2019 IEEE 58th Conference on Decision and Control (CDC)
Palais des Congrès et des Expositions Nice Acropolis
Nice, France, December 11-13, 2019

Incentive Design for Temporal Logic Objectives

Yagiz Savas, Vijay Gupta, Melkior Ornik, Lillian J. Ratliff, Ufuk Topcu

Abstract— **We study the problem of designing an optimal sequence of incentives that a principal should offer to an agent so that the agent's optimal behavior under the incentives realizes the principal's objective expressed as a temporal logic formula. We consider an agent with a finite decision horizon, and model its behavior as a Markov decision process (MDP). Under certain assumptions, we present a polynomial-time algorithm to synthesize an incentive sequence that minimizes the cost to the principal. We show that if the underlying MDP has only deterministic transitions, the principal can hide its objective from the agent and still realize the desired behavior through incentives. On the other hand, an MDP with stochastic transitions may require the principal to share its objective with the agent. Finally, we demonstrate the proposed method in motion planning examples where a principal changes the optimal trajectory of an agent by providing incentives.**

I. Introduction

Consider a scenario in which a principal provides incentives to an agent so that the optimal behavior of the agent under the provided incentives satisfies the principal's objective. If the principal had enough resources to provide arbitrarily large incentives, it would be straightforward to obtain the desired agent behavior. However, since the resources are typically limited in practice, it is important to establish the minimum amount of incentives that leads to the desired behavior. In this paper, we study the problem of designing a sequence of incentives that minimizes the cost to the principal while guaranteeing the realization of its objective by the agent.

We model the behavior of the agent as a Markov decision process (MDP) [1], and assume that the agent's objective is to maximize its expected total reward at the end of a finite planning horizon. Although each planning horizon is finite, the agent plans its future decisions infinitely many times. Examples of such an agent can be a person who plans her schedule on a weekly basis or an autonomous system with a limited computational power which plans its route by considering only a subset of all possible environment states.

The principal's objective is described by a syntactically co-safe linear temporal logic (LTL) formula. LTL specifications are widely used to describe complex tasks for autonomous robots [2], design security protocols [3] and check the reliability of software [4]. For example, in a navigation scenario, syntactically co-safe LTL formulae allow one to specify tasks such as liveness (eventually visit the region A) or priority (first visit the region A and then B).

We assume that the principal is aware of the agent's reward function and the length of its planning horizon. In many real-world applications, the decision horizon and the reward structure of an agent can be known or at least inferred through observations. For example, a manufacturing company is generally interested in maximizing its profit at the end of a fiscal year, and an autonomous car aims to reach its destination within certain time interval.

From a practical point of view, an interesting question is whether an adversarial principal can convince an agent to satisfy its objective through incentives. In such a scenario, if the agent knows the principal's objective explicitly, it will reject the provided incentives because the resulting behavior under the incentives will serve to the benefit of the enemy. However, if the principal can design an incentive sequence without sharing its objective with the agent, then the incentives may lead to the desired agent behavior. Therefore, it is important to establish conditions under which the principal can actually hide its objective from the agent.

The contributions of this paper are threefold. First, we present an algorithm, based on a series of linear optimization problems, to synthesize a sequence of incentives that minimizes the cost to the principal while ensuring that the optimal agent behavior under the provided incentives satisfies a syntactically co-safe LTL formula with maximum probability. Second, we present an example scenario where the principal has to share its objective with the agent to induce the desired behavior. The provided example illustrates that, unless the principal states its objective explicitly, inducing the desired agent behavior through incentives is, in general, not possible. Third, we provide sufficient conditions on the structure of the MDP and the length of the agent's decision horizon under which there exists an optimal incentive sequence that allows the principal to hide its objective from the agent.

Related work. The problem of obtaining desired agent behavior through a sequence of incentives has been extensively studied in the literature. In [5] and [6], the authors present methods to design incentive sequences with *limited* resources that maximize the value of the principal's objective function. They employ techniques from the inverse reinforcement learning literature, and prove the NP-hardness of the problem [5]. The work in [7] provides a polynomial-time algorithm to synthesize minimum incentives for inducing a *specific* agent policy. Reference [8] considers a bandit model and

This work was supported in part by the grants AFRL # FA9550-19-1-0169, DARPA # D19AP00004, and AFRL # FA8650-15-C-2546.

Y. Savas and U. Topcu are with the Department of Aerospace Engineering, University of Texas at Austin, TX, USA. E-mail: {yagiz.savas, utopcu}@utexas.edu

V. Gupta is with the Department of Electrical Engineering, University of Notre Dame, IN, USA. E-mail: vgupta@nd.edu

M. Ornik is with the Department of Aerospace Engineering and the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, IL, USA. E-mail: mornik@illinois.edu

L. J. Ratliff is with the Department of Electrical Engineering, University of Washington, WA, USA. E-mail: ratliffl@uw.edu

978-1-7281-1398-2/19/$31.00 ©2019 IEEE

presents methods to induce desired agent actions under different constraints on the incentives. Although it is quite different from the problem considered in this paper, the design of feasible incentives that align the objectives of an agent and a principal is discussed in [9] from a control-theoretic perspective. Unlike the references mentioned above, in this paper, we consider the problem of designing minimum incentives that maximize the value of the principal's objective expressed as a temporal logic formula. We also note that establishing the complexity of the design problem considered in this paper was mentioned as an open problem in [5].

II. Preliminaries

For a set S, we denote its power set and cardinality by 2^S and $|S|$, respectively. Additionally, $\mathbb{N}{=}\{1,2,\ldots\}$, $\mathbb{N}_0{=}\{0,1,2,\ldots\}$ and $\mathbb{R}_{\geq 0}{=}[0,\infty)$.

A. Markov Decision Processes

Definition 1: A *Markov decision process* (MDP) is a tuple $\mathcal{M}{=}(S,s_0,\mathcal{A},\mathcal{P},\mathcal{AP},\mathcal{L},\mathcal{R})$ where S is a finite set of states, $s_0{\in}S$ is an initial state, $\mathcal{A}$ is a finite set of actions, $\mathcal{P}{:}S{\times}\mathcal{A}{\times}S{\to}[0,1]$ is a transition function such that $\sum_{s'\in S}\mathcal{P}(s,a,s'){=}1$ for all $s{\in}S$ and $a{\in}\mathcal{A}(s)$ where $\mathcal{A}(s)$ denote the available actions in s, $\mathcal{AP}$ is a set of atomic propositions, $\mathcal{L}{:}S{\to}2^{\mathcal{AP}}$ is a function that labels each state with a subset of atomic propositions, and $\mathcal{R}{:}S{\times}\mathcal{A}{\to}\mathbb{R}$ is a reward function.

We denote the transition probability $\mathcal{P}(s,a,s')$ by $\mathcal{P}_{s,a,s'}$.

Definition 2: For an MDP $\mathcal{M}$, a *decision rule* $d{:}S{\times}\mathcal{A}{\to}[0,1]$ is a function such that $\sum_{a\in\mathcal{A}(s)}d(s,a){=}1$ for all $s{\in}S$. A decision rule d is said to be *deterministic* if for all $s{\in}S$ there exists $a{\in}\mathcal{A}(s)$ such that $d(s,a){=}1$, and *randomized* otherwise. For an MDP $\mathcal{M}$, we denote the set of all (deterministic) decision rules by $(\mathcal{D}^D(\mathcal{M}))$ $\mathcal{D}(\mathcal{M})$.

For an MDP $\mathcal{M}$, a decision-maker, i.e., an agent, chooses a decision rule $d{\in}\mathcal{D}(\mathcal{M})$ at each *stage*.

Definition 3: An N-stage *policy* for an MDP $\mathcal{M}$ is a sequence $\pi{=}(d_1,d_2,\ldots,d_N)$ where $N{\leq}\infty$ and $d_t{\in}\mathcal{D}(\mathcal{M})$ for all $t{\leq}N$. A *stationary* policy is a policy such that $d_t{=}d_1$ for all $t{\leq}N$. A policy is said to be *deterministic* if $d_t{\in}\mathcal{D}^D(\mathcal{M})$ for all t, and *randomized* otherwise. For an MDP $\mathcal{M}$, we denote the set of all N-stage policies by $\Pi_N(\mathcal{M})$. For notational simplicity, we denote the set of ∞-stage policies by $\Pi(\mathcal{M})$.

For an MDP $\mathcal{M}$ and a policy $\pi{\in}\Pi(\mathcal{M})$, let $\mu_t^\pi(s,a)$ be the joint probability of being in state $s{\in}S$ and taking the action $a{\in}\mathcal{A}(s)$ at stage t, which is uniquely determined through the recursive formula

$$\mu_{t+1}^\pi(s',a') = \sum_{s\in S}\sum_{a\in\mathcal{A}(s)}\mathcal{P}_{s,a,s'}\mu_t^\pi(s,a)d_{t+1}(s',a') \tag{1}$$

where $\mu_1^\pi(s,a){=}d_1(s,a)\mu_0(s)$ and $\mu_0{:}S{\to}\{0,1\}$ is a function such that $\mu_0(s_0){=}1$ and $\mu_0(s){=}0$ for all $s{\in}S\backslash\{s_0\}$.

Definition 4: For an MDP $\mathcal{M}$ and a policy $\pi{\in}\Pi(\mathcal{M})$, the *expected residence time* in a state-action pair (s,a) is

$$\xi^\pi(s,a) := \sum_{t=1}^{\infty}\mu_t^\pi(s,a). \tag{2}$$

An infinite sequence $\varrho^\pi{=}s_0s_1s_2\ldots$ of states generated in $\mathcal{M}$ under a policy $\pi{\in}\Pi(\mathcal{M})$, which starts from the initial state s_0 and satisfies $\sum_{a_t\in\mathcal{A}(s_t)}d_k(s_t,a_t)\mathcal{P}_{s_t,a_t,s_{t+1}}{>}0$ for all $t{\geq}0$, is called a *path*. Any finite prefix of ϱ^π a finite path fragment. We define the set of all paths and finite path fragments in $\mathcal{M}$ under the policy π by $Paths^\pi(\mathcal{M})$ and $Paths_{fin}^\pi(\mathcal{M})$, respectively. We use the standard probability measure over the outcome set $Paths^\pi(\mathcal{M})$ [10, Chapter 10].

Definition 5: An *incentive sequence* Γ for an MDP $\mathcal{M}$ is a sequence $(\gamma_1,\gamma_2,\ldots)$ of functions where $\gamma_t{:}S{\times}\mathcal{A}{\to}\mathbb{R}_{\geq 0}$. For an MDP $\mathcal{M}$, we denote the set of all admissible incentive sequences by $\Theta(\mathcal{M})$.

B. Linear temporal logic

We consider syntactically co-safe linear temporal logic (scLTL) formulae to specify tasks and refer the reader to [10], [11] for the syntax and semantics of scLTL.

An scLTL formula is built up from a set $\mathcal{AP}$ of atomic propositions, logical connectives such as conjunction ($\wedge$) and negation ($\neg$), and temporal modal operators such as until ($\mathcal{U}$) and eventually ($\Diamond$). An infinite sequence of subsets of $\mathcal{AP}$ defines an infinite *word*, and an scLTL formula is interpreted over infinite words on $2^{\mathcal{AP}}$. We denote by $w{\models}\varphi$ that a word $w{=}w_0w_1w_2\ldots$ satisfies an scLTL formula φ.

For an MDP $\mathcal{M}$ under a policy π, a path $\varrho^\pi{=}s_0s_1\ldots$ generates a word $w{=}w_0w_1\ldots$ where $w_k{=}\mathcal{L}(s_k)$ for all $k{\geq}0$. With a slight abuse of notation, we use $\mathcal{L}(\varrho^\pi)$ to denote the word generated by ϱ^π. For an scLTL formula φ, the set $\{\varrho^\pi{\in}Paths^\pi(\mathcal{M}){:}\mathcal{L}(\varrho^\pi){\models}\varphi\}$ is measurable [10]. We define

$$\text{Pr}^\pi_{\mathcal{M}}(s_0\models\varphi) := \text{Pr}^\pi_{\mathcal{M}}\{\varrho^\pi\in Paths^\pi(\mathcal{M}):\mathcal{L}(\varrho^\pi)\models\varphi\}$$

as the probability of satisfying the scLTL formula φ for an MDP $\mathcal{M}$ under the policy $\pi{\in}\Pi(\mathcal{M})$.

III. Problem Statement

We consider an *agent* whose behavior is modeled as an MDP $\mathcal{M}$, and a *principal* that provides the agent a sequence $\Gamma{\in}\Theta(\mathcal{M})$ of incentives.

The agent's objective is to maximize its expected total reward after N stages. However, since the incentive sequence offered by the principal may be non-stationary, the agent computes an N-stage policy in every N stages. A graphical illustration of the agent's planning method is shown in Fig. 1. Formally, let $N{\in}\mathbb{N}$ be a constant, and $\mathcal{R}(S_t,A_t)$ and $\gamma_t(S_t,A_t)$ be the random reward and incentive received in stage $t{\leq}N$. Additionally, let $J{:=}(J_0,J_1,\ldots)$ be a sequence of objective functions where $J_k{:}\Pi_N(\mathcal{M}){\times}\Theta(\mathcal{M}){\to}\mathbb{R}^{|S|}$ is defined as

$$J_k(\pi,\Gamma)(s) := \mathbb{E}_s^\pi\Big[\sum_{t=1}^{N}(\mathcal{R}(S_t,A_t)+\gamma_{kN+t}(S_t,A_t))\Big]$$

for all $s{\in}S$. Here, the expectation is taken over the finite path fragments that are generated by the policy $\pi{\in}\Pi_N(\mathcal{M})$ and have the initial state s. Then, for a given sequence $\Gamma{\in}\Theta(\mathcal{M})$

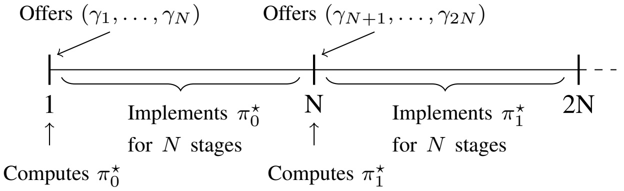

Fig. 1: An illustration of the incentive implementation and the agent's decision-making process. The principal offers incentives for the next N stages. After receiving the incentive offers, the agent computes and implements its optimal decisions for the next N stages.

of incentives, the agent's optimal ∞-stage policy is given by $\pi^\star{:=}(\pi_0^\star, \pi_1^\star, \dots)$, where $\pi_k^\star$ is such that

$$\pi_k^\star \in \arg \max_{\pi\in\Pi_N(\mathcal{M})} J_k(\pi,\Gamma)(s) \tag{3}$$

for all $s{\in}S$ and $k{\in}\mathbb{N}_0$. Note that the agent's policy $\pi_k^\star$ maximizes the total reward starting from any $s{\in}S$.

The principal's objective is to design a sequence of incentives that incurs the minimum cost while ensuring that the agent's optimal policy, under the provided incentives, satisfies an scLTL formula φ with maximum probability. We make the following assumptions:

(i) Agent's reward function $\mathcal{R}$ is known to the principal.
(ii) Agent's decision horizon N is known to the principal.
(iii) The principal pays the offered incentives if and only if the agent takes the incentivized action.

Then, the optimization problem that we are interested in solving is the following:

$$\min_{\Gamma\in\Theta(\mathcal{M})} \ \mathbb{E}_{s_0}^{\pi^\star}\Big[\sum_{t=1}^{\infty}\gamma_t(s,a)\Big] \tag{4a}$$

$$\text{subject to: } \pi^\star = (\pi_0^\star, \pi_1^\star, \dots) \tag{4b}$$

$$\pi_k^\star \in \arg\max_{\pi\in\Pi_N(\mathcal{M})} J_k(\pi,\Gamma)(s) \ \forall s\in S, \forall k\in\mathbb{N}_0 \tag{4c}$$

$$\text{Pr}_{\mathcal{M}}^{\pi^\star}(s_0 \models \varphi) = \max_{\pi\in\Pi(\mathcal{M})} \text{Pr}_{\mathcal{M}}^{\pi}(s_0\models\varphi) \tag{4d}$$

where $\Gamma{=}(\gamma_1, \gamma_2, \dots)$. Note in (4a)-(4d) that the principal's objective is to minimize its cost over infinite horizon whereas the agent aims to maximize its total reward at the end of each finite decision horizon.

IV. The Synthesis of Incentive Sequences

In this section, we provide a method to synthesize an incentive sequence that solves the problem (4a)-(4d). For simplicity, we restrict our attention to reachability specifications, i.e., $\varphi{=}\Diamond p$ where $p{\in}\mathcal{AP}$. The synthesis of incentive sequences for general scLTL specifications is discussed in Section VI.

We first partition the states into three disjoint sets as follows. Let $B{\subseteq}S$ be the set of all states such that $\{p\}{\subseteq}\mathcal{L}(s)$, i.e., the set of states that the principal wants the agent to reach, and $S_0{\subseteq}S$ be the set of states that have zero probability of reaching the states in B under any policy. More precisely, $s{\in}S_0$ if $\text{Pr}_{\mathcal{M}}^{\pi}(s\models\Diamond p){=}0$ for all $\pi{\in}\Pi(\mathcal{M})$. Finally, we let $S_r{=}S\backslash(B\cup S_0)$ be the set of all states that are not in B and have nonzero probability of reaching a state in B under some policy. These sets can be found in time polynomial in the size of the MDP using graph search algorithms [10, Chapter 10].

The agent's initial state $s_0{\in}S$ can belong to either B, S_0 or S_r. However, we only consider the case $s_0{\in}S_r$, since otherwise the optimal incentive sequence is trivially $\gamma_t(s,a){=}0$ for all $t{\in}\mathbb{N}$.

A. The cost of control

Recall that the agent's first objective function $J_0{:}\Pi_N(\mathcal{M}){\times}\Theta(\mathcal{M}){\to}\mathbb{R}^{|S|}$ is

$$J_0(\pi,\Gamma)(s) = \mathbb{E}_s^{\pi}\Big[\sum_{t=1}^{N}(\mathcal{R}(S_t,\mathcal{A}_t)+\gamma_t(S_t,\mathcal{A}_t))\Big]$$

for all $s{\in}S$. Let $V_n{:}S{\to}\mathbb{R}$ be the agent's *value function* at stage n such that

$$V_n(s) := \max_{\pi\in\Pi_N(\mathcal{M})} \mathbb{E}_s^{\pi}\Big[\sum_{t=n}^{N}(\mathcal{R}(S_t,\mathcal{A}_t)+\gamma_t(S_t,\mathcal{A}_t))\Big]$$

for all $s{\in}S$, where the expectation is taken over the paths that occupy s at stage n. Then, we have the recursive formula

$$V_n(s) = \max_{a\in\mathcal{A}(s)} \mathcal{R}(s,a) + \gamma_n(s,a) + \sum_{s'\in S}\mathcal{P}_{s,a,s'}V_{n+1}(s')$$

for all $1{\leq}n{\leq}N$, where $V_{N+1}(s){=}0$ for all $s{\in}S$. Let $Q_n{:}S{\times}\mathcal{A}{\to}\mathbb{R}$ be the agent's *Q-function* at stage n such that

$$Q_n(s,a) := \mathcal{R}(s,a) + \gamma_n(s,a) + \sum_{s'\in S}\mathcal{P}_{s,a,s'}V_{n+1}(s').$$

By the principle of optimality [1], [12], the agent's optimal policy $\pi_0^\star{=}(d_1^\star, d_2^\star, \dots, d_N^\star)$ is such that, for all $1{\leq}n{\leq}N$, $d_n(s,a'){>}0$ only if

$$a' \in \arg\max_{a\in\mathcal{A}(s)} Q_n(s,a).$$

We recursively define

$$\overline{Q}_n(s,a) := \mathcal{R}(s,a) + \sum_{s'\in S}\mathcal{P}_{s,a,s'}\overline{V}_{n+1}(s'), \tag{5}$$

$$\overline{V}_n(s) := \max_{a\in\mathcal{A}(s)} \overline{Q}_n(s,a), \tag{6}$$

for all $s{\in}S$ and $a{\in}\mathcal{A}(s)$. For a given $\epsilon{\geq}0$, we finally define a real-valued function $\phi_t^\epsilon{:}S\times\mathcal{A}{\to}\mathbb{R}_{\geq 0}$ such that

$$\phi_n^\epsilon(s,a) := \begin{cases}\overline{V}_n(s) - \overline{Q}_n(s,a) + \epsilon & \text{if } s\in S_r, \ a\in\mathcal{A}(s)\\ 0 & \text{otherwise.}\end{cases}$$

For an arbitrarily small $\epsilon{>}0$, the value of $\phi_n^\epsilon(s,a)$, referred as *the cost of control* for the state-action pair (s,a), is the minimum incentive that should be offered to the agent in order to make the action $a{\in}\mathcal{A}(s)$ uniquely optimal at stage t. It is worth noting that although the cost of control $\phi_n^\epsilon(s,a)$ depends on the stage number n, it is independent of the objective number, i.e., it is the same for all J_k. This is because the agent's reward function $\mathcal{R}$ is stationary, and therefore, $\overline{V}_n(s)$ and $\overline{Q}_n(s,a)$ do not change with the objective number k as can be seen from (5)-(6).

B. An $\overline{\epsilon}$-optimal incentive design

To synthesize an incentive sequence that minimizes the cost to the principal, we should specify the actions to be incentivized by the principal at each state for each stage. To this aim, we modify the MDP $\mathcal{M}$ by considering the agent's decision horizon N as another dimension in the state-space.

Definition 6: For an MDP $\mathcal{M}$ and $T{=}\{1,2,\ldots,N\}$, the *expanded MDP* is a tuple $\overline{\mathcal{M}}{=}(\overline{S},\overline{s}_0,\mathcal{A},\overline{\mathcal{P}},\mathcal{AP},\overline{\mathcal{L}},\overline{\mathcal{R}})$ where

- $\overline{S}{=}S\times T$,
- $\overline{s}_0{=}(s_0,1)$ is the initial state,
- $\overline{\mathcal{P}}{:}\overline{S}\times\mathcal{A}\times\overline{S}{\to}[0,1]$ is such that

$$\overline{\mathcal{P}}_{(s,n),a,(s',n')} = \begin{cases}\mathcal{P}_{s,a,s'} & \text{if } 1\leq n\leq N-1 \text{ and } n'=n+1\\ \mathcal{P}_{s,a,s'} & \text{if } n=N \text{ and } n'=1\\ 0 & \text{otherwise,}\end{cases}$$

- $\overline{\mathcal{L}}{:}\overline{S}{\to}2^{\mathcal{AP}}$ is such that $\overline{\mathcal{L}}((s,n)){=}\mathcal{L}(s)$ for all $s{\in}S$ and for all $n{\in}T$,
- $\overline{\mathcal{R}}{:}\overline{S}\times\mathcal{A}{\to}\mathbb{R}_{\geq 0}$ is such that $\overline{\mathcal{R}}((s,n),a){=}\mathcal{R}(s,a)$ for all $s{\in}S$, for all $a{\in}\mathcal{A}$, and for all $n{\in}T$,

and $\mathcal{A}$, and $\mathcal{AP}$ are as defined for $\mathcal{M}$.

We note that the transition function $\overline{\mathcal{P}}$ is defined such that the agent's initial state while computing the k-th N stage policy is the state occupied by the agent at $kN{+}1$-st stage on the expanded MDP.

Let $\overline{B}{\cup}\overline{S}_0{\cup}\overline{S}_r$ be the partition of the states of $\overline{\mathcal{M}}$ such that if $s{\in}B$, then $(s,n){\in}\overline{B}$ for all $n{\in}T$, and the sets $\overline{S}_0$ and $\overline{S}_r$ are defined similarly. Then, the principal's objective on $\overline{\mathcal{M}}$ is to induce an agent policy that *eventually* reaches the set $\overline{B}$ with maximum probability.

To synthesize an incentive sequence under which the optimal agent policy satisfies the desired property, we modify the expanded MDP $\overline{\mathcal{M}}$ by making its states $s{\in}\overline{B}{\cup}\overline{S}_0$ absorbing, and denote the resulting MDP by $\overline{\mathcal{M}}'$. Then, for a given constant $\epsilon{\geq}0$, we define the cost of control for a state-action pair on $\overline{\mathcal{M}}'$ through the function $\underline{\phi}^{\epsilon}{:}\overline{S}{\times}\mathcal{A}{\to}\mathbb{R}_{\geq 0}$ such that

$$\underline{\phi}^{\epsilon}((s,n),a) := \begin{cases}\overline{V}_n(s)-\overline{Q}_n(s,a)+\epsilon & \text{if } s\in\overline{S}_r, a\in\mathcal{A}(s)\\ 0 & \text{otherwise.}\end{cases}$$

Let $\Xi(\overline{\mathcal{M}}'){\subseteq}\Pi(\overline{\mathcal{M}}')$ be a subset of the set of ∞-stage policies such that $\pi'{\in}\Xi(\overline{\mathcal{M}}')$ if and only if

$$\pi' \in \arg\max_{\pi\in\Pi(\overline{\mathcal{M}}')} \Pr{}^{\pi}(\overline{s}_0\models\varphi), \tag{7}$$

and for $\epsilon{\geq}0$, $f_{\epsilon}{:}\Xi(\overline{\mathcal{M}}'){\to}\mathbb{R}$ be a function such that

$$f_{\epsilon}(\pi) := \mathbb{E}^{\pi}_{\overline{s}_0}\Big[\sum_{t=1}^{\infty}\underline{\phi}^{\epsilon}(S_t,\mathcal{A}_t)\Big]. \tag{8}$$

Then, for an arbitrarily small $\overline{\epsilon}{>}0$, an $\overline{\epsilon}$-optimal incentive sequence can be designed in two steps as follows.

Step 1: Compute $\overline{V}_n(s)$ and $\overline{Q}_n(s,a)$ given in (5)-(6), and construct the cost of control function $\underline{\phi}^{\epsilon}$. Then, for the modified expanded MDP $\overline{\mathcal{M}}'$, compute a *stationary deterministic* policy $\widetilde{\pi}{=}(\widetilde{d},\widetilde{d},\ldots)$ such that

$$\widetilde{\pi} \in \arg\min_{\pi\in\Xi(\overline{\mathcal{M}}')} f_{\epsilon}(\pi). \tag{9}$$

Step 2: Let $\varrho^{\pi}{\in}Paths^{\pi}(\mathcal{M})$ be the path followed by the agent. At stage kN where N is the agent's decision horizon and $k{\in}\mathbb{N}_0$, provide the agent with the incentive sequence $(\widetilde{\gamma}_1,\widetilde{\gamma}_2,\ldots,\widetilde{\gamma}_N)$ such that

- if $\varrho^{\pi}[n]{\notin}B\cup S_0$ for all $n{\leq}kN$,

$$\widetilde{\gamma}_n(s,a) := \begin{cases}\underline{\phi}^{\epsilon}((s,n),a) & \text{if } s\in S_r \text{ and } \widetilde{d}((s,n),a)>0,\\ \epsilon & \text{if } s\notin S_r \text{ and } \widetilde{d}((s,n),a)>0,\\ 0 & \text{otherwise,}\end{cases} \tag{10}$$

- $\widetilde{\gamma}_n(s,a){:=}0$ otherwise.

Under the proposed incentive sequence (10), the agent's value function V_n satisfies $V_n(s){=}\overline{V}_n(s){+}(N+1-n)\epsilon$ for all $s{\in}S$, $n{\leq}N$. Additionally, if $\varrho^{\pi}[n]{\notin}B{\cup}S_0$ for all $n{\leq}kN$, then for all $s{\in}S$, $\widetilde{d}((s,n),a){>}0$ implies that the agent's Q-function satisfies

$$\begin{aligned}Q_n(s,a) &= \mathcal{R}(s,a)+\widetilde{\gamma}_n(s,a)+\sum_{s'\in S}\mathcal{P}_{s,a,s'}V_{n+1}(s')\\ &= \widetilde{\gamma}_n(s,a)+\overline{Q}_n(s,a)+(N-n)\epsilon\\ &= (N+1-n)\epsilon+\overline{V}_n(s)\\ &> (N-n)\epsilon+\overline{V}_n(s) = \max_{a'\in\mathcal{A}(s)\backslash\{a\}} Q_n(s,a').\end{aligned}$$

Consequently, the agent is guaranteed to take the incentivitized actions at each stage until reaching the set $B{\cup}S_0$.

We now show $\overline{\epsilon}$-optimality of the proposed incentive sequence. Note that an optimal incentive sequence, i.e., $\overline{\epsilon}{=}0$, does not exist since choosing $\epsilon{=}0$ in the cost of control function ϕ^{ϵ}_n may not make the incentivized action uniquely optimal for the agent. As a result, the principal may not be able to control the agent's actions by offering such incentives.

We use the following lemma to state the main result.

Lemma 1: There exists a policy $\widetilde{\pi}{\in}\arg\min_{\pi\in\Xi(\overline{\mathcal{M}}')}f_0(\pi)$ such that $\xi^{\widetilde{\pi}}(s,a){<}\infty$ for all $s{\in}\overline{S}_r$ and $a{\in}\mathcal{A}(s)$.

Proof (Sketch): The problem of synthesizing a policy $\widetilde{\pi}$ such that $\widetilde{\pi}{\in}\arg\min_{\pi\in\Xi(\overline{\mathcal{M}}')}f_0(\pi)$ can be recast as a stochastic shortest path (SSP) problem with dead ends and zero-cost loops. Specifically, the dead ends are the states $\overline{S}_0$ and zero-cost loops are formed by states $\overline{S}_r$. The existence of stationary policies for such SSP problems can be established by slightly modifying the statement of Theorem 1 in [13]. Since any stationary policy $\pi{\in}\Xi(\overline{\mathcal{M}}')$ is guaranteed to reach the set $\overline{B}\cup\overline{S}_0$ with probability 1 within finite number of stages, the result follows. □

Theorem 1: For any given $\overline{\epsilon}{>}0$, there exists $\epsilon{>}0$ such that

$$\min_{\pi\in\Xi(\overline{\mathcal{M}}')} f_{\epsilon}(\pi) \leq \min_{\pi\in\Xi(\overline{\mathcal{M}}')} f_0(\pi)+\overline{\epsilon}.$$

Proof: For any policy $\pi{\in}\Xi(\overline{\mathcal{M}}')$ such that $\xi^{\pi}(s,a){<}\infty$ for all $s{\in}\overline{S}_r$ and $a{\in}\mathcal{A}(s)$, we have

$$f_{\epsilon}(\pi) = f_0(\pi)+\sum_{s\in\overline{S}_r}\sum_{a\in\mathcal{A}(s)}\xi^{\pi}(s,a)\epsilon. \tag{11}$$

Now, for a given $\overline{\epsilon}>0$, we evaluate both sides of the above equation at $\overline{\pi}\in\arg\min_{\pi\in\Xi(\overline{\mathcal{M}}')} f_0(\pi)$, which satisfies the condition $\xi^{\overline{\pi}}(s,a)<\infty$ due to Lemma 1. Choosing

$$\epsilon = \frac{\overline{\epsilon}}{\sum_{s\in S_r}\sum_{a\in\mathcal{A}(s)}\xi^{\overline{\pi}}(s,a)} > 0$$

and taking the minimum of the left hand side of (11) over the set $\Xi(\mathcal{M})$, we conclude the result. □

We conclude this section by noticing a remarkable property of the proposed incentive sequence. Specifically, to implement the proposed sequence (10), the principal should use only a simple switch mode which offers the same incentives until the agent reaches the set $B\cup S_0$ and shifts all incentives to zero after the agent either satisfies the principal's objective or fails to satisfy it.

V. Computation of Optimal Incentive Sequences

In the previous section, we developed a method to synthesize an $\overline{\epsilon}$-optimal incentive sequence which require us to solve a constrained cost minimization problem given in (8). Specifically, to solve the problem in (4a)-(4d), one should synthesize a *stationary deterministic* policy $\widetilde{\pi}$ such that

$$\widetilde{\pi} \in \arg\min_{\pi\in\Xi(\overline{\mathcal{M}}')} \mathbb{E}^{\pi}_{s_0}\Big[\sum_{t=1}^{\infty}\underline{\phi}^{\epsilon}(S_t,\mathcal{A}_t)\Big]. \tag{12}$$

In this section, we develop a method to solve the above optimization problem. For the ease of notation, we consider an scLTL formula of the form $\varphi=\lozenge p$. The design of incentives for general scLTL formulae is shown in Section VI.

A. Construction of the feasible policy space

To solve the problem (12), we first represent the set $\Xi(\overline{\mathcal{M}}')$ of feasible policies as a set of policies that maximizes the expected total reward with respect to a specific reward function.

For a given MDP $\overline{\mathcal{M}}$, we partition the set of states into three disjoint sets $\overline{B}$, $\overline{S}_0$, and $\overline{S}_r$ as explained in Section IV, and make the states $s\in\overline{B}\cup\overline{S}_0$ absorbing to form the modified MDP $\overline{\mathcal{M}}'$. For the modified MDP, we define a reward function $r{:}\overline{S}\times\mathcal{A}\rightarrow\mathbb{R}_{\geq 0}$ such that

$$r(s,a) = \begin{cases}\sum_{s'\in\overline{B}}\overline{\mathcal{P}}_{s,a,s'} & \text{if } s\in\overline{S}_r\\ 0 & \text{otherwise.}\end{cases}$$

By making use of the known results, e.g., Theorem 10.100 in [10], it can be easily shown that for any $s\in\overline{S}$ and $\pi\in\Pi(\overline{\mathcal{M}}')$,

$$\mathbb{E}^{\pi}_{s}\Big[\sum_{t=1}^{\infty} r(S_t,\mathcal{A}_t)\Big] = \Pr^{\pi}(s\models\varphi)$$

where $\varphi=\lozenge p$, $p\in\mathcal{AP}$, and $\{p\}\subseteq\mathcal{L}(s')$ if and only if $s'\in\overline{B}$. Let $x^{\star}_{s}{:=}\max_{\pi\in\Pi(\overline{\mathcal{M}}')}\Pr^{\pi}(s\models\varphi)$. Then, the problem (12) can be rewritten as

$$\min_{\pi\in\Pi(\overline{\mathcal{M}}')} \ \mathbb{E}^{\pi}_{s_0}\Big[\sum_{t=1}^{\infty}\underline{\phi}^{\epsilon}(S_t,\mathcal{A}_t)\Big] \tag{13a}$$

$$\text{subject to: } \ \mathbb{E}^{\pi}_{s_0}\Big[\sum_{t=1}^{\infty} r(S_t,\mathcal{A}_t)\Big] = x^{\star}_{s_0}. \tag{13b}$$

B. Synthesis of an optimal stationary deterministic policy

Using Lemma 1, one can formulate the problem (13a)-(13b) as a linear optimization problem and synthesize an optimal stationary policy. First, we compute the maximum probability of satisfying the specification φ, i.e., $x^{\star}_{s_0}{=}\max_{\pi\in\Pi(\overline{\mathcal{M}}')}\Pr^{\pi}(s_0\models\varphi)$, by solving a linear program (LP) [10] (see Chapter 10). Then we solve the following LP

$$\underset{\lambda(s,a)}{\text{minimize}} \quad \sum_{s\in\overline{S}_r}\sum_{a\in\mathcal{A}}\lambda(s,a)\underline{\phi}^{\epsilon}(s,a) \tag{14a}$$

$$\text{subject to:} \quad \sum_{s\in\overline{S}_r}\sum_{a\in\mathcal{A}}\lambda(s,a)r(s,a) = x^{\star}_{s_0} \tag{14b}$$

$$\forall s\in\overline{S}_r,\ \sum_{a\in\mathcal{A}(s)}\lambda(s,a) - \sum_{s'\in\overline{S}_r}\sum_{a\in\mathcal{A}(s)}\overline{\mathcal{P}}_{s',a,s}\lambda(s',a) = \alpha(s) \tag{14c}$$

$$\forall s\in\overline{S}_r,\ a\in\mathcal{A}(s),\ \lambda(s,a)\geq 0 \tag{14d}$$

where $\alpha{:}\overline{S}\rightarrow\{0,1\}$ is a function such that $\alpha(s_0){=}1$ and $\alpha(s){=}0$ for all $s{\in}\overline{S}\backslash\{\overline{s}_0\}$. The variable $\lambda(s,a)$ denotes the expected residence time in the state-action pair (s,a) [1], [14]. The constraint (14b) ensures that the probability of satisfying the specification φ is maximized, and the constraints (14c) represent the balance between the "inflow" to and "outflow" from states.

For each $s{\in}\overline{S}_r$ and $a{\in}\mathcal{A}(s)$, let $\lambda^{\star}(s,a)$ be optimal decision variables in (14a)-(14d). An optimal stationary policy $\pi^{\star}{=}\{d^{\star},d^{\star},\ldots\}$ that solves the problem (13a)-(13b) is then given by

$$d^{\star}(s,a) := \begin{cases}\frac{\lambda^{\star}(s,a)}{\sum_{a\in\mathcal{A}(s)}\lambda^{\star}(s,a)} & \text{if } \sum_{a\in\mathcal{A}(s)}\lambda^{\star}(s,a)>0\\ \text{arbitrary} & \text{otherwise}\end{cases} \tag{15}$$

for $s{\in}\overline{S}_r$, and $d^{\star}(s,a){=}1$ for an arbitrary $a{\in}\mathcal{A}(s)$ for $s{\notin}\overline{S}_r$.

We note that a policy constructed through (15) is randomized in general. One can argue that choosing one of the actions $a{\in}\mathcal{A}(s)$ such that $d^{\star}(s,a){>}0$ deterministically yields an optimal stationary deterministic policy. However, the following example illustrates that such an approach may result in an infeasible policy for the problem (14a)-(14d).

Example 1: Consider the MDP given in Fig. 2, where the cost of control $\underline{\phi}^{\epsilon}$ is such that $\underline{\phi}^{\epsilon}(s_1,a_2){=}1$ and $\underline{\phi}^{\epsilon}(s,a){=}0$ otherwise. Suppose that the specification is $\varphi{=}\lozenge s_2$, i.e., $r(s_1,a_2){=}1$ and $r(s,a){=}0$ otherwise. For the LP (14a)-(14d), a set of optimal decision variables is given by $\lambda^{\star}(s_0,a_1){=}2$, $\lambda^{\star}(s_1,a_1){=}1$, and $\lambda^{\star}(s_1,a_2){=}1$. Therefore, an optimal policy synthesized through (15) is $d^{\star}(s_0,a_1){=}1$, $d^{\star}(s_1,a_1){=}1/2$, and $d^{\star}(s_1,a_2){=}1/2$. Clearly, if we consider

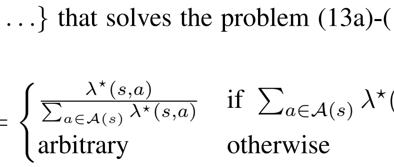

Fig. 2: An MDP example for which arbitrarily choosing one of the optimal actions and taking it deterministically yields an infeasible policy.

a deterministic policy such that $d(s_1, a_1){=}1$, the probability of satisfying the specification φ under this policy is zero. Hence, choosing an arbitrary action $a{\in}\mathcal{A}(s)$ such that $d^\star(s,a){>}0$ deterministically violates the constraint and yields an infeasible policy.$\triangleleft$

As Example 1 illustrates, a structured approach is required to synthesize an optimal deterministic policy from the solution of the LP (14a)-(14d). Let $\upsilon^\star$ be the optimal value of the LP in (14a)-(14d). To synthesize an optimal deterministic policy, we first solve the following LP,

$$\underset{\lambda(s,a)}{\text{minimize}} \qquad \sum_{s\in\overline{S}_r}\sum_{a\in\mathcal{A}}\lambda(s,a) \tag{16a}$$

$$\text{subject to:} \qquad \sum_{s\in\overline{S}_r}\sum_{a\in\mathcal{A}}\lambda(s,a)r(s,a) = x^\star_{s_0} \tag{16b}$$

$$\sum_{s\in\overline{S}_r}\sum_{a\in\mathcal{A}}\lambda(s,a)\underline{\phi}^\epsilon(s,a) = \upsilon^\star \tag{16c}$$

$$\forall s\in\overline{S}_r, \sum_{a\in\mathcal{A}(s)}\lambda(s,a) - \sum_{s'\in\overline{S}_r}\sum_{a\in\mathcal{A}(s)}\overline{\mathcal{P}}_{s',a,s}\lambda(s',a) = \alpha(s) \tag{16d}$$

$$\forall s\in\overline{S}_r, \quad a\in\mathcal{A}(s), \ \lambda(s,a)\geq 0. \tag{16e}$$

From the optimal decision variables $\lambda^\star(s,a)$ of (16a)-(16e), an optimal policy $\pi^\star{=}\{d^\star, d^\star, \ldots\}$ can be generated as follows. Let $\mathcal{A}^\star(s){:=}\{a{\in}\mathcal{A}(s) : \lambda^\star(s,a){>}0\}$. If $\mathcal{A}^\star(s){\neq}\emptyset$, we choose $d^\star(s,a){=}1$ for an arbitrary $a{\in}\mathcal{A}^\star(s)$, and if $\mathcal{A}^\star(s){=}\emptyset$, we choose $d^\star(s,a){=}1$ for an arbitrary $a{\in}\mathcal{A}(s)$.

Proposition 1: A stationary deterministic policy generated from the optimal decision variables $\lambda^\star(s,a)$ of (16a)-(16e) is a solution to the problem (13a)-(13b).[1]

Intuitively, the LP in (16a)-(16e) computes the minimum expected time to reach the set $\overline{B}$ with probability $x^\star_{s_0}$ with the cost of $\upsilon^\star$. Therefore, if $\lambda^\star(s,a){>}0$, by taking the action $a{\in}\mathcal{A}(s)$, the agent has to "get closer" to the set $\overline{B}$ with nonzero probability. Otherwise, the minimum expected time to reach the set $\overline{B}$ would be strictly decreased. Consequently, by choosing an arbitrary action $a{\in}\mathcal{A}^\star(s)$, the agent is guaranteed to reach the set $\overline{B}$ with the desired probability.

VI. Incentive Design for General scLTL Specifications

In previous sections, we have developed methods to synthesize $\overline{\epsilon}$-optimal incentive sequences for reachability specifications $\varphi{=}\Diamond p$. For such specifications, the principal induces the desired agent behavior by sharing only the incentive sequences with the agent. In other words, the principal does not have to inform the agent explicitly about the specification. In this section, we show that for general scLTL formulae, the problem (4a)-(4d) may not have a feasible solution, in which case the principal must share its objective with the agent to induce the desired behavior.

To solve the problem (4a)-(4d) for general scLTL formulae, one needs to utilize the techniques from automata theory [10]. In particular, we use the fact that for any scLTL formula φ built up from $\mathcal{AP}$, we can construct a *deterministic finite automata* (DFA) $A_\varphi{=}(\mathcal{Q}, q_0, 2^{\mathcal{AP}}, \delta_\varphi, \mathcal{F})$ where $\mathcal{Q}$ is a finite set of memory states, $2^{\mathcal{AP}}$ is the alphabet, $\delta_\varphi{:}\mathcal{Q}{\times}2^{\mathcal{AP}}{\rightarrow}\mathcal{Q}$ is a transition function and $\mathcal{F}{\subseteq}\mathcal{Q}$ is the set of accepting states [15]. Then, after forming the expanded MDP $\overline{\mathcal{M}}$ for a given MDP $\mathcal{M}$ and a decision horizon N as explained in Section IV-B, one can construct the product MDP which is defined as follows.

Definition 7: Let $\overline{\mathcal{M}}{=}(\overline{S}, \overline{s}_0, \mathcal{A}, \overline{\mathcal{P}}, \mathcal{AP}, \overline{\mathcal{L}}, \overline{\mathcal{R}})$ be an expanded MDP and $A_\varphi{=}(\mathcal{Q}, q_0, 2^{\mathcal{AP}}, \delta_\varphi, \mathcal{F})$ be a DFA. The *product MDP* $\mathcal{M}_p{=}(S_p, s_{0_p}, \mathcal{A}, \mathbb{P}, \mathcal{AP}, \mathcal{L}_p, \mathcal{F}_p, \overline{\mathcal{R}})$ is a tuple where

- $S_p{=}\overline{S}{\times}Q$,
- $s_{0_p} = (\overline{s}_0, q)$ such that $q = \delta(q_0, \overline{\mathcal{L}}(\overline{s}_0))$,
- $\mathbb{P}((s,q), a, (s',q')){=}\begin{cases}\overline{\mathcal{P}}_{s,a,s'} & \text{if} \quad q' = \delta(q, \overline{\mathcal{L}}(s'))\\ 0 & \text{otherwise,}\end{cases}$
- $\mathcal{L}_p((s,q)) = \{q\}$,
- $\mathcal{F}_p{=}\overline{S}\times\mathcal{F}$.

The problem in (4a)-(4d) can now be solved on the product MDP $\mathcal{M}_p$ in three steps. First, we partition the states of $\mathcal{M}_p$ into three disjoint sets. Let $B{:=}\mathcal{F}_p$, S_0 be the set of states that have zero probability of reaching the set B, and $S_r{:=}S_p\backslash B{\cup}S_0$. Second, we form the modified product MDP $\mathcal{M}'_p$ by making all states $B{\cup}S_0$ absorbing. Finally, we apply the methods developed in Section IV to synthesize an $\overline{\epsilon}$-optimal incentive sequence on $\mathcal{M}'_p$.

Note that the incentive sequence is designed on the product MDP $\mathcal{M}_p$. Therefore, the principal must share the DFA structure, i.e., it's objective, with the agent to use the computed sequence. However, for the existence of a solution to the problem (4a)-(4d), the incentive sequence should be designed on the MDP $\mathcal{M}$. The following example illustrates that the problem (4a)-(4d) may have no feasible solution, even though the existence of an $\overline{\epsilon}$-optimal incentive sequence on $\mathcal{M}_p$ is guaranteed.

Example 2: Consider the MDP given in Fig. 3, where the numbers next to actions a_i represent the transition probabilities, e.g., $\mathcal{P}_{s_0,a_1,s_1}{=}0.4$, and the letters next to state numbers represent labels, e.g., $\mathcal{L}(s_0){=}A$. Let the agent's decision horizon be $N{=}3$, and the reward function $\mathcal{R}$ be such that $\mathcal{R}(s_0,a_1){=}1$ and $\mathcal{R}(s,a){=}0$ otherwise. Additionally, let the principal's objective be expressed by the scLTL formula $\varphi{=}\Diamond(B{\wedge}\Diamond C)$, i.e., first visit state B and then state C. The maximum probability of satisfying φ is $x^\star_0{=}0.5$, which can be computed by solving an LP [10]. The value $x^\star_0$ is attainable if and only if the agent takes the action $a_2{\in}\mathcal{A}(s_0)$

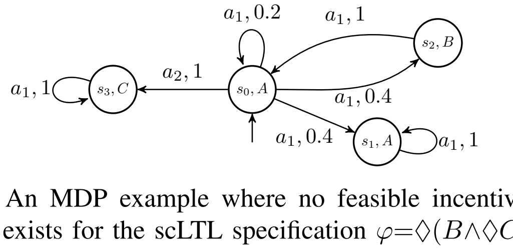

Fig. 3: An MDP example where no feasible incentive sequence exists for the scLTL specification $\varphi{=}\Diamond(B{\wedge}\Diamond C)$.

[1] Due to space restrictions, we provide a proof of Proposition 1 at https://mornik.ae.illinois.edu/wp-content/uploads/SGORT19.pdf

with probability 1 after visiting state s_2.

The principal should decide on which actions to incentivize in the first three stages $t{=}1,2,3$ since the agent's decision horizon is $N{=}3$. Clearly, the action a_1 should be incentivized for $t{=}1,2$ so that the agent visits state s_2. At $t{=}3$, the agent will be in state s_0 with nonzero probability. Now, if the principal incentivize a_1, the agent will take action $a_2{\in}\mathcal{A}(s_0)$ with probability less then 1 after visiting state s_2. On the other hand, if a_2 is incentivized, then the agent cannot satisfy the specification with probability higher than 0.46. Consequently, no incentive sequence on the given MDP can guarantee the satisfaction of the specification φ with maximum probability. $\triangleleft$

We now present a sufficient condition on the structure of the MDP $\mathcal{M}$ which guarantees the existence of an $\overline{\epsilon}$-optimal incentive sequence on $\mathcal{M}$.

For the product MDP $\mathcal{M}_p$ and a policy $\pi{\in}\Pi(\mathcal{M}_p)$, let $M^{\pi}_{s,t}{:=}\{q{\in}\mathcal{Q}{:}\sum_{a\in\mathcal{A}}\mu^{\pi}_t((s,q),a){>}0\}$ be the set of occupied memory states when the agent is in state $s{\in}\overline{S}$ at stage $t{\in}\mathbb{N}$.

Theorem 2: For an MDP $\mathcal{M}$ and a decision horizon N, let $\overline{S}$ be the finite set of states for the expanded MDP. There exists an $\overline{\epsilon}$-optimal incentive sequence on $\mathcal{M}$ if there exists an $\overline{\epsilon}$-optimal incentive sequence on $\mathcal{M}_p$ such that the agent's optimal policy $\pi{\in}\Pi(\mathcal{M}_p)$ under the provided incentives satisfies $|M^{\pi}_{s,t}|{\leq}1$ for all $s{\in}\overline{S}$ and $t{\in}\mathbb{N}$.

Proof: Let Γ be an $\overline{\epsilon}$-optimal incentive sequence on $\mathcal{M}_p$ with the desired property. Note that the function in (10) is a mapping from the expanded MDP $\overline{\mathcal{M}}$ to the MDP $\mathcal{M}$ that preserves $\overline{\epsilon}$-optimality of the incentive sequence. Therefore, in what follows, we construct an incentive mapping from $\mathcal{M}_p$ to $\overline{\mathcal{M}}$ that preserves $\overline{\epsilon}$-optimality of the sequence Γ, and conclude the result.

An $\overline{\epsilon}$-optimality-preserving mapping ψ such that $\Gamma'{=}\{\gamma'_1,\gamma'_2,\ldots\}{:=}\psi(\Gamma)$ where $\Gamma{=}\{\gamma_1,\gamma_2,\ldots\}$ is given as follows. For a given $t{\in}\mathbb{N}$,

- if $|M^{\pi}_{s,t}|{=}0$, $\gamma'_t(s,a){:=}\gamma_t((s,q),a)$ for an arbitrary $q{\in}\mathcal{Q}$ and for all $a{\in}\mathcal{A}(s)$,
- if $|M^{\pi}_{s,t}|{=}1$, $\gamma'_t(s,a){:=}\gamma_t((s,q),a)$ for $q{\in}M^{\pi}_{s,t}$ and for all $a{\in}\mathcal{A}(s)$. $\square$

The following corollary follows from the fact that the principal can induce a stationary deterministic agent policy on the product MDP through the methods explained in Section V.

Corollary 1: For an MDP, there exists an $\overline{\epsilon}$-optimal incentive sequence if $\mathcal{P}_{s,a,s'}{\in}\{0,1\}$ for all $s,s'{\in}S$ and $a{\in}\mathcal{A}(s)$.

Finally, we provide a sufficient condition on the agent's decision horizon N that ensures the existence of an $\overline{\epsilon}$-optimal incentive sequence on $\mathcal{M}$.

Proposition 2: For an MDP $\mathcal{M}$, there exists an $\overline{\epsilon}$-optimal incentive sequence if the agent's decision horizon is $N{=}1$.

Proof (Sketch): There is a one-to-one correspondence between the paths of the product MDP $\mathcal{M}_p$ and the MDP $\mathcal{M}$ [10]. Therefore, the principal can observe the path followed by the agent on $\mathcal{M}$, and provide the incentives according to the corresponding path on $\mathcal{M}_p$ at each stage. Because $N{=}1$, the principal knows the memory state occupied by the agent at each stage. Consequently, it becomes possible to map the incentives from $\mathcal{M}_p$ to $\mathcal{M}$ at each stage. $\square$

VII. Numerical Simulations

In this section, we demonstrate the proposed methods on two simple motion planning examples. Considering the availability of off-the-shelf solvers, e.g., Gurobi [16], MOSEK [17], that can efficiently solve large-scale linear optimization problems, we restrict our attention to small scale examples to better emphasize the properties of the proposed methods. We synthesize the incentive sequences for the following examples through the use of MOSEK [17] solver together with CVXPY [18] interface.

A. Incentives for reachability objectives

In this example, we consider a $5{\times}5$ grid world environment, shown in Fig. 4, and an agent with decision horizon $N{=}1$. At each state, the agent has four actions, i.e., $\mathcal{A}{=}\{left,right,up,down\}$, and a transition to the chosen direction occurs with probability 1. If the adjacent state in the chosen direction is the boundary of the environment, the agent stays in its current state. A reward function for the agent is generated by choosing all rewards $\mathcal{R}(s,a)$ from the set $\{0,1,\ldots,9\}$ uniformly randomly.

The agent starts from the bottom left corner, i.e., *Start* state in Fig. 4, and aims to maximize its immediate reward at each stage. The principal provides incentives to the agent so that the agent reaches the top right corner, i.e., *Target* state in Fig. 4.

In the absence of incentives, i.e., $\gamma_t(s,a){=}0$ for all $t{\in}\mathbb{N}$, the agent's optimal path is shown by blue arrows in Fig. 4. Under its optimal policy, the agent cycles between two states infinitely often. We synthesize an incentive sequence for the agent so that it reaches the target state with probability 1. The agent's optimal path under the provided incentives is shown by red arrows in Fig. 4. The total cost to the principal is computed as $9{+}10\epsilon$ units of resources (UR) where $\epsilon{>}0$ is an arbitrarily small constant.

As can be seen from Fig. 4, under the provided incentives, the agent follows the lowest cost path rather than the shortest

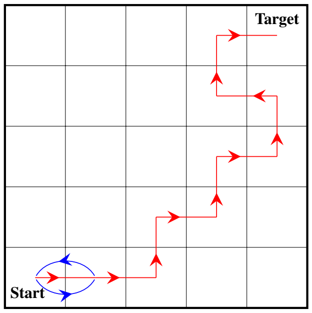

Fig. 4: The motion of an agent on a grid world. The agent's decision horizon is $N{=}1$, and it starts from the *Start* state. The principal's objective is to induce an agent policy that reaches the *Target* state with probability 1. Blue and red arrows indicate the agent's optimal policy in the absence and the presence of incentives, respectively.

one to the target state. Specifically, the shortest path would take 8 stages to reach the target state and cost 12 UR to the principal, whereas the lowest cost path takes 10 stages to reach the target and cost 9 UR. Quantitatively, the proposed incentive sequence allows the principal to save 25% of the resources that would be paid to the agent if it was to follow the shortest path.

B. Incentives for general scLTL specifications

In this example, we consider the same grid world environment introduced in the previous example with different state labels. The agent's decision horizon is $N{=}4$, and its objective is to reach the state labeled as C in Fig. 5. The principal's objective is to induce an agent policy that satisfies the scLTL specification $\varphi{=}\Diamond(A \wedge \Diamond(B \wedge \Diamond C))$, i.e., the agent should first visit state A, then B, and then C, with probability 1.

The agent receives the reward of 2 for transitioning to the top left state and the reward of 5 for transitioning to the top right state. Its optimal path in the absence of incentives is shown in Fig. 5 with blue arrows (top path). We synthesize an optimal incentive sequence under which the agent's optimal path is shown in Fig. 5 with red arrows (bottom path).

The total cost of the incentives to the principal is computed as $2{+}13\epsilon$ units of resources. Specifically, the principal provides $2{+}\epsilon$ incentives for the $right$ action in the start state and then ϵ incentives at each stage for desired actions. An interesting property of the incentivized (red) path is that the agent stays in the same state in third stage by taking $down$ action. This is due to the fact that the state s on the left of the state labeled as B has value $V_n(s){=}0$ for all n. Therefore, the principal wants that state to be the agent's initial state when it computes its second 4-stage policy. By doing so, the principal ensures that the states s' occupied by the agent in the next 4 stages will always have a value zero, i.e., $V_n(s'){=}0$ if $\sum_{a\in\mathcal{A}(s')} \mu^{\pi}_{4+n}(s',a){>}0$, and therefore the cost of control will only be ϵ.

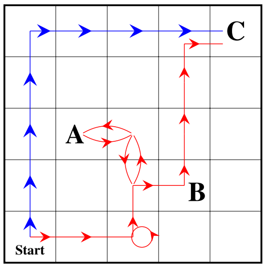

Fig. 5: The motion of an agent on a grid world. The agent's decision horizon is $N{=}4$, and it starts from the *Start* state. The principal's objective is to induce an agent policy that first visits A, then B, and then C. The optimal path of the agent in the absence of incentives is shown by blue arrows (top path). Red arrows indicate the agent's optimal path under the provided incentives (bottom path).

VIII. Conclusions and Future Directions

We considered a principal-agent model and studied the problem of designing an optimal sequence of incentives that the principal should offer to the agent in order to induce a desired agent behavior expressed as a syntactically co-safe linear temporal logic (scLTL) formula. For reachability objectives, we presented a polynomial-time algorithm to synthesize an incentive design that minimizes the cost to the principal. We showed with an example scenario that a feasible incentive design may not exists for general scLTL formulae, and the principal may need to share its objective with the agent to induce the desired behavior. Furthermore, we provided sufficient conditions under which the principal can induce the desired behavior without sharing the scLTL formula with the agent.

The results of this paper are obtained under the assumptions that the agent's reward function and the length of its decision horizon are known to the principal. An interesting future direction may be to develop methods for inferring the length of the agent's decision horizon, or to design an incentive sequence that does not require the knowledge of the length of the decision horizon.

References

[1] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., 1994.

[2] M. Kloetzer and C. Mahulea, "Multi-robot path planning for syntactically co-safe LTL specifications," in *International Workshop on Discrete Event Systems (WODES)*, 2016, pp. 452–458.

[3] A. Armando, R. Carbone, and L. Compagna, "LTL model checking for security protocols," in *IEEE Computer Security Foundations Symposium*, 2007, pp. 385–396.

[4] L. Tan, O. Sokolsky, and I. Lee, "Specification-based testing with linear temporal logic," in *IEEE International Conference on Information Reuse and Integration*, 2004, pp. 493–498.

[5] H. Zhang and D. C. Parkes, "Value-based policy teaching with active indirect elicitation." in *AAAI Conference on Artificial Intelligence*, 2008, pp. 208–214.

[6] H. Zhang, Y. Chen, and D. C. Parkes, "A general approach to environment design with one agent." in *International Joint Conference on Artifical Intelligence*, 2009, pp. 2002–2014.

[7] H. Zhang, D. C. Parkes, and Y. Chen, "Policy teaching through reward function learning," in *ACM Conference on Electronic commerce*, 2009, pp. 295–304.

[8] Y. Chen, J. Kung, D. C. Parkes, A. D. Procaccia, and H. Zhang, "Incentive design for adaptive agents," in *International Conference on Autonomous Agents and Multiagent Systems*, 2011, pp. 627–634.

[9] Y.-C. Ho, P. B. Luh, and G. J. Olsder, "A control-theoretic view on incentives," *Automatica*, vol. 18, no. 2, pp. 167–179, 1982.

[10] C. Baier and J.-P. Katoen, *Principles of Model Checking*. MIT Press, 2008.

[11] O. Kupferman and M. Y. Vardi, "Model checking of safety properties," *Formal Methods in System Design*, vol. 19, no. 3, pp. 291–314, 2001.

[12] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*. Athena Scientific, 1996.

[13] F. Teichteil-Königsbuch, "Stochastic safest and shortest path problems." in *AAAI Conference on Artificial Intelligence*, 2012.

[14] K. Etessami, M. Kwiatkowska, M. Y. Vardi, and M. Yannakakis, "Multi-objective model checking of Markov decision processes," in *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, 2007, pp. 50–65.

[15] C. Belta, B. Yordanov, and E. A. Gol, *Formal Methods for Discrete-Time Dynamical Systems*. Springer, 2017.

[16] L. Gurobi Optimization, "Gurobi optimizer reference manual," 2018. [Online]. Available: http://www.gurobi.com

[17] M. ApS, *MOSEK Optimizer API for Python. Version 8.1.*, 2019. [Online]. Available: https://docs.mosek.com/8.1/pythonapi/index.html

[18] S. Diamond and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization," *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016.