Optimal Control of Partially Observable Markov Decision Processes with Finite Linear Temporal Logic Constraints (Supplementary Material)

Krishna C. Kalagarla[1] **Dhruva Kartik**[1] **Dongming Shen**[1] **Rahul Jain**[1] **Ashutosh Nayyar**[1] **Pierluigi Nuzzo**[1]

[1]Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, CA, USA

A PROOF OF THEOREM 1

For any policy μ, we have

$$\begin{aligned} \mathcal{R}^{\mathscr{M}^\times}(\mu) &= \mathbb{E}_\mu \left[\sum_{t=0}^{T} r_t^\times (X_t, A_t) \right] && (1) \\ &= \mathbb{E}_\mu \left[\sum_{t=0}^{T} r_t^\times ((S_t, Q_t), A_t) \right] && (2) \\ &\overset{a}{=} \mathbb{E}_\mu \left[\sum_{t=0}^{T} r_t (S_t, A_t) \right] = \mathcal{R}^{\mathscr{M}}(\mu), && (3) \end{aligned}$$

where the equality in (a) follows from the definition of $r_t^\times$ in (7). Further, using (8), we have

$$r^f(X_{T+1}) = r^f((S_{T+1}, Q_{T+1})) = \mathbb{1}_F(Q_{T+1}). \quad (4)$$

Following the acceptance condition of the DFA $\mathscr{A}$, which is equivalent to the LTL$_f$ specification φ, a run ξ of the POMDP satisfies φ if and only if the word generated by the run satisfies the acceptance condition of the DFA $\mathscr{A}$, i.e., its run on $\mathscr{A}$, $\xi_{\mathscr{A}}$, ends in the acceptance set F. Hence,

$$\mathcal{R}^f(\mu) = \mathbb{E}_\mu \left[r^f(X_{T+1}) \right] = \mathbb{P}_\mu^{\mathscr{M}}(\varphi). \quad (5)$$

B PROOF OF LEMMA 1

We have

$$\begin{aligned} \mathcal{R}^* &= l^* && (6) \\ &\leq l_B^* && (7) \\ &\leq \inf_{0 \leq \lambda \leq B} L(\bar{\mu}, \lambda) + \epsilon && (8) \\ &= \mathcal{R}^{\mathscr{M}^\times}(\bar{\mu}) + \inf_{0 \leq \lambda \leq B} \lambda(\mathcal{R}^f(\bar{\mu}) - 1 + \delta) + \epsilon. && (9) \end{aligned}$$

There are two possible cases: (i) $\mathcal{R}^f(\bar{\mu}) - 1 + \delta \geq 0$ and (ii) $\mathcal{R}^f(\bar{\mu}) - 1 + \delta < 0$.

If case (i) is true, then (16) is trivially satisfied. Further, in this case, we have

$$\inf_{0 \leq \lambda \leq B} \lambda(\mathcal{R}^f(\bar{\mu}) - 1 + \delta) = 0. \quad (10)$$

Therefore, $\mathcal{R}^* \leq \mathcal{R}^{\mathscr{M}^\times}(\bar{\mu}) + \epsilon$, and hence, (15) is satisfied. If case (ii) is true, we have

$$\begin{aligned} \inf_{0 \leq \lambda \leq B} \lambda(\mathcal{R}^f(\bar{\mu}) - 1 + \delta) &= B(\mathcal{R}^f(\bar{\mu}) - 1 + \delta) && (11) \\ &< 0. && (12) \end{aligned}$$

Therefore, $\mathcal{R}^* \leq \mathcal{R}^{\mathscr{M}^\times}(\bar{\mu}) + \epsilon$, and hence, (15) is satisfied. Further, we have

$$\begin{aligned} B(\mathcal{R}^f(\bar{\mu}) - 1 + \delta) &\geq \mathcal{R}^* - \mathcal{R}^{\mathscr{M}^\times}(\bar{\mu}) - \epsilon && (13) \\ &\geq \mathcal{R}^* - R_m - \epsilon. && (14) \end{aligned}$$

The last inequality holds because R_m is the maximum achievable reward. Hence, (16) is satisfied.

C PROOF OF THEOREM 2

Consider the dual of (P4). Let

$$u_B^* := \inf_{0 \leq \lambda \leq B} \sup_\mu L(\mu, \lambda). \quad \text{(P5)}$$

Accepted for the 38th Conference on Uncertainty in Artificial Intelligence (UAI 2022).

We have

$$
\begin{aligned}
l_B^* &\overset{a}{\leq} u_B^* && (15)\\
&= \inf_{0\leq\lambda\leq B} \sup_{\mu} L(\mu,\lambda) && (16)\\
&\leq \sup_{\mu} L(\mu,\bar{\lambda}) && (17)\\
&\overset{b}{=} \frac{1}{K}\sum_{k=1}^{K} L(\mu_{\bar{\lambda}},\lambda_k) && (18)\\
&\overset{c}{\leq} \frac{1}{K}\sum_{k=1}^{K} L(\mu_k,\lambda_k) && (19)\\
&\overset{d}{\leq} \frac{1}{K}\inf_{0\leq\lambda\leq B}\sum_{k=1}^{K} L(\mu_k,\lambda) + 2B\sqrt{2\log 2/K} && (20)\\
&\overset{e}{=} \inf_{0\leq\lambda\leq B} L(\bar{\mu},\lambda) + 2B\sqrt{2\log 2/K}. && (21)
\end{aligned}
$$

The inequality in (a) holds because of weak duality [Boyd and Vandenberghe, 2004]. The equality in (b) holds because of the bilinearity (affine) of $L(\cdot)$. The inequality in (c) holds because μ_k is the maximizer associated with λ_k. Inequality (d) follows from Corollary 5.7 in [Hazan et al., 2016]. Equality in (e) is again a consequence of bilinearity of $L(\cdot)$.

D PROOF OF LEMMA 2

The rewards $\mathcal{R}^{\mathscr{M}^\times}(\mu)$ and $\mathcal{R}^f(\mu)$ in the corresponding product POMDP are given by

$$
\begin{aligned}
\mathcal{R}^{\mathscr{M}^\times}(\mu) &= \mathbb{E}_\mu\left[\sum_{t=0}^{T} r_t^\times(X_t,A_t)\right] && (22)\\
&= \mathbb{E}_\mu\left[\sum_{t=0}^{\infty} \gamma^t r_t^\times(X_t,A_t)\right] && (23)\\
\mathcal{R}^f(\mu) &= \mathbb{E}_\mu\left[r^f(X_{T+1})\right] && (24)\\
&= (1-\gamma)\mathbb{E}_\mu\left[\sum_{t=0}^{\infty} \gamma^t r^f(X_{t+1})\right] && (25)\\
&= \frac{(1-\gamma)}{\gamma}\mathbb{E}_\mu\left[\sum_{t=1}^{\infty} \gamma^t r^f(X_t)\right]. && (26)
\end{aligned}
$$

Therefore, we have

$$
\begin{aligned}
&L(\mu,\lambda) && (27)\\
&= \mathbb{E}_\mu\left[\sum_{t=0}^{\infty}\gamma^t\left(r_t^\times(X_t,A_t) + \frac{\lambda(1-\gamma)}{\gamma}\gamma^t r^f(X_t)\right)\right]\\
&\quad - \frac{\lambda(1-\gamma)}{\gamma}\mathbb{E}[r^f(X_0)] - \lambda(1-\delta).
\end{aligned}
$$

E ADDITIONAL DETAILS ON EXPERIMENTS

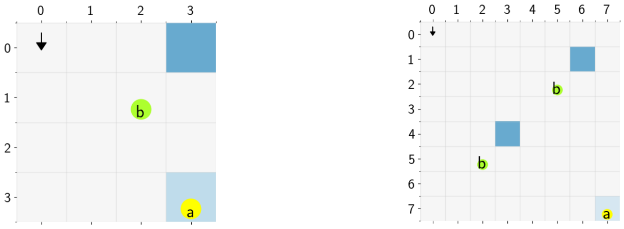
E.1 MODEL DESCRIPTION

In this subsection, we provide further details on the grid world POMDP models used in our experiments. The images corresponding to the various models indicate the state space and the labeling function, e.g, in Fig. 1a, we have $L[(1,2)] = \{b\}, L[(3,3)] = \{a\}$, and $L[(i,j)] = \{\}$ for all other grid locations (i,j). In all models, the agent starts from the grid location $(0,0)$. Further, the reward for all actions is 0 in all grid locations, unless specified otherwise. In the supplementary material, we also provide videos that capture some representative behaviors of the policies generated by Algorithm 1. We will discuss them in greater detail below.

E.1.1 Location Uncertainty

Reach-Avoid Tasks. In model $\mathscr{M}_1$, reward $r((0,3),a) = 2$ and $r((3,3),a) = 1$ for all actions a. We observe that the agent satisfies the reach-avoid constraint with high probability and ends up in the top-right corner where the reward is highest. A representative trajectory for this model can be found in the video `mu1_1.mp4`.

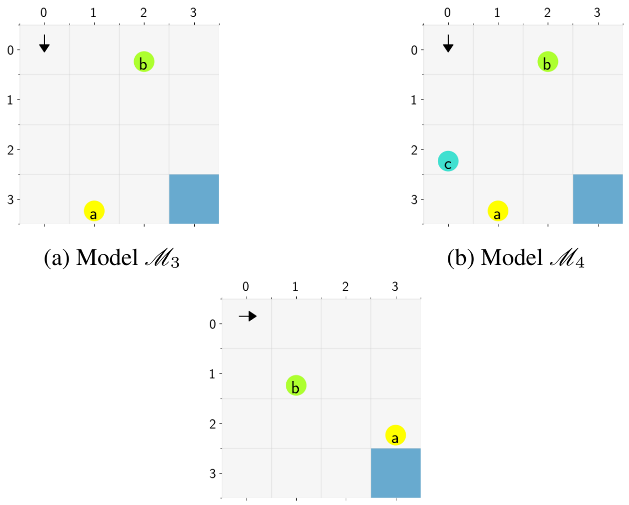
In model $\mathscr{M}_2$, reward $r((1,6)) = 3, r((4,3),a) = 3$, and $r((7,7),a) = 1$ for all actions a. In this model, we observe two characteristic behaviors. The agent reaches the goal state a and remains there (see video `mu2_1.mp4`). This behavior ensures that the specification is met but the reward is relatively lower. The other behavior is that the agent goes towards the location $(4,3)$ and tries to remain there to obtain higher reward (see video `mu2_2.mp4`). However, since the obstacle is very close and the transitions are stochastic, it is prone to violating the constraint. Nonetheless, this violation is rare enough such that the overall satisfaction probability exceeds the desired threshold.



(a) Model $\mathscr{M}_1$ (b) Model $\mathscr{M}_2$

Figure 1: Reach-Avoid Tasks

Ordered Tasks. For models $\mathscr{M}_3, \mathscr{M}_4$, and $\mathscr{M}_5$, reward $r((3,3),a) = 1$ for all actions a. In model $\mathscr{M}_3$, the agent visits a and then b in that order most of the time (see video `mu3_1.mp4`). Very rarely, the agent narrowly misses one of the goals due to the stochasticity in transitions and partial observability (see video `mu3_2.mp4`). In model $\mathscr{M}_4$, the agent is almost always successful in satisfying the constraint and maximizing the reward (see video `mu4_1.mp4`). In model $\mathscr{M}_5$, we see both successes (see video `mu5_1.mp4`) and failures (see video `mu5_2.mp4`). However, the failure probability is within the threshold, as suggested by Table 1.



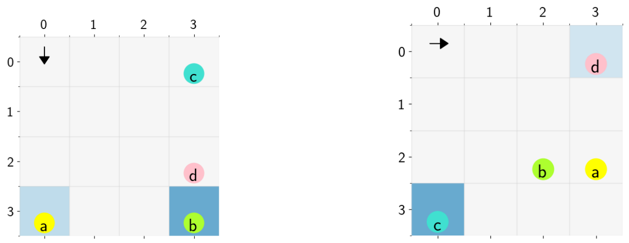
(a) Model $\mathscr{M}_3$

(b) Model $\mathscr{M}_4$

(c) Model $\mathscr{M}_5$

Figure 2: Ordered Tasks

Reactive Tasks. In model $\mathscr{M}_6$, reward $r((3,0),a) = 1$ and $r((3,3),a) = 2$ for all actions a. In this case, the agent goes to a and remains there, thus satisfying the constraint (see video `mu6_1.mp4`). Occasionally, the agent also goes to state b and remains there to obtain a large reward. However, this violates the constraint since if the agent ever visits b, it must eventually go to c (see video `mu6_2.mp4`). In model $\mathscr{M}_7$, reward $r((3,0),a) = 5$ and $r((0,3),a) = 2$ for all actions a. In this model, the agent goes to a and then to b so that it can go to c. If it had not gone to b immediately after reaching a, then it will be compelled to go to d. We observe that the agent consistently visits b after a (see video `mu7_1.mp4`).



(a) Model $\mathscr{M}_6$

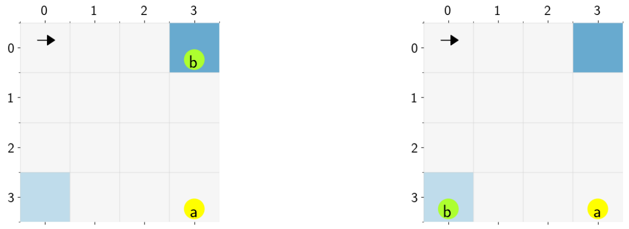
(b) Model $\mathscr{M}_7$

Figure 3: Reactive Tasks

E.1.2 Predicate Uncertainty

In the experiments of this section, there are two possible locations for object b: $(3,0)$ and $(0,3)$. In both cases, whenever the agent is 'far' away (Manhattan distance greater than 1) from the object b, it gets an observation 'F' indicating that it is *far* with probability 1. When the object is at the bottom left and the agent is adjacent to it, the agent gets an observation 'C' with probability 0.9 indicating that the object is *close*. However, if object b is at the top right and the agent is adjacent to it, the agent gets an observation 'C' only with probability 0.1. Therefore, the detection capability of the agent is stronger when the object is in the bottom-left location as opposed to when it is in the top-right location.

Reach-Avoid Tasks. In model $\mathscr{M}_8$, reward $r((3,0),a) = 2$ and $r((0,3),a) = 4$ for all actions a. In this model, generally, the agent first collects some information from the bottom-left, reaches a, and goes to the rewarding location that is not an obstacle (see videos `mu8_1.mp4`, `mu8_2.mp4`, `mu8_3.mp4`). We see rare instances where the agent completely ignores the constraint and maximizes the reward (see video `mu8_4.mp4`).

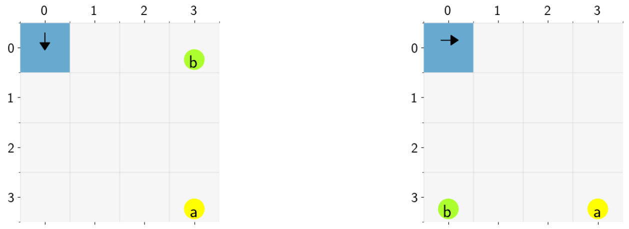


(a) Model $\mathscr{M}_8$ with obstacle at $(0,3)$

(b) Model $\mathscr{M}_8$ with obstacle at $(3,0)$

Figure 4: Reach-Avoid Tasks

Ordered Tasks. In model $\mathscr{M}_9$, reward $r((0,0),a) = 2$ for all actions a. In this model, we observe that the agent mostly succeeds in satisfying the constraint and maximizing the reward (see videos `mu9_1.mp4` and `mu9_2.mp4`).



(a) Model $\mathscr{M}_9$ with obstacle at $(0,3)$

(b) Model $\mathscr{M}_9$ with obstacle at $(3,0)$

Figure 5: Ordered Tasks

E.2 HYPER-PARAMETERS AND RUNTIMES

The parameter δ in all the experiments is chosen in the following manner: i) We first solve a POMDP problem in which we are only interested in maximizing the probability of satisfaction of the LTL$_f$ constraint. Let this probability be denoted by p_{max}. The SARSOP solver gives concrete approximation bounds on its solution, and therefore, on our estimate of p_{max}. ii) Since any threshold $1-\delta$ larger than p_{max} is infeasible, we choose a δ such that $1-\delta$ is around $0.9p_{max}$. The values η and B are hyperparameters in our experiments. The η suggested by Theorem 2 in our paper is guaranteed to result in convergence, but in practice, slightly larger values of η can lead to faster convergence.

In Table. 1, we provide additional hyper-parameters that were used in our experiments. The parameter $simu$ denotes the number of Monte-Carlo simulations that were used to estimate the constraint in each iteration. T_{solve} is the total time (over K iterations) spent in solving the unconstrained POMDP using the SARSOP solver Kurniawati et al. [2008]. T_{simu} is the total time spent in simulating policies generated by the SARSOP solver. T_{total} is the overall computation time for that model.

Most of our models have a state size of 16 (4×4). However, the runtime (see Table 1) for these models is drastically different. This is because of two factors: (i) the DFA size and (ii) the complexity of the POMDP problem. The size of the DFA can be large for a complex task. This naturally scales up the state space of the product POMDP. SARSOP returns an alpha-vector policy Kurniawati et al. [2008]. When the POMDP is complex, the alpha-vector policy returned by SARSOP may have many alpha vectors. This would imply that whenever the agent has to make a decision, it needs to solve a fairly large maximization problem. This makes the simulations time-consuming.

Table 1: Performance Value and Hyper-parameters

Model	**Spec**	$\lvert S\rvert$	$\lvert Q\rvert$	$\mathcal{R}^{\mathscr{M}}(\bar{\mu})$	$\mathcal{R}^{f}(\bar{\mu})$	$1-\delta$	B	η	K	$simu$	T_{solve}	T_{simu}	T_{total}
$\mathscr{M}_1$	φ_1	16	3	1.72	0.75	0.75	5	2	100	200	142	3518	3661
$\mathscr{M}_2$	φ_1	64	3	0.95	0.70	0.70	8	2	50	100	17299	7825	25125
$\mathscr{M}_3$	φ_2	16	3	0.83	0.76	0.75	5	2	100	200	158	3614	3773
$\mathscr{M}_4$	φ_3	16	4	0.80	0.71	0.70	6	2	100	200	1893	14534	16428
$\mathscr{M}_5$	φ_4	16	4	0.83	0.71	0.70	6	2	100	200	368	8440	8809
$\mathscr{M}_6$	φ_5	16	4	1.01	0.79	0.80	10	2	100	200	109	718	828
$\mathscr{M}_7$	φ_6	16	10	4.28	0.82	0.80	25	2	50	100	5865	57833	63699
$\mathscr{M}_8$	φ_1	32	3	2.73	0.81	0.85	20	0.02	100	200	370	21676	22046
$\mathscr{M}_9$	φ_4	32	4	1.68	0.81	0.75	10	0.2	100	200	973	25618	26591