IEEE Control Systems Letters paper presented at
2023 American Control Conference (ACC)
San Diego, CA, USA. May 31 - June 2, 2023

On the Complexity and Approximability of Optimal Sensor Selection for Mixed-Observable Markov Decision Processes

Jayanth Bhargav, Mahsa Ghasemi and Shreyas Sundaram

Abstract— **Mixed-Observable Markov Decision Processes (MOMDPs) are used to model systems where the state space can be decomposed as a product space of a set of state variables, and the controlling agent is able to measure only a subset of those state variables. In this paper, we consider the setting where we have a set of potential sensors to select for the MOMDP, where each sensor measures a certain state variable and has a selection cost. We formulate the problem of selecting an optimal set of sensors for MOMDPs (subject to certain budget constraints) to maximize the expected infinite-horizon reward of the agent and show that this sensor placement problem is NP-Hard, even when one has access to an oracle that can compute the optimal policy for any given instance. We then study a greedy algorithm for approximate optimization and show that there exist instances of the MOMDP sensor selection problem where the greedy algorithm can perform arbitrarily poorly. Finally, we provide some empirical results of greedy sensor selection over randomly generated MOMDP instances and show that, in practice, the greedy algorithm provides near-optimal solutions for many cases, despite the fact that one cannot provide general theoretical guarantees for its performance. In total, our work establishes fundamental complexity results for the problem of optimal sensor selection (at design-time) for MOMDPs.**

I. Introduction

Real-world decision problems such as autonomous navigation, robotic task execution, data center operation and machine maintenance are made difficult by imperfect knowledge about the state of the system (due to partial or noisy observations). These systems can be modelled as Markov Decision Processes (MDPs), and variants like Partially-Observable MDPs (POMDPs) and Multi-Objective MDPs. Many algorithms have been developed to find the optimal policy for sequential decision-making for such systems.

Mixed-Observable Markov Decision Processes (MOMDPs) are a variant of POMDPs, where a part of the state is observable. We consider a class of MOMDPs where the state space can be decomposed into a product space of a set of state variables, and only a subset of the state variables are measurable. For classical POMDPs, several algorithms like *Batch Enumeration* [1] and the *Witness or Incremental Pruning* [2] have been proposed that can compute optimal policies. In order to solve high-dimensional POMDPs, [3] proposes a *Point-Based Value Iteration* approach and [4] proposes an efficient point-based POMDP planning algorithm for approximating belief-space reachability. These algorithms have been extended to solve MOMDPs by exploiting the structure to reduce the dimensionality of the value function [5]. However, these algorithms primarily focus on reducing the computational time required for solving MOMDPs, and do not study the problem of sensor (or observation) set selection for such systems in order to achieve optimal performance. While the problem of sensor selection has been very well studied for other classes of systems (e.g., linear dynamical systems [6]-[7]), there has been no prior work on optimal sensor selection for MOMDPs.

*This work was supported by National Science Foundation Collaborative Research Grant CCRI 2120430. The authors are with Elmore Family School of Electrical & Computer Engineering, Purdue University, West Lafayette IN 47907. Email addresses: {jbhargav, mahsa, sundara2}@purdue.edu.

A. Motivation

In many autonomous systems, the number of sensors that can be installed is limited by a certain budget and system design constraints [8]. In case of robotics, system designers often face the challenge of optimizing the sensor placement in order to achieve certain design objectives, such as maximizing observability and performance of the robot [9]-[10]. The authors of [8] consider a path planning task where a mobile robot has to map an unknown environment by gathering information from a large sensor network. Due to limited communication budget, the robot can access only a limited number of sensors. Reinforcement learning techniques have also been employed in applications like network congestion control [11], load-balancing [12] and energy optimization for large data-centers [13], where one has partial or limited observability of the system. However, the problem of selecting the optimal set of sensors that can result in better performance of these systems has not been studied in the literature. Given such scenarios where one can only utilize a limited number of sensors in a system for sequential decision-making, in this paper, we focus on the problem of selecting the best set of sensors at *design-time* (under some budget constraints) for a MOMDP which can maximize the optimal expected infinite-horizon return for an agent.

B. Contributions

The problem of finding an optimal policy for general finite-horizon POMDPs is *PSPACE Complete* [14]. In this paper, we consider a special case of POMDPs, namely MOMDPs, and show that the sensor selection problem for MOMDPs is NP-Hard, even when one has access to an oracle that can compute the optimal policy for any given instance of MOMDP. Second, we show how greedy algorithms for sensor selection can perform arbitrarily poorly for some instances of this problem. This also shows that the value function (objective function) of this problem is not generally submodular in the set of sensors selected. Finally, we provide experimental results for the greedy algorithm for several randomly generated instances of the budgeted sensor selection problem and observe that,

979-8-3503-2806-6/$31.00 ©2023 AACC

although greedy algorithms can perform poorly for certain instances, they produce near-optimal solutions in many cases.

C. Related Work

In [15], the authors consider active perception under a limited budget for POMDPs to selectively gather information at runtime. However, in our problem, we consider design-time sensor selection for MOMDPs, where the sensor set is not allowed to dynamically change at runtime. A body of literature considers the problem of sensor placement or sensor scheduling for sequential decision-making tasks and model the task of sensor placement itself as a POMDP [16], [17]. However, we consider the problem of selecting the optimal set of sensors for a MOMDP.

In [6] and [7], the authors study the sensor selection and sensor attack problems for Kalman filtering of linear dynamical systems, where the objective is to reduce the trace of the steady-state error covariance of the filter. The authors of [7] show that these problems are NP-Hard and there exists no polynomial-time constant-factor approximation algorithms for such class of problems. Linear system models often cannot accurately capture the dynamics of complex systems in robotics applications; instead, such systems are often modelled as MDPs or its variants like POMDPs and MOMDPs. We analyze the complexity and approximability of sensor selection for MOMDPs in this paper.

For combinatorially-hard sensor selection problems, various approximation algorithms have proven to produce near-optimal solutions [18]-[19]. The authors of [20] exploit the weak-submodularity property of the objective function and provide near-optimal greedy algorithms for sensor selection. In contrast to these results, we demonstrate that greedy algorithms for sensor selection in MOMDPs can perform arbitrarily poorly and the value function of a MOMDP is not generally submodular in the set of sensors selected.

II. Background & Preliminaries

A general MOMDP is defined by the tuple $\mathcal{M} := (\mathcal{S} = \mathcal{S}_v \times \mathcal{S}_h, \mathcal{O} = \mathcal{O}_v \times \mathcal{O}_w, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma, b_0)$, where $\mathcal{S}$ is a finite discrete state space (decomposed into the visible part $\mathcal{S}_v$ and the hidden part $\mathcal{S}_h$), $\mathcal{O}$ is a finite discrete observation space (decomposed into $\mathcal{O}_v$ – the part of the observations that match the visible state space $\mathcal{S}_v$ and $\mathcal{O}_w$ – the remaining observations), $\mathcal{A}$ is a finite discrete action space, $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is a probabilistic transition function, $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, and $0 < \gamma < 1$ is the discount factor. The initial probability distribution over the states (initial belief) is given by b_0.

We consider a class of MOMDPs where the state space is a product space of a set of state variables defined by the tuple $\mathcal{M} := (\mathcal{S} = \mathcal{S}_1 \times \mathcal{S}_2 \times \mathcal{S}_3 \times \cdots \times \mathcal{S}_n, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma, b_0)$, in which we denote $\mathcal{S}_v = \prod_{i \in \mathcal{V}} \mathcal{S}_i$ to be the visible state space and $\mathcal{S}_h = \prod_{i \in \mathcal{H}} \mathcal{S}_i$ to be the hidden state space, where $\mathcal{V}$ and $\mathcal{H}$ denote the indices of the visible and hidden state variables. We do not define an observation space explicitly for this class of MOMDPs, since the observation space $\mathcal{O}$ is just $\mathcal{O}_v$ which exactly equals $\mathcal{S}_v$.

The agent maintains a belief over the true state of the environment $b \in \mathcal{B}$, where $\mathcal{B}$ is the belief space, which is the set of probability distributions over the states in $\mathcal{S}$. Denote $\mathcal{T}(s, a, s')$ to be the state-transition function such that $\mathcal{T}(s, a, s') = \mathbb{P}(s_{t+1} = s' \mid s_t = s, a_t = a)$.

In MOMDPs, the reward obtained by the agent is *belief-based*, denoted as $\rho(b, a)$, and is given by $\rho(b, a) = \sum_s b(s) r(s, a)$, where $b(s)$ is the belief over the state s, a is the action and $r(s, a)$ is the reward obtained for taking action a in the state s. The goal of an agent is to maximize the expected infinite-horizon reward, given the initial belief b_0. The objective is then to find an optimal policy satisfying $\pi^* = \arg\max_{\pi \in \Pi} V^\pi(b_0)$ with $V^\pi(b_0) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \rho_t \mid b_0, \pi\right]$, where ρ_t is the reward obtained at time t. The policy π belongs to a class of policies Π which map the history of observations, actions, and rewards to the next action. The function $V^\pi(b)$ can be computed using the value iteration algorithm based on dynamic programming with $V_0^\pi(b) = 0$, and $V^\pi(b) = \lim_{n \to \infty} V_n^\pi(b)$ (as described in [5]).

III. Problem Formulation

Consider an agent interacting with a MOMDP $\mathcal{M} : (\mathcal{S} = \mathcal{S}_1 \times \mathcal{S}_2 \times \mathcal{S}_3 \times \cdots \times \mathcal{S}_n, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma, b_0)$, where $\mathcal{S}$ is the state space decomposed into sub-spaces $\mathcal{S}_i$, each corresponding to a state-variable s_i (which takes values from $\mathcal{S}_i$), and $\mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma, b_0$ as defined in the previous section.

Define $\Omega = \{\omega_i \mid i = 1, 2, \ldots, n\}$ to be a collection of sensors, where the sensor ω_i measures exactly the state variable s_i. Let $c_i \in \mathbb{R}_{\geq 0}$ be the cost we pay to measure state s_i by placing the sensor ω_i, and let $C \in \mathbb{R}_{>0}$ denote the total budget for the sensor placement. Let $\Gamma \subseteq \Omega$ be the subset of sensors selected (at design-time) that generates observations $Y_\Gamma(t) = \{s_i(t) \mid \omega_i \in \Gamma\}$. At time t, the agent has the following information: observations $Y_t = \{Y_\Gamma(0), Y_\Gamma(1), Y_\Gamma(2), \cdots, Y_\Gamma(t)\}$, actions $A_t = \{a_0, a_1, a_2, \cdots, a_{t-1}\}$ and rewards $R_t = \{r_0, r_1, r_2, r_3, \cdots, r_{t-1}\}$.

Let $\Theta_t = \{Y_t, A_t, R_t\}$ denote the set containing all the information the agent has until time t. Define $\Pi_\Gamma = \{\pi_\Gamma \mid \pi_\Gamma : \Theta_t \to \mathcal{A}\}$ to be a class of history-dependent policies that map from a set containing all the information known to the agent until time t to the action a_t which the agent takes at time t. The goal is to find an optimal subset of sensors $\Gamma^* \subseteq \Omega$, under the budget constraint, to be placed that can maximize the expected value V_Γ^* of the infinite-horizon discounted reward obtained by the agent under the optimal policy for that subset of sensors. We aim to solve the optimization problem:

$$\max_{\Gamma \subseteq \Omega} \; V_\Gamma^*$$
$$\textit{s.t.} \sum_{\omega_i \in \Gamma} c_i \leq C.$$

We now define a decision version for the above optimization problem as the *Mixed Observable Markov Decision Process Sensor Selection Problem (MOMDP-SS Problem)*.

Problem 1 (MOMDP-SS Problem): Consider a MOMDP $\mathcal{M}$ and a set of n sensors Ω, where each sensor $\omega_i \in \Omega$ is associated with a cost $c_i \in \mathbb{R}_{\geq 0}$. For a value $V_\Gamma \in \mathbb{R}$ and sensor budget $C \in \mathbb{R}_{>0}$, is there a subset of sensors $\Gamma \subseteq \Omega$, such that the optimal infinite-horizon expected discounted return (or just referred to as return) V_Γ^* for the optimal policy in Π_Γ satisfies $V_\Gamma^* \geq V_\Gamma$ and the total cost of the sensors selected satisfies $\sum_{\omega_i \in \Gamma} c_i \leq C$?

IV. Complexity Analysis

We begin with some preliminary lemmas, which we will use in characterizing the complexity of MOMDP-SS.

A. Preliminary Results

Consider the following instance of MOMDP-SS Problem.

Example 1: Consider a MOMDP given by $\bar{\mathcal{M}} := \{\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma, b_0\}$ having state space $\mathcal{S} = \mathcal{S}_1 = \{A, B\}$, action space $\mathcal{A} = \{0, 1\}$, transition function $\mathcal{T} = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}$ for each action $a \in \mathcal{A}$, reward function $\mathcal{R}(s, a) = (r(A, 0) = R, r(A, 1) = -R, r(B, 0) = -R, r(B, 1) = R)$ with $R > 0$, discount factor $\gamma \in (0, 1)$ and $b_0 = [0.5, 0.5]$. Fig. 1 describes state-action transitions along with their probabilities. The state space of this MOMDP has only one sub-space $\mathcal{S}_1$, with state variable s and the agent can measure this state by selecting a noiseless sensor $\omega = s$. Let the sensor cost be $c_1 = 1$ and the budget be $C = 1$.

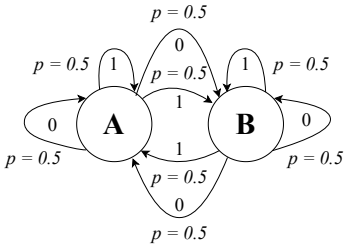

Fig. 1. State transition diagram of $\bar{\mathcal{M}}$.

Lemma 1: For the MOMDP $\bar{\mathcal{M}}$ defined in Example 1, the following holds for $\gamma \in (0, 1)$:

(i) If the state of $\bar{\mathcal{M}}$ is measured (i.e., the agent knows if $s = A$ or $s = B$), the optimal infinite-horizon expected reward beginning at any state is $V^*(s) = \frac{R}{(1-\gamma)}$.

(ii) If the state of $\bar{\mathcal{M}}$ is not measured i.e., the agent only has access to the sequence of actions and rewards, but not the current state s, then the optimal infinite-horizon expected reward beginning at a uniform belief is $V^*(b) = 0$.

Proof: We will prove both (i) and (ii) as follows.

Case (i): Consider the case when state of the MOMDP $\bar{\mathcal{M}}$ is measured using sensor ω. Based on the specified reward function, we can see that the agent can obtain the maximum reward (R) at each time-step by choosing action 0 when $s = A$, and action 1 when $s = B$. This yields $V^*(s) = \max_{\pi_\Gamma} V^{\pi_\Gamma}(s) = \sum_{t=0}^{\infty} \gamma^t R = \frac{R}{(1-\gamma)}$.

Case (ii): Consider the case when the state of the MOMDP $\bar{\mathcal{M}}$ is not measured (i.e., the sensor ω is not selected and as a result the agent does not know the current state but only has access to the sequence of actions and rewards).

Due to uncertainty in the state, the agent maintains a belief b. The agent performs a Bayesian update of its belief at each time step using the information it has (i.e., the history of actions and observations) [3]. Consider a uniform initial belief for the agent. By construction, the agent has an equal probability of being in either state A or state B, at each time-step, regardless of the history. One can easily verify that the agent's belief over the states will always be equal to uniform belief (stationary distribution of the state transition matrix), i.e., $b = [0.5, 0.5]$. It is also easy to verify that the optimal policy for the given instance of MOMDP is to take action $'1'$ when in state B and to take action $'0'$ when in state A. Therefore, the expected reward at each time-step is 0 (since the state could be either A or B with equal probability). Thus, we have $V^*(b) = 0$. ■

B. NP-Hardness of the MOMDP-SS Problem

In this section, we provide a reduction from the well-known NP-Complete Knapsack Problem to the MOMDP-SS Problem and prove that MOMDP-SS is NP-Hard. Consider the decision-version of the Knapsack Problem (KSP) [21].

Problem 2 (The Knapsack Problem (KSP)): There are a total of n items, $[1, 2, ..., n]$, where each item i has a value $v_i \in \mathbb{R}_{\geq 0}$ and a weight $w_i \in \mathbb{R}_{\geq 0}$. The Knapsack problem is to decide, given two positive numbers W and V_0, whether there exists a subset $I \subseteq [1, ..., n]$ such that $\sum_{i \in I} w_i \leq W$ and $\sum_{i \in I} v_i \geq V_0$.

Theorem 1: The MOMDP-SS Problem is NP-Hard.

Proof: We give a reduction from Knapsack to the MOMDP-SS Problem. Consider the KSP with n items with non-negative weights $(w_1, w_2, ..., w_n)$, non-negative values $(v_1, v_2, ..., v_n)$, weight threshold $W > 0$ and value threshold $V_0 > 0$. The goal is to decide if there is a subset of items with total weight at most W, such that the corresponding total value is at least V_0. Given the above instance of the KSP, we now proceed to construct an instance of MOMDP-SS Problem.

Let the MOMDP $\mathcal{M}^*$ consist of n identical sub-MOMDPs $\{\mathcal{M}_1, \mathcal{M}_2, ..., \mathcal{M}_n\}$ where each MOMDP $\mathcal{M}_i : \{\mathcal{S}_i, \mathcal{A}_i, \mathcal{T}_i, \mathcal{R}_i, \gamma_i, (b_0)_i\}$ is the MOMDP $\bar{\mathcal{M}}$ as defined in Example 1. We will now define the state-space, action-space, transition function, reward function and discount factor for $\mathcal{M}^*$ as follows:

State Space $\mathcal{S}$: Consider states S_i corresponding to the MOMDP $\mathcal{M}_i$. We define the state space $\mathcal{S}$ of the n-state MOMDP $\mathcal{M}^*$ as

$$\mathcal{S} := \{(S_1, S_2, \ldots, S_n) \colon S_i \in \{A, B\}\}. \tag{1}$$

Action Space $\mathcal{A}$: Consider the actions A_i corresponding to the MOMDP $\mathcal{M}_i$. We define the action space $\mathcal{A}$ of the n-state MOMDP $\mathcal{M}^*$ as

$$\mathcal{A} := \{(A_1, A_2, \ldots, A_n) \colon A_i \in \{0, 1\}\}. \tag{2}$$

Transition Function $\mathcal{T}$: The probabilistic transition function $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ of the n - state MOMDP $\mathcal{M}^*$ is

a constant function for any present state (s), action (a) and next state (s') combination, given by

$$\mathcal{T} := \mathbb{P}(s' = \cdot |\ a = \cdot, s = \cdot) = \frac{1}{2^n}. \tag{3}$$

Note that the transition function is a constant and can be compactly represented, for polynomial-time reduction.

Discount Factor γ: Let the discount factors γ_i 's of all MOMDP's $\mathcal{M}_i$ be equal to each other and equal to the discount factor of MOMDP $\mathcal{M}^*$,

$$\gamma = \gamma_1 = \gamma_2 = \ldots = \gamma_n = 0.95. \tag{4}$$

Reward Function $\mathcal{R}$: Let reward R_i of MOMDP $\mathcal{M}_i$ be chosen such that $R_i = v_i(1-\gamma)$ where v_i corresponds to the value of the i^{th} item of Knapsack. Denote $\mathcal{R}_i(s_i, a_i) : \mathcal{S}_i \times \mathcal{A}_i \rightarrow \{-R_i, R_i\}$ to be the reward function of the MOMDP $\mathcal{M}_i$ (as defined in Example 1). We define the reward function of the n - state MOMDP $\mathcal{M}^*$ as

$$\mathcal{R} := \sum_{i=1}^{n} \mathcal{R}_i(s_i, a_i). \tag{5}$$

Denote the set of available sensors as Ω. The sensor $\omega_i \in \Omega$ can exactly measure the state s_i (i.e., the sensor ω_i can identify if the state s_i of the MOMDP $\mathcal{M}_i$ is A or B). Set the cost of sensor ω_i to be $c_i = w_i$, where w_i is the weight of the i^{th} item of Knapsack. Set the value function threshold to be $V_\Gamma = V_0$ and the sensor budget to be $C = W$, where V_0 and W are the value and weight thresholds of the Knapsack problem, respectively. Let the initial belief b_0 be a uniform probability vector.

We have now successfully created an instance of the MOMDP-SS problem using an instance of the Knapsack problem. The n-state MOMDP-SS Problem instance is thus: MOMDP $\mathcal{M}^* : \{\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma, b_0\}$ (1) - (5), a set of n sensors Ω with costs $\{c_1, c_2, \ldots c_n\}$, sensor budget C and expected infinite-horizon return (value function threshold) V_Γ. The goal is to decide if there is a subset of sensors $\Gamma \subseteq \Omega$ with total cost at most C, such that the corresponding infinite-horizon expected return for an agent interacting with the MOMDP is at least V_Γ.

Suppose the answer to the Knapsack Problem is True, then there exists a subset of indices $I \subseteq [1, 2, ..., n]$ such that the total value satisfies $\sum_{i \in I} v_i \geq V_0$ and the total weight satisfies $\sum_{i \in I} w_i \leq W$. By construction $\sum_{i \in I} w_i \leq W$ implies $\sum_{i \in I} c_i \leq C$. Since the state transitions of MOMDP $\mathcal{M}^*$ are decoupled and the reward $\mathcal{R}$ is an algebraic sum of the rewards $\mathcal{R}_i$, the value function of the MOMDP $\mathcal{M}^*$ is also an algebraic sum of the value functions of the individual MOMDPs, i.e, $V(s) = \sum_{i=1}^{n} V_i(s_i)$. By Lemma 1, the return for the MOMDP $\mathcal{M}_i$ is $R_i/(1-\gamma)$ in case the sensor ω_i is selected or 0 in case the sensor ω_i is not selected. Thus, we get $V_\Gamma^* = \sum_{i \in I} R_i/(1-\gamma) = \sum_{i \in I} v_i$. Since $\sum_{i \in I} v_i \geq V_0 = V_\Gamma$, we have $V_\Gamma^* \geq V_\Gamma$, and thus the answer to the constructed MOMDP-SS instance is also True.

Conversely, if the answer to the MOMDP-SS is True, then there exists a subset of sensors $\Gamma \subseteq \Omega$ such that $\sum_{\omega_i \in \Gamma} c_i \leq C$ and $V_\Gamma^* \geq V_\Gamma$. Let $I \subseteq [1, 2, ..., n]$ denote the indices of the sensor subset. It follows from the previous arguments that $\sum_{i \in I} v_i \geq V_0$ and $\sum_{i \in I} w_i \leq W$. Thus, the answer to the KSP is True. We now have a polynomial-time reduction from the Knapsack Problem (KSP) to the MOMDP-SS Problem. Since the Knapsack Problem is NP-Complete [21], the MOMDP-SS Problem is NP-Hard. ■

V. Approximability of Sensor Selection

Greedy algorithms, which iteratively and myopically choose items that provide the largest immediate benefit, provide computationally tractable and near-optimal solutions to many combinatorial optimization problems [19],[22]-[23]. Algorithm 1 provides a greedy approach for any given instance of MOMDP-SS with uniform sensor costs to output a subset of sensors to be selected.

Algorithm 1: Greedy Algorithm for MOMDP-SS

Data: MOMDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma, b_0)$, set of candidate sensors Ω, uniform sensor costs $\mathcal{C} = (c_1, c_2, ..., c_n)$, and sensor budget C
Result: A set Γ of selected sensors
$k \leftarrow 0, \Gamma \leftarrow \emptyset$
while $k \leq C$ **do**
  for $i \in (\Omega \setminus \Gamma)$ **do**
    Calculate optimal expected return $V^*(\Gamma \cup \{i\})$
  end
  $j = \arg\max_i (V^*(\Gamma \cup \{i\}))$
  $\Gamma \leftarrow \Gamma \cup \{j\}, k \leftarrow k + c_j$
end

In this section, we present an explicit example showing that the greedy algorithm can perform arbitrarily poorly for even simple cases of the MOMDP-SS with just 4 states.

Example 2: Consider an instance of the MOMDP-SS problem with the MOMDP $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma, b_0\}$ constructed with 4 sub-MOMDPs $\{\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3, \mathcal{M}_4\}$, where MOMDPs $\{\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3\}$ are the single-state variable MOMDPs as defined in Example 1. The MOMDP $\mathcal{M}_4$, has the state s_4, which depends on states s_2 and s_3 as $s_4 = s_2 \oplus s_3$, where $\oplus$ is the Exclusive-OR (XOR) Boolean function over the binary states s_2 and s_3.[1] The state transitions of s_4 depend on the independent state transitions of s_2 and s_3, while the state transitions of s_1, s_2, s_3 are independent of each other. However, the action space and reward function for the MOMDP $\mathcal{M}_4$ is the same as that of the single-state variable MOMDP as defined in Example 1.
State Space $\mathcal{S}$*:* The state space $\mathcal{S}$ of the MOMDP $\mathcal{M}$ is a set of 4-tuples defined as $\mathcal{S} := \{(s_1, s_2, s_3, s_4) | (s_1, s_2, s_3) \in \{A, B\}^3, s_4 = s_2 \oplus s_3\}$;
Action Space $\mathcal{A}$*:* The action space $\mathcal{A}$ of the MOMDP $\mathcal{M}$ is a set of 4-tuples defined as $\mathcal{A} := \{(a_1, a_2, a_3, a_4) | (a_1, a_2, a_3, a_4) \in \{0, 1\}^4\}$;
Transition Function $\mathcal{T}$*:* The probabilistic transition function

[1]The XOR function over the binary states $\{A, B\}$ is defined as follows: $\{A \oplus A = A, A \oplus B = B, B \oplus A = B, B \oplus B = A\}$

$\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ of the MOMDP $\mathcal{M}$ is a constant function given by $\mathcal{T} := \mathbb{P}(s' = \cdot |\ a = \cdot, s = \cdot) = \frac{1}{16}$;
Reward Function $\mathcal{R}$: Let the reward functions $\mathcal{R}_i(s_i, a_i)$ of MOMDPs $\mathcal{M}_i$ for $i = \{1, 2, 3, 4\}$ be defined as in Example 1, with $R_i \in \mathbb{R}_{\geq 0}$ such that $R_4 > R_2 = R_1 > R_3$.
We now define the reward function for the MOMDP $\mathcal{M}$ as

$$\mathcal{R} := \mathcal{R}_1(s_1, a_1) + \mathcal{R}_2(s_2, a_2) + \mathcal{R}_4(s_4, a_4); \qquad (6)$$

Discount Factor γ: Let the discount factors of MOMDPs $\mathcal{M}_i$ be equal to the discount factor of $\mathcal{M}$ i.e., $\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = \gamma$.

Let $\Omega = \{\omega_1, \omega_2, \omega_3\}$ be the set of sensors which can measure states, s_1, s_2, s_3 respectively. Let the cost of the sensors be $\mathcal{C} = (c_1 = 1, c_2 = 1, c_3 = 1)$ and the sensor budget be $C = 2$. Assume uniform initial beliefs (b_0) for all the MOMDPs $\mathcal{M}_i$ and $\mathcal{M}$. We now apply the greedy algorithm described in Algorithm 1 to this instance of the MOMDP-SS. For any such instance of MOMDP-SS, define $r_{gre}(\Gamma) = \frac{V_\Gamma^{gre}}{V_\Gamma^{opt}}$, where V_Γ^{gre} and V_Γ^{opt} are the infinite-horizon expected return obtained by the greedy algorithm and the optimal infinite-horizon expected return respectively. Define $h = R_4/R_2$.

Proposition 1: For the instance of MOMDP-SS problem described in Example 2, the ratio $r_{gre}(\Gamma)$ satisfies $\lim_{h\to\infty} r_{gre}(\Gamma) = 0$.

Proof: By Equation (6), we know that the overall reward of the MOMDP $\mathcal{M}$ depends on the individual rewards of MOMDPs $\mathcal{M}_1$, $\mathcal{M}_2$ and $\mathcal{M}_4$. However, there is no sensor that can measure the state s_4 directly. In the first iteration, the greedy algorithm will have to break-tie between ω_1 and ω_2, because $R_1 = R_2$. In general, many greedy algorithms use an arbitrary tie-breaking heuristic. Without loss of generality, we can assume that greedy chooses ω_1. In the second iteration, greedy would pick ω_2 (because $R_1 = R_2 > R_3$) and terminate due to the budget constraint. Therefore, the sensor subset selected by the greedy algorithm is $\Gamma = \{\omega_1, \omega_2\}$. By Lemma 1 and Equation (6), the infinite-horizon expected reward of the greedy algorithm is

$$V_\Gamma^{gre} = \frac{R_1}{1-\gamma} + \frac{R_2}{1-\gamma} = \frac{2R_2}{1-\gamma}. \qquad (7)$$

Consider the following selection of sensors for the MOMDP-SS instance: $\Gamma = \{\omega_2, \omega_3\}$. By selecting sensors ω_2 and ω_3, the states s_2 and s_3 can be measured. Since the state s_4 is a function of states s_2 and s_3, the agent can estimate the state s_4 (measure it indirectly) via sensors ω_2 and ω_3. As a result, both s_2 and s_4 will be measurable. By Lemma 1 and Equation (6), the infinite-horizon expected reward is

$$V_\Gamma^{opt} = \frac{R_2}{1-\gamma} + \frac{R_4}{1-\gamma} = \frac{R_2 + R_4}{1-\gamma}. \qquad (8)$$

By Equations (7) and (8), we have,

$$r_{gre}(\Gamma) = \frac{2R_2}{R_2 + R_4} = \frac{2}{1 + \frac{R_4}{R_2}} = \frac{2}{1+h}. \qquad (9)$$

Therefore, $\lim_{h\to\infty} r_{gre}(\Gamma) = 0$. ■

Remark 1: Proposition 1 means that if we make R_4 arbitrarily larger than R_2, the expected return obtained by greedy can get arbitrarily small compared to the expected value obtained by optimal selection of sensors. This is because greedy picks sensors ω_1 and ω_2 due to its myopic behavior. It does not consider the fact that, in spite of R_3 being the least reward, selecting ω_3 would eventually lead to an indirect measurement of s_4 having the highest reward R_4. An expected consequence of the arbitrarily poor performance of the greedy algorithm is that the optimal value function of the MOMDP is not necessarily submodular in the set of sensors selected. For completeness, we now state this explicitly.

A. Lack of Submodularity of the Value Function

Submodular set functions have a property of diminishing returns, which makes them suitable for approximation algorithms.

Definition 1 (Submodular Function): A submodular function over a finite set Ω is a set function $f : 2^\Omega \to \mathbb{R}$, which satisfies $f(X \cup \{x\}) - f(X) \geq f(Y \cup \{x\}) - f(Y)$, for all $X, Y \subseteq \Omega$ with $X \subseteq Y$ and for all $x \in \Omega \setminus Y$.

Greedy algorithms have a guarantee of producing at least $(1 - 1/e)$ times the optimal (maximal) solution for monotonically increasing, submodular and non-negative (or normalized) objective functions. It is easy to verify that the value function of MOMDP-SS is a monotonically increasing, non-negative function. Since we showed that a greedy algorithm can perform arbitrarily poorly for MOMDP-SS, we have the following result.

Corollary 1: The value function of the MOMDP-SS problem is not necessarily submodular in the set of sensors (Γ) selected. ■

VI. Experiments

In the previous section, we showed that the greedy algorithm for MOMDP-SS can perform arbitrarily poorly. However, this arbitrary poor performance was for a specific instance of MOMDP-SS, and in general, greedy might not actually perform poorly for all instances. In this section, we evaluate the greedy algorithm for several randomly generated instances of MOMDP-SS. Since MOMDPs are a special class of POMDPs, we use the `SolvePOMDP` software package [24], a Java program that can solve POMDPs optimally using incremental pruning [2] combined with state-of-the-art vector pruning methods [25]. We run the *exact algorithm* in this package to compute the optimal solution for infinite-horizon cases by setting a value function tolerance $\eta = 1 \times 10^{-6}$ as a stopping criterion. We generate 20 instances of MOMDP-SS with each instance having $|\mathcal{S}| = 16$ states (4 binary state variables) and $|\mathcal{A}| = 16$ actions. The transition function $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ for each starting state $s \in \mathcal{S}$, action $a \in \mathcal{A}$ and ending state $s' \in \mathcal{S}$ is a value uniformly sampled over a probability simplex ($\Delta_{|\mathcal{S}|}$). The rewards $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}_{>0}$ for each state-action pair (s, a) are randomly sampled from $\texttt{abs}(\mathcal{N}(0, \sigma))$, with $\sigma \sim \texttt{uniform random}(0, 10)$. We consider a set of 4 noise-less sensors $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4\}$,

which can measure the states (s_1, s_2, s_3, s_4), respectively. We consider uniform sensor costs $c_1 = c_2 = c_3 = c_4 = 1$ and a sensor budget of $C = 2$. We apply a brute-force technique by generating all possible sensor subsets $\Gamma \subset \Omega$ of size $|\Gamma| = 2$, to compute the optimal set of sensors Γ^* and compute the optimal return V_Γ^{opt} using the solver. We run Algorithm 1 for each of these instances to compute the return V_Γ^{gre}.

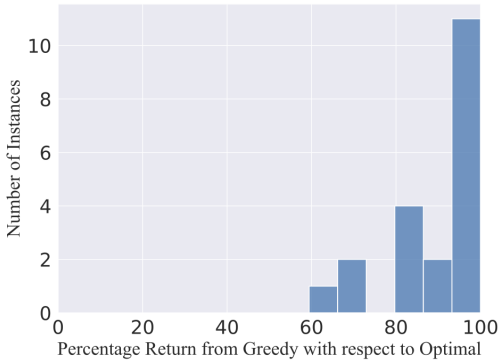

Fig. 2. Empirical distribution of percentage of expected infinite-horizon return from greedy sensor selection with respect to that of optimal sensor selection for 20 randomly generated MOMDP-SS instances.

It can be seen from Fig. 2 that greedy shows near-optimal performance for many instances, with an average of 90.19% of the optimal. We conclude from these results that, in spite of arbitrarily poor performance for some instances, in practice, greedy may be able to achieve near-optimal solutions.

VII. Conclusions

In this paper, we studied the budgeted design-time sensor selection problem for MOMDPs, and proved that it is NP-hard in general. We analyzed the performance of greedy algorithms for sensor selection, and explicitly provided an example showing that greedy algorithms can perform arbitrarily poorly on some instances. Thus, one cannot provide theoretical guarantees for the performance of the greedy algorithm. Further, we showed the lack of submodularity of the value function of the MOMDP, to conclude that this problem is more difficult than other variants of sensor selection problem that have submodular objectives. Finally, we demonstrated the empirical performance of the greedy algorithm for randomly generated MOMDP-SS instances and concluded that although greedy performed arbitrarily poorly for some instances, it provided near-optimal solutions for many instances. Future works on extending the results to MOMDPs over finite time horizons, and identifying classes of systems that admit near-optimal approximation algorithms are of interest.

References

[1] G. E. Monahan, "State of the art—a survey of partially observable Markov decision processes: Theory, models, and algorithms," *Management science*, vol. 28, no. 1, pp. 1–16, 1982.

[2] A. R. Cassandra, M. L. Littman, and N. L. Zhang, "Incremental pruning: A simple, fast, exact method for partially observable Markov decision processes," *arXiv preprint arXiv:1302.1525*, 2013.

[3] J. Pineau, G. Gordon, and S. Thrun, "Point-based value iteration: An anytime algorithm for POMDPs," in *IJCAI*, vol. 3, 2003.

[4] H. Kurniawati, D. Hsu, and W. S. Lee, "SARSOP: Efficient point-based POMDP planning by approximating optimally reachable belief spaces." in *Robotics: Science and systems*, 2008.

[5] M. Araya-López, V. Thomas, O. Buffet, and F. Charpillet, "A closer look at MOMDPs," in *22nd International Conference on Tools with Artificial Intelligence*, vol. 2. IEEE, 2010, pp. 197–204.

[6] H. Zhang, R. Ayoub, and S. Sundaram, "Sensor selection for Kalman filtering of linear dynamical systems: Complexity, limitations and greedy algorithms," *Automatica*, vol. 78, pp. 202–210, 2017.

[7] L. Ye, N. Woodford, S. Roy, and S. Sundaram, "On the complexity and approximability of optimal sensor selection and attack for Kalman filtering," *IEEE Transactions on Automatic Control*, vol. 66, no. 5, pp. 2146–2161, 2020.

[8] C. S. Laurent and R. V. Cowlagi, "Coupled sensor configuration and path-planning in unknown static environments," in *American Control Conference (ACC)*. IEEE, 2021, pp. 1535–1540.

[9] A. W. Mahoney, T. L. Bruns, P. J. Swaney, and R. J. Webster, "On the inseparable nature of sensor selection, sensor placement, and state estimation for continuum robots or "where to put your sensors and how to use them"," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 4472–4478.

[10] H.-P. Chiu, X. S. Zhou, L. Carlone, F. Dellaert, S. Samarasekera, and R. Kumar, "Constrained optimal selection for multi-sensor robot navigation using plug-and-play factor graphs," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 663–670.

[11] W. Li, F. Zhou, K. R. Chowdhury, and W. Meleis, "Qtcp: Adaptive congestion control with reinforcement learning," *IEEE Transactions on Network Science and Engineering*, vol. 6, no. 3, pp. 445–458, 2018.

[12] M. Duggan, K. Flesk, J. Duggan, E. Howley, and E. Barrett, "A reinforcement learning approach for dynamic selection of virtual machines in cloud data centres," in *International Conference on Innovative Computing Technology (INTECH)*. IEEE, 2016, pp. 92–97.

[13] C. Tessler, Y. Shpigelman, G. Dalal, A. Mandelbaum, D. Haritan Kazakov, B. Fuhrer, G. Chechik, and S. Mannor, "Reinforcement learning for datacenter congestion control," *ACM SIGMETRICS Performance Evaluation Review*, vol. 49, no. 2, pp. 43–46, 2022.

[14] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of Markov decision processes," *Mathematics of operations research*, vol. 12, no. 3, pp. 441–450, 1987.

[15] M. Ghasemi and U. Topcu, "Online active perception for partially observable Markov decision processes with limited budget," in *Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 6169–6174.

[16] V. Krishnamurthy and D. V. Djonin, "Structured threshold policies for dynamic sensor scheduling—a partially observed Markov decision process approach," *IEEE Transactions on Signal Processing*, vol. 55, no. 10, pp. 4938–4957, 2007.

[17] S. Ji, R. Parr, and L. Carin, "Nonmyopic multiaspect sensing with partially observable Markov decision processes," *IEEE Transactions on Signal Processing*, vol. 55, no. 6, pp. 2720–2730, 2007.

[18] A. Hashemi, H. Vikalo, and G. de Veciana, "On the benefits of progressively increasing sampling sizes in stochastic greedy weak submodular maximization," *IEEE Transactions on Signal Processing*, vol. 70, pp. 3978–3992, 2022.

[19] A. Hashemi, M. Ghasemi, H. Vikalo, and U. Topcu, "Randomized greedy sensor selection: Leveraging weak submodularity," *IEEE Transactions on Automatic Control*, vol. 66, no. 1, pp. 199–212, 2020.

[20] R. Khanna, E. Elenberg, A. Dimakis, S. Negahban, and J. Ghosh, "Scalable greedy feature selection via weak submodularity," in *Artificial Intelligence and Statistics*. PMLR, 2017, pp. 1560–1568.

[21] M. R. Garey, "A Guide to the Theory of NP-Completeness," *Computers and Intractability*, 1979.

[22] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions—i," *Mathematical programming*, vol. 14, no. 1, pp. 265–294, 1978.

[23] A. Krause and C. Guestrin, "Near-optimal observation selection using submodular functions," in *AAAI*, vol. 7, 2007, pp. 1650–1654.

[24] "SolvePOMDP-a java program that solves Partially Observable Markov Decision Processes." https://www.erwinwalraven.nl/solvepomdp/.

[25] E. Walraven and M. Spaan, "Accelerated vector pruning for optimal POMDP solvers," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, 2017.