-i active projects in state $j \in S_i$. We use $\vec{r}_i \in \mathscr{R}^{n_i}_+$ to denote the type-i reward vector. The states of both active and inactive projects can evolve; however, no reward is collected from inactive projects. Let a_i be a binary decision variable that is 1 if project i is chosen. The coupling constraint thus is $\sum\limits_{i=1}^{I} a_i = K$. Bellman equation for this problem is

$$V(\vec{x}_1,\vec{x}_2,\ldots,\vec{x}_I) = \max_{\vec{a}=\{0,1\}^I,\ \sum\limits_{i\in\mathscr{I}} a_i = K} \left\{ \sum_{i\in\mathscr{I}} a_i \vec{x}_i.\vec{r}_i + \alpha \sum_{\vec{o}'\in\mathscr{O}} \prod_{i\in\mathscr{I}} \phi_i(o'_i|\vec{x}_i,a_i) V(\vec{x'}_1,\vec{x'}_2,\ldots,\vec{x'}_I) \right\}. \tag{10}$$

For semi-stochastic CEC and Thompson sampling, we first get an estimated $\tilde{s}_i$ from the information vector $\vec{x}_i$ as described above. Then we apply Lagrangian relaxation to approximately solve

$$V(\tilde{s}_1,\tilde{s}_2,\ldots,\tilde{s}_I)=\max_{\vec{a}=\{0,1\}^I,\ \sum\limits_{i\in\mathscr{I}}a_i=K}\left\{\sum_{i\in\mathscr{I}}a_i\vec{r}_i(\tilde{s}_i)+\alpha\sum_{\vec{s}'\in S}\prod_{i\in\mathscr{I}}p_i(s_i'|\tilde{s}_i,a_i)V(s_1',s_2',\ldots,s_I')\right\}. \tag{11}$$

The steps involved in this Lagrangian relaxation method are briefly described next.

3.2.1 Lagrangian Relaxation within Semi-Stochastic CEC or Thompson Sampling for Partially Observable Restless Multi-Armed Bandits

The relaxed Lagrangian counterpart of (11) is written as

$$V^\lambda(\tilde{s}_1,\tilde{s}_2,\ldots,\tilde{s}_I)=\max_{\vec{a}=\{0,1\}^I}\left\{\sum_{i\in\mathscr{I}}a_i\vec{r}_i(\tilde{s}_i)+(\sum_{i\in\mathscr{I}}a_i-K)\lambda+\alpha\sum_{\vec{s}'\in S}\prod_{i\in\mathscr{I}}p_i(s_i'|\tilde{s}_i,a_i)V^\lambda(s_1',s_2',\ldots,s_I')\right\},$$

for any λ. As described in Section 1, we know that, for all $s\in S$, $V^\lambda(s)=\frac{1}{1-\alpha}\lambda K+\sum\limits_{i=1}^{I}L_i^\lambda(s_i)$, where $L_i^\lambda(s_i)=\max\limits_{a_i=\{0,1\}}\{a_i\vec{r}_i(s_i)-\lambda a_i+\alpha\sum\limits_{s_i'\in S_i}p_i(s_i'|s_i,a_i)L_i^\lambda(s_i')\}$. The decision-maker solves the LP

$$\begin{aligned}&\min_{\lambda,L}\sum_{i\in\mathscr{I}}\sum_{s_i\in S_i}\beta_iL_i(s_i)+\lambda\frac{K}{1-\alpha}\\&L_i(s_i)\geq a_i\vec{r}_i(s_i)-\lambda a_i+\alpha\sum_{s_i'\in S_i}p_i(s_i'|s_i,a_i)L_i(s_i'),\ \forall s_i\in S_i,\ \forall i\in\mathscr{I}\ \text{and}\ a_i=\{0,1\},\end{aligned}$$

to obtain the optimal λ and value function approximations $L_i(s_i)$. Since this LP does not depend on the current information state x, we solve it only once to get λ^* and value function L^*. For solving problem (11) at each time-step, we use $V(s')\approx\frac{K\lambda^*}{1-\alpha}+\sum\limits_{i\in\mathscr{I}}L_i^*(s_i')$ as the approximate value function. This yields the linear binary problem $\max\limits_{\vec{a}=\{0,1\}^I,\ \sum\limits_{i\in\mathscr{I}}a_i=k}\left\{\sum\limits_{i\in\mathscr{I}}a_i\vec{r}_i(\tilde{s}^i)+\alpha\sum\limits_{i\in\mathscr{I}}\sum\limits_{\vec{s}_i'\in S_i}[a_ip_i(s_i'|\tilde{s}_i,a_i=1)L_i^*(s'^i)+(1-a_i)p_i(s_i'|\tilde{s}_i,a_i=0)L_i^*(s'^i)]\right\}$.
The resulting decision is implemented, the system evolves to a new physical state, and the belief state is updated after making a new observation. This process then continues.

3.2.2 Discretization for Partially Observable Restless Multi-Armed Bandits

We need to first approximately solve for $V(\vec{\sigma}^1,\vec{\sigma}^2,\ldots,\vec{\sigma}^I)$ and use it in (9) to retrieve the decision. The relaxed Lagrangian Bellman equation counterpart of $V(\vec{\sigma}^1,\ldots,\vec{\sigma}^I)$ for this application is given by $V^\lambda(\vec{\sigma}^1,\ldots,\vec{\sigma}^I)=\max\limits_{\vec{a}\{0,1\}^I}\left\{\sum\limits_{i\in\mathscr{I}}\sum\limits_{s_i\in S_i}a_i\sigma_i(s_i)\vec{r}_i(s_i)+(\sum\limits_{i\in\mathscr{I}}a_i-K)\lambda+\alpha\sum\limits_{\vec{o}'\in\mathscr{O}}\prod\limits_{i\in\mathscr{I}}\phi_i(o_i'|\vec{\sigma}_i,a_i)V^\lambda(\vec{\sigma'}^1,\ldots,\vec{\sigma'}^I)\right\}$, $\forall\vec{\sigma}_i\in\mathscr{X}^{n_i}$. The best λ and approximate value functions $L_i(\vec{\sigma}_i)$ are obtained by solving the LP

$$\begin{aligned}&\min_{\lambda,L}\sum_{i\in\mathscr{I}}\sum_{\vec{\sigma}_i\in\mathscr{X}^{n_i}}\beta_iL_i(\vec{\sigma}_i)+\lambda\frac{K}{1-\alpha}\\&L_i(\vec{\sigma}_i)\geq\sum_{s_i\in S_i}\vec{\sigma}_i(s_i)a_i\vec{r}_i(s_i)-\lambda a_i+\alpha\sum_{o_i'\in O_i}\phi_i(o_i'|\vec{\sigma}_i,a_i)L_i(\vec{\sigma'}_i),\ \forall i\in\mathscr{I},\ \forall\vec{\sigma}_i\in\mathscr{X}^{n_i},\ a_i\in\{0,1\}.\end{aligned} \tag{12}$$

Problem (12) is only solved once, since it does not depend on the current information state. The resulting λ^* and L^* are employed to approximately retrieve a decision. Problem (10) is then solved by using

$V(\vec{x}'_1,\vec{x}'_2,\dots,\vec{x}'_I) \approx \frac{K\lambda^*}{1-\alpha} + \sum_{i\in\mathscr{I}} L_i^*(\vec{\sigma}'_i)$ at every time-step. Here, $\vec{\sigma}'_i$ is the rounded value of $\vec{x}'_i$. Specifically, the decision retrieval problem

$$\max_{\vec{a}=\{0,1\}^I,\ \sum_{i\in\mathscr{I}} a_i=k} \left\{ \sum_{i\in\mathscr{I}} a_i\vec{r}_i\vec{x}_i + \alpha \sum_{i\in\mathscr{I}} \sum_{o'_i\in O_i} [a_i\phi_i(o'_i|\vec{x}_i,a_i=1)L_i^*(\vec{\sigma}'_i) + (1-a_i)\phi_i(o'_i|\vec{x}_i,a_i=0)L_i^*(\vec{\sigma}'_i)] \right\}$$

is solved at every time-step, given state $(\vec{x}_1,\vec{x}_2,\dots,\vec{x}_I)$. Here, for every $o'_i \in O_i$, the corresponding $\vec{\sigma}'_i$ is calculated using the update rule (7) and rounding.

The number of constraints in problem (12) grows exponentially in n_i so its solution is slow for large problems. In the computational results in Section 3.2.3 below, we tackled (12) via the affine approximation and constraint generation approach that was originally proposed in Gocgun and Ghate (2012) for large-scale weakly coupled MDPs. That is, similar to Parizi and Ghate (2016b), and Gocgun and Ghate (2012), we defined an affine approximation for $L_i(\vec{x}_i)$ and used constraint generation to solve the resulting approximation of (12). The number of constraints then grows linearly in I and n_i. This approximation of (12) was computationally much faster to solve. The affine approximation was then also used to retrieve decisions. This retrieval problem was a linear binary problem. A brief description is presented next.

Affine approximation and constraint generation: Here, value functions L_i are approximated by affine functions. That is, we substitute $L_i(\vec{x}) = d_i + \sum_{s_i\in S_i} x(s_i)c_i(s_i)$ with unknown coefficients $d_i \in \mathbb{R}$ and $c_i \in \mathbb{R}^{n_i}$. Optimal coefficient values are found by solving (12) after making this substitution. This yields

$$\begin{aligned}
&\min_{\lambda,c,d} \sum_{i\in\mathscr{I}} \sum_{\vec{\sigma}_i\in\mathscr{X}^{n_i}} \beta_i[d_i + \sum_{s_i\in S_i} \sigma_i(s_i)c_i(s_i)] + \lambda\frac{K}{1-\alpha} \\
&d_i + \sum_{s_i\in S_i} \sigma_i(s_i)c_i(s_i) \geq \sum_{s_i\in S_i} \sigma_i(s_i)a_i\vec{r}_i(s_i) - \lambda a_i + \alpha \sum_{o'_i\in O_i} \phi_i(o'_i|\vec{\sigma}_i,a_i)d_i + \\
&\alpha \sum_{o'_i\in O_i} \sum_{s'_i\in S_i} c_i(s_i)\Big(f_i(o'_i|s'_i,a_i) \sum_{s_i\in S_i} p_i(s'_i|s_i,a_i)\sigma_i(s_i)\Big),\ \forall i\in\mathscr{I},\ \forall\vec{\sigma}_i\in\mathscr{X}^{n_i} \text{ and } a_i\in\{0,1\}.
\end{aligned} \tag{13}$$

We apply constraint generation to (13). The first step is to find an initial subset of constraints in (13) that yields an optimal solution. In the numerical implementation, we randomly generated a vector $\vec{\sigma}_i \in \mathscr{X}^{n_i}$ and we added the corresponding constraints with $a_i \in \{0,1\}$. This process was repeated until an initial optimal solution λ^*, d^*, c^* was found. Using λ^*, d^* and c^*, we then solved a maximum constraint violation problem to find a constraint that is most violated by the current optimal values λ^*, d^* and c^*. In particular, for every $i \in \mathscr{I}$ and every $a_i \in \{0,1\}$, we solved problem

$$\begin{aligned}
&\max_{\vec{\sigma}_i\in\mathscr{X}^{n_i}} \sum_{s_i\in S_i} \vec{\sigma}_i(s_i)a_i\vec{r}_i(s_i) - \lambda^*a_i + \alpha \sum_{o'_i\in O_i} \phi_i(o'_i|\vec{\sigma}_i,a_i)d_i^* + \\
&\alpha \sum_{o'_i\in O_i} \sum_{s'_i\in S_i} c_i^*(s_i)\Big(f_i(o'_i|s'_i,a_i) \sum_{s_i\in S_i} p_i(s'_i|s_i,a_i)\sigma_i(s_i)\Big) - \Big(d_i^* + \sum_{s_i\in S_i} \sigma_i(s_i)c_i^*(s_i)\Big),
\end{aligned}$$

wherein the decision variable is $\vec{\sigma}_i$. The optimal objective values for each $i \in \mathscr{I}$ and each $a_i \in \{0,1\}$ are saved, and then the largest among them equals the maximum violation. The corresponding constraint was added to the initial set of constraints. The optimization problem (13) is solved using this new expanded set of constraints to get new optimal coefficients. This is repeated until a stopping criterion proposed in Gocgun and Ghate (2012) and Parizi and Ghate (2016b) is met. According to this stopping criterion, the constraint generation process is terminated when the value $\frac{\mathscr{I}Z^*}{(1-\alpha H_c)}$ drops below a small tolerance level. Here, Z^* is the latest amount of maximum violation, and H_c is the optimal value of (13) with the recent feasible region. This iterative constraint generation procedure is executed only once at the beginning and the resulting optimal affine value function approximation is employed for decision retrieval in every visited state $(\vec{x}^1, \vec{x}^2, \ldots, \vec{x}^I)$ during runtime. The decision retrieval problem is given by

$$
\begin{aligned}
&\max_{\vec{a}=\{0,1\}^I,\ \sum\limits_{i\in\mathscr{I}} a_i=k} \Bigg\{ \sum_{i\in\mathscr{I}} a_i \vec{r}_i \vec{x}_i + \alpha \sum_{i\in\mathscr{I}} \sum_{o_i'\in O_i} [a_i \phi_i(o_i'|\vec{x}_i, a_i=1) d_i^* + (1-a_i)\phi_i(o_i'|\vec{x}_i, a_i=0) d_i^*] + \\
&\alpha \sum_{i\in\mathscr{I}} \sum_{o_i'\in O_i} \sum_{s_i'\in S_i} \Bigg[a_i c_i^*(s_i') \Big(f_i(o_i'|s_i', a_i=1) \sum_{s_i\in S_i} p_i(s_i'|s_i, a_i=1) x_i(s_i) \Big) + \\
&(1-a_i) c_i^*(s_i') \Big(f_i(o_i'|s_i', a_i=0) \sum_{s_i\in S_i} p_i(s_i'|s_i, a_i=0) x_i(s_i) \Big) \Bigg] \Bigg\}.
\end{aligned}
$$

A solution of this linear binary problem provides an implementable decision.

3.2.3 Computational Results for Partially Observable Restless Multi-Armed Bandits

Similar to previous sections, we randomly generated test instances of restless multi-armed bandit problems. The first column of Table 2 lists the four parameters that were fixed before generating these problem instances. The first one is I, the number of projects. We assumed for simplicity that the number of possible project states is equal across all projects. That is, $n_i = N$, for all $i \in \mathscr{I}$. Thus, N is the second parameter we used for problem generation. Recall that K equals the number of projects selected by the decision-maker at each time-step. This is the third parameter. We assume for simplicity that the number of possible observations $O_i = O$ for each i. Thus, O is the fourth parameter of our test instances. We generated 30 instances for each of 6 different (I,N,K,O) combinations. This produced 180 problem instances.

The reward vector $\vec{r}_i$, which is of size N, stores the reward obtained by selecting project i in different states. These reward numbers were fixed to equal random uniformly distributed integers from the interval $[1, 100]$. The transition probabilities $p_i(s_i'|s_i, 1)$ and $p_i(s_i'|s_i, 0)$ were generated randomly for each project i as follows. For each project, the probabilities of transiting from a given state to all possible next states is a set of normalized random values between $(0, 1)$. A similar approach was pursued for generating outcome measurement probabilities $f_i(o_i'|s_i', a_i)$. For each project, the probabilities of observing $\{1, 2, \ldots, O\}$ from a given state s_i' were fixed at normalized random values between $(0, 1)$. The discount factor α was 0.99.

We ran simulations over 200 independent sample paths each with 50 times steps. We started every sample path with a uniform belief vector for the actual state of every project. We also started each path by randomly sampling a physical state for each project. Four policies were compared: semi-stochastic CEC, Thompson sampling, discretized, and affine-discretized. The first three are as discussed in sections 3.1.1, 3.1.2 and 3.1.3, respectively. The "affine-discretized" obtains a policy via discretization as discussed in 3.1.3, but by using an affine value function approximation along with constraint generation for the relaxed LP problem. The affine-discretized policy is computationally tractable for large problems whereas discretization (without affine approximation) is not. For each problem instance, the performance of all policies is normalized with respect to the same restless multi-armed bandit problem but with perfect state observations. This latter is called "true-MDP." This true-MDP serves as an idealized "best-case" scenario since we expect the performance of various approximation methods to be worse in the partially observable case than with perfect observations. In particular, for every test instance, the average discounted profit over 200 sample paths using semi-stochastic CEC was divided by the corresponding value for the true-MDP. The average of all such values over 30 test instances is reported in the second column of 2. A similar calculation was performed and is reported for Thompson sampling, discretization, and affine-discretization in columns three, four, and five, respectively. For $I = 10$, discretization (without affine approximation) was not applied since it was slow. The table shows that Thompson sampling performed better than other policies, however, we do not have an intuitive explanation for this and it may not hold in other examples.

Table 2: Results of 6 problem sets, each with 30 instances.

Problem setting (I,N,K,O)	Semi-CEC over true-MDP	Thompson over true-MDP	Discretized over true-MDP	Affine-Discretized over true-MDP
(10,7,5,6)	0.60	0.77	-	0.63
(10,7,1,6)	0.56	0.85	-	0.56
(5,5,4,4)	0.67	0.89	0.72	0.71
(5,5,3,4)	0.69	0.89	0.72	0.71
(5,5,2,4)	0.74	0.94	0.71	0.77
(5,5,1,4)	0.66	0.90	0.69	0.71

REFERENCES

Adelman, D., and A. J. Mersereau. 2008. "Relaxations of Weakly Coupled Stochastic Dynamic Programs". *Operations Research* 56(3):712–727.

Bertsekas, D. P. 2007. *Dynamic Programming and Optimal Control*, Volume 1 and 2. Nashua, New Hampshire: Athena Scientific.

Gittins, J. C. 1979. "Bandit Processes and Dynamic Allocation Indices". *Journal of the Royal Statistical Society: Series B (Methodological)* 41(2):148–164.

Gocgun, Y., and A. Ghate. 2010. "A Lagrangian Approach to Dynamic Resource Allocation". In *Proceedings of the 2010 Winter Simulation Conference*, edited by B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yücesan, 3330–3338. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Gocgun, Y., and A. Ghate. 2012. "Lagrangian Relaxation and Constraint Generation for Allocation and Advance Scheduling". *Computers & Operations Research* 39(10):2323–2336.

Hawkins, J. 2003. *A Lagrangian Decomposition Approach to Weakly Coupled Dynamic Optimization Problems and Its Applications*. Ph.D.thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts. https://dspace.mit.edu/handle/1721.1/29599.

Kalathil, D., N. Nayyar, and R. Jain. 2014. "Decentralized Learning for Multiplayer Multiarmed Bandits". *IEEE Transactions on Information Theory* 60(4):2331–2345.

Krishnamurthy, V. 2016. *Partially Observed Markov Decision Processes*. Cambridge, UK: Cambridge University Press.

Liu, H., K. Liu, and Q. Zhao. 2013. "Learning in a Changing World: Restless Multiarmed Bandit with Unknown Dynamics". *IEEE Transactions on Information Theory* 59(3):1902–1916.

Meshram, R., D. Manjunath, and A. Gopalan. 2016. "On the Whittle Index for Restless Multi-Armed Hidden Markov Bandits". *IEEE Transactions on Automatic Control* 63(9):3046 – 3053.

Parizi, M. S., and A. Ghate. 2016a. "Lot-Sizing in Sequential Auctions while Learning Bid and Demand Distributions". In *Proceedings of the 2016 Winter Simulation Conference*, edited by T. M. K. Roeder, P. I. Frazier, R. Szechtman, E. Zhou, T. Huschka, and S. E. Chick, 895–906. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Parizi, M. S., and A. Ghate. 2016b. "Multi-Class, Multi-Resource Advance Scheduling with No-Shows, Cancellations and Overbooking". *Computers & Operations Research* 67:90–101.

Parizi, M. S., Y. Gocgun, and A. Ghate. 2017. "Approximate Policy Iteration for Dynamic Resource-Constrained Project Scheduling". *Operations Research Letters* 45(5):442–447.

Puterman, M. 1994. *Markov Decision Processes*. Hoboken, New Jersey: John Wiley and Sons.

Strens, M. 2000. "A Bayesian Framework for Reinforcement Learning". In *Proceedings of the 2000 International Conference on Machine Learning*, 943–950. San Francisco, California: International Conference on Machine Learning.

Thompson, W. R. 1933. "On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples". *Biometrika* 25(3-4):285–294.

Whittle, P. 1988. "Restless Bandits: Activity Allocation in a Changing World". *Journal of Applied Probability* 25(A):287–298.

AUTHOR BIOGRAPHIES

MAHSHID SALEMI PARIZI is a data scientist at Microsoft. She holds a PhD in Industrial and Systems Engineering from University of Washington, Seattle. Her email address is msalemip@uw.edu.

ARCHIS GHATE is a Professor in the Department of Industrial and Systems Engineering at the University of Washington, Seattle. He holds a PhD in Industrial and Operations Engineering from University of Michigan. His email address is archis@uw.edu.