8 Supplementary Material

8.1 Background on Linearly Solvable MDP

Since the Reference-Based POMDP expands the Linearly Solvable (fully observed) MDPs[18, 19, 20] to POMDPs, for completeness, here, we summarise Linearly Solvable MDPs.

A standard infinite horizon MDP is specified by tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, R, \gamma \rangle$, where $\mathcal{S}$ and $\mathcal{A}$ are the state and action spaces, $\mathcal{T}(s, a, s')$ is the conditional probability function $P(s' \mid s, a)$ that specifies the probability the agent arrives at state $s' \in \mathcal{S}$ after performing action $a \in \mathcal{A}$ at state $s \in \mathcal{S}$, R is the reward function, and $\gamma \in (0, 1)$ is the discount factor. The solution to an MDP problem is a an optimal policy $\pi^* : \mathcal{S} \to \mathcal{A}$ that maximises the value function:

$$V^*(s) = \max_{a \in \mathcal{A}} \Big[R(s, a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s, a, s') V^*(s') \Big] \tag{30}$$

The works in [18, 19, 20] consider a class of MDPs where, the state space $\mathcal{S}$ is finite and for any states $s, s' \in \mathcal{S}$, there exists a one-step (not necessarily time-homogeneous) transition probability $p(s' \mid s)$ representing the *passive dynamics* of the system. They propose a new formulation of MDPs, called Linearly Solvable MDPs, to be specified by $\langle \mathcal{S}, p, r, \gamma \rangle$, where $r : \mathcal{S} \to \mathbb{R}$ is the reward function. A solution to the Linearly Solvable MDP is a stochastic state-to-state transition probability $u(\cdot \mid s)$ that maximises:

$$\begin{aligned} v(s) = \sup_{u(\cdot \mid s) \in \mathscr{U}_p(s)} \Big(r(s) - \mathrm{KL}\left(u(\cdot \mid s) \,\|\, p(\cdot \mid s) \right) \\ + \gamma \sum_{s' \in \mathcal{S}} u(s' \mid s) v(s') \Big). \end{aligned} \tag{31}$$

where $\mathscr{U}_p(s)$ is the set of admissible controls. An admissible control $u(\cdot \mid s)$ is one that prohibits state transitions that are not feasible under the passive dynamics $p(\cdot \mid s)$.

Now, suppose $w(s) := e^{v(s)}$ for any $s \in \mathcal{S}$, then (31) is equivalent to

$$w(s) = e^{r(s)} \sum_{s' \in \mathcal{S}} p(s' \mid s) w^{\gamma}(s'). \tag{32}$$

Moreover, the solution w^* to the above equation exists and is unique. The optimal stochastic transition to the equation (31) is given by

$$u^*(\cdot \mid s) = \frac{p(\cdot \mid s) {w^*}^{\gamma}(\cdot)}{D[{w^*}^{\gamma}](s)}. \tag{33}$$

where $D[{w^*}^{\gamma}](s) := \sum_{s' \in \mathcal{S}} p(s' \mid s) {w^*}^{\gamma}(s')$ is a normaliser. Intuitively, one can view w^* as the desirability score, so that (33) represents distorting the passive dynamics to transition dynamics that favour transitioning to states with higher desirability scores. Of course, w^* is not known a priori but it can be determined by iterating the Bellman backup operator given by (32). This computation essentially reduces to taking expectations under the reference dynamics, which can be computed faster than searching for the optimal value function in (30) directly.

A standard MDP can be *embedded* in a linearly solvable MDP. This implies that, for a given standard MDP problem $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, R, \gamma \rangle$, one can embed it as an instance of a linearly solvable MDP, use the above efficient machinery to determine the solution to the linearly solvable MDP $u^*(\cdot \mid s)$, and then choose the symbolic action $a^* \in \mathcal{A}$ such that $\mathcal{T}(s' \mid s, a^*)$ is as close as possible to $u^*(\cdot \mid s)$. Empirical results in [19] indicate that there is a close correspondence between the optimal value of the embedded standard MDP $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, R, \gamma \rangle$ and the optimal value of the linearly solvable MDP.

8.2 Proof of Lemma 3.1

Step 1. We first need to verify that a maximiser to the supremum in (10) exists. To this end, define $\mathcal{W}(b) := e^{\mathcal{V}(b)}$ for any $b \in \mathcal{B}$ and notice that the terms inside the supremum in the RHS of equation

(10) can be rewritten as

$$
\begin{aligned}
\sum_{a,o} \mathbb{U}(a,o\,|\,b)\Big[\mathcal{R}(b,a) - \log\Big\{\frac{\mathbb{U}(a,o\,|\,b)}{\mathbb{U}^{\mathfrak{p}}(a,o\,|\,b)}\Big\} \\
+ \gamma \sum_{a,o} \mathcal{V}\big(\tau(b,a,o)\big)\Big] \\
&= - \sum_{a,o} \mathbb{U}(a,o\,|\,b)\Big[\log\Big\{\frac{\mathbb{U}(a,o\,|\,b)}{\mathbb{U}^{\mathfrak{p}}(a,o\,|\,b)e^{\mathcal{R}(b,a)}\mathcal{W}^{\gamma}(\tau(b,a,o))}\Big\}\Big] \\
&= - \sum_{a,o} \mathbb{U}(a,o\,|\,b)\Big[\log\Big\{\frac{\mathbb{U}(a,o\,|\,b)\mathcal{D}[\mathcal{W}^{\gamma}](b)}{\mathbb{U}^{\mathfrak{p}}(a,o\,|\,b)e^{\mathcal{R}(b,a)}\mathcal{W}^{\gamma}(\tau(b,a,o))}\Big\} \\
&\qquad - \log\big\{\mathcal{D}[\mathcal{W}^{\gamma}](b)\big\}\Big] \\
&= - \mathrm{KL}\left(\mathbb{U}(\cdot,\cdot\,|\,b)\,\Big\|\, \frac{\mathbb{U}^{\mathfrak{p}}(\cdot,\cdot\,|\,b)e^{\mathcal{R}(b,a)}\mathcal{W}^{\gamma}\big(\tau(b,a,o)\big)}{\mathcal{D}[\mathcal{W}^{\gamma}](b)}\right) \\
&\qquad + \log\big\{\mathcal{D}[\mathcal{W}^{\gamma}](b)\big\}
\end{aligned} \tag{34}
$$

where $\mathcal{D}[\mathcal{W}^{\gamma}](b) := \sum_{a,o} \mathbb{U}^{\mathfrak{p}}(a,o\,|\,b)e^{\mathcal{R}(b,a)}\mathcal{W}^{\gamma}\big(\tau(b,a,o)\big)$ is a normalising factor. Only the KL divergence term in the last line above depends on $\mathbb{U}$. We know that the KL divergence is minimised when its two component distributions are identical. That is, when

$$
\mathbb{U}^{*}(a,o\,|\,b) = \frac{\mathbb{U}^{\mathfrak{p}}(a,o\,|\,b)e^{\mathcal{R}(b,a)}\mathcal{W}^{\gamma}\big(\tau(b,a,o)\big)}{\mathcal{G}(b)}. \tag{35}
$$

It is clear that $\mathbb{U}^{*}$ belongs to the space $\mathscr{U}_{\mathfrak{p}}(b)$ since $\mathbb{U}^{\mathfrak{p}}(a,o\,|\,b) = 0$ implies that $\mathbb{U}^{*}(a,o\,|\,b) = 0$ too. Therefore, we conclude that the supremum is attained and that $\mathbb{U}^{*}$ is the maximiser.

Step 2. Now, we can essentially repeat the classical argument from Ross [15] (see e.g. Theorem 6.5). Namely, let $\Phi : \mathbb{B}(\mathcal{B}) \to \mathbb{B}(\mathcal{B})$ be the Bellman backup operator

$$
\begin{aligned}
\Phi\mathcal{V}(b) := \sup_{\mathbb{U}\in\mathscr{U}_{\mathfrak{p}}(b)} \Big(\mathcal{R}(b,\mathbb{U}) - \mathrm{KL}(\mathbb{U}\,\|\,\mathbb{U}^{\mathfrak{p}}) \\
+ \gamma\, \mathbb{E}_{\mathbb{U}}\big[\mathcal{V}\big(\tau,\cdot,\cdot)\big)\big]\Big) \quad \forall b \in \mathcal{B}
\end{aligned} \tag{36}
$$

where, for brevity, we write

$$
\mathrm{KL}(\mathbb{U}\,\|\,\mathbb{U}^{\mathfrak{p}}) := \mathrm{KL}\left(\mathbb{U}(\cdot,\cdot\,|\,b)\,\|\,\mathbb{U}^{\mathfrak{p}}(\cdot,\cdot\,|\,b)\right) \tag{37}
$$

and

$$
\mathbb{E}_{\mathbb{U}}\big[\mathcal{V}\big(\tau,\cdot,\cdot)\big)\big] := \sum_{a,o} \mathbb{U}(a,o\,|\,b)\mathcal{V}\big(\tau(b,a,o)\big). \tag{38}
$$

We want to show that Φ is a contraction. For any $b \in \mathcal{B}$ and any $\mathcal{V}_1, \mathcal{V}_2 \in \mathbb{B}(\mathcal{B})$,

$$
\begin{aligned}
(\Phi\mathcal{V}_1)(b) - (\Phi\mathcal{V}_2)(b) \\
&= \sup_{\mathbb{U}\in\mathscr{U}_{\mathfrak{p}}(b)} \Big(\mathcal{R}(b,\mathbb{U}) - \mathrm{KL}(\mathbb{U}\,\|\,\mathbb{U}^{\mathfrak{p}}) + \gamma\, \mathbb{E}_{\mathbb{U}}\big[\mathcal{V}_1\big(\tau,\cdot,\cdot)\big)\big]\Big) \\
&\quad - \sup_{\tilde{\mathbb{U}}\in\mathscr{U}_{\mathfrak{p}}(b)} \Big(\mathcal{R}(b,\tilde{\mathbb{U}}) - \mathrm{KL}(\tilde{\mathbb{U}}\,\|\,\mathbb{U}^{\mathfrak{p}}) + \gamma\, \mathbb{E}_{\tilde{\mathbb{U}}}\big[\mathcal{V}_2\big(\tau,\cdot,\cdot)\big)\big]\Big) \\
&\le \Big(\mathcal{R}(b,\mathbb{U}^{*}) - \mathrm{KL}(\mathbb{U}^{*}\,\|\,\mathbb{U}^{\mathfrak{p}}) + \gamma\, \mathbb{E}_{\mathbb{U}^{*}}\big[\mathcal{V}_1\big(\tau,\cdot,\cdot)\big)\big]\Big) \\
&\quad - \Big(\mathcal{R}(b,\mathbb{U}^{*}) - \mathrm{KL}(\mathbb{U}^{*}\,\|\,\mathbb{U}^{\mathfrak{p}}) + \gamma\, \mathbb{E}_{\mathbb{U}^{*}}\big[\mathcal{V}_2\big(\tau,\cdot,\cdot)\big)\big]\Big) \\
&= \gamma \sum_{a,o} \mathbb{U}^{*}(a,o\,|\,b)\Big[\mathcal{V}_1\big(\tau(b,a,o)\big) - \mathcal{V}_2\big(\tau(b,a,o)\big)\Big] \\
&\le \gamma\,\|\mathcal{V}_1 - \mathcal{V}_2\|_{\infty}
\end{aligned} \tag{39}
$$

where $\mathbb{U}^*$ is the maximiser of

$$\mathcal{R}(b,\mathbb{U}) - \mathrm{KL}(\mathbb{U}\,\|\,\mathbb{U}^{\mathfrak{p}}) + \gamma\,\mathbb{E}_{\mathbb{U}}\big[\mathcal{V}_1\big(\tau,\cdot,\cdot\big)\big]. \tag{40}$$

Reversing the roles of $\mathcal{V}_1$ and $\mathcal{V}_2$ and using the fact that $b \in \mathcal{B}$ is arbitrary, we conclude that

$$\|\Phi\mathcal{V}_1 - \Phi\mathcal{V}_2\|_\infty \leq \gamma\,\|\mathcal{V}_1 - \mathcal{V}_2\|_\infty. \tag{41}$$

Since we assumed that $\gamma \in (0,1)$, we conclude that Φ is a contraction.

8.3 Proof of Theorem 3.1

Repeating the argument in Step 1 of 8.2, we see that the Bellman equation (10) reduces to

$$\begin{aligned}\mathcal{V}(b) &= \log\big[\mathcal{D}[w^\gamma](b)\big]\\ &= \log\Big[\sum_{a,o}\mathbb{U}^{\mathfrak{p}}(a,o\,|\,b)e^{\mathcal{R}(b,a)}\mathcal{W}^\gamma\big(\tau(b,a,o)\big)\Big]\end{aligned} \tag{42}$$

which, after taking exponents, justifies the equivalence to (12). Given this equivalence and Lemma 3.1, it is clear that (12) has a unique solution. To be more explicit, suppose for a contradiction that (12) does not have exactly one solution (up to $\|\cdot\|$-equivalence of solutions). Then by the equivalence between the two Bellman equations, (10) would either have no solutions or more than one solution which contradicts the existence and uniqueness guaranteed by Lemma 3.1. Finally, (13) follows from the form of the maximiser at each Bellman step.

8.4 Proof of Proposition 3.1

For brevity, we will fix a $b \in \mathcal{B}$ and drop it from our notation. Also write $\mathfrak{u} = \mathfrak{u}(\cdot\,|\,b)$ and $\mathfrak{u}_a = \mathfrak{u}(a\,|\,b)$. The Lagrangian for the constrained problem is

$$\begin{aligned}\mathcal{L}(\mathfrak{u},\lambda) = \sum_{a,o}P(o\,|\,a)\mathfrak{u}_a\log\Big[\frac{P(o\,|\,a)\mathfrak{u}_a}{\mathbb{U}^*(a,o)}\Big]\\ + \lambda\Big(\sum_a\mathfrak{u}_a - 1\Big).\end{aligned} \tag{43}$$

We require, in addition, that the minimiser $\mathfrak{u}^*$ (which exists due to the Weierstrass extreme value theorem) is such that $\mathfrak{u}^*_a \geq 0$ for each $a \in \mathcal{A}$. The first order necessary conditions gives

$$\mathfrak{u}_a = e^{-(1+\lambda)}\exp[-\Pi(a)] \quad \forall a \in \mathcal{A} \tag{44}$$

and the constraint equation gives

$$1 = \sum_a\mathfrak{u}_a = e^{-(1+\lambda)}\sum_a\exp[-\Pi(a)]. \tag{45}$$

Hence the only candidate for the minimiser is $\mathfrak{u}^*$ such that

$$\mathfrak{u}^*_a = \frac{\exp[-\Pi(a)]}{\sum_{\hat{a}\in\mathcal{A}}\exp[-\Pi(\hat{a})]} \quad \forall a \in \mathcal{A}. \tag{46}$$

The Hessian of $\mathcal{L}$ is positive definite for any λ and $\mathfrak{u} \in \Delta(\mathcal{A})$, so we conclude that $\mathfrak{u}^*$ is a minimiser. Finally, that $\mathfrak{u}^*_a \geq 0$ for every $a \in \mathcal{A}$ is clear from (46).

8.5 Proof of Proposition 4.1

Proof. Fix an $\hat{a} \in \mathcal{A}$ and $b \in \mathcal{B}$. Clearly $\mathfrak{p}$ as defined in (24) has full support on $\mathcal{A}$. Thus, if we set

$$\mathfrak{u}^{\hat{a}}(a\,|\,b) := \begin{cases}1, & a = \hat{a}\\ 0, & \text{otherwise}\end{cases} \tag{47}$$

and

$$\mathbb{U}^{\hat{a}}(a,o\,|\,b) := P(o\,|\,\hat{a},b)\mathfrak{u}^{\hat{a}}(a\,|\,b) \in \mathscr{U}_{\mathfrak{p}} \tag{48}$$

for any $\mathfrak{u}^{\hat{a}} \in \Delta(\mathcal{A})$ then the constraint (21) is satisfied trivially. Straightforward computations show that the constraint (22) is satisfied by $\mathbb{U}^{\hat{a}}(a,o\,|\,b)$ with $(\rho,\mathfrak{p})$ as defined in (23) and (24). □

8.6 Proof of Proposition 4.2

Proof. For a fixed $\delta > 0$, the matrix equation (25) has a solution $\mathbf{x}_\delta$ if and only if

$$\sum_{s \in \mathcal{S}} b(s)\rho_\delta(s) = \ell(b) \quad \forall b \in C_\delta. \tag{49}$$

As $\delta \downarrow 0$ the δ-cover converges to the set $\mathscr{R}$ which proves that the pair $(\rho, \mathfrak{p})$ is an embedding. $\square$

8.7 Algorithm RefSolver

Algorithm 1 RefSolver

parameters:
$\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma \rangle$
max-depth
max-rollout-depth
α $\triangleright$ expl const $= 1 - \alpha$
require: $\gamma \in (0,1), \alpha \in [0,1)$

PRE-PROCESS (OFFLINE)

$\pi^{\text{FO}} \leftarrow$ Generate-FO-Policy$(\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma \rangle)$

RUNTIME (ONLINE)

procedure Plan-And-Execute(h)
  repeat
    if $h = \emptyset$ **then**
      $s \sim \mathcal{I}$
    else
      $s \sim \mathcal{B}(h)$
    end if
    Simulate$(s, h, 0)$
  until Timeout()
  return Optimal-Stochastic-Policy(h)
end procedure

procedure Rollout(s, h, depth)
  $a \leftarrow \pi^{\text{FO}}(s)$
  if $s \in \mathcal{G}$ or depth $>$ max-rollout-depth **then**
    return $\mathcal{R}(s, a)$
  end if
  $(s', o, \mathcal{R}) \sim \mathscr{G}(s, a)$ $\triangleright$ generative model
  return $\mathcal{R}(s, a)+$Rollout$(s', hao, \text{depth}+1)$
end procedure

procedure Simulate(s, h, depth)
  if $s \in \mathcal{G}$ or depth $>$ max-depth **then**
    return $\exp($Rollout$(s, h, \text{max-depth}))$
  end if
  $\mathcal{B}(h) \leftarrow \mathcal{B}(h) \cup \{s\}$
  $N(h) \leftarrow N(h) + 1$
  $X \sim$ Bernoulli(α)
  $a \leftarrow \pi^{\text{FO}}(s) I_{\{X=1\}} + (1-\alpha) \times I_{\{X=0\}}$
  $(s', o, \mathcal{R}) \sim \mathscr{G}(s, a)$
  $N(ha) \leftarrow N(ha) + 1$
  $\widehat{\mathcal{R}}(ha) \leftarrow \widehat{\mathcal{R}}(ha) + \frac{\mathcal{R}(s,a) - \widehat{\mathcal{R}}(ha)}{N(ha)}$
  $\widehat{\mathcal{W}} \leftarrow \widehat{\mathcal{W}} + \frac{e^{\widehat{\mathcal{R}}(ha)}\text{Simulate}(s', hao, \text{depth}+1) - \widehat{\mathcal{W}}(h)}{N(h)}$
  return $\widehat{\mathcal{W}}(h)^\gamma$
end procedure

procedure Optimal-Stochastic-Policy(h)
  $\mathcal{D} \leftarrow 0$ $\triangleright$ Normaliser
  for $a \in \mathcal{A}$ and $o \in \mathcal{O}$ **do**
    if $hao \notin T$ **then**
      $\widehat{\mathbb{U}}^*(hao) = 0$
    else
      $\widehat{\mathbb{U}}^*(hao) \leftarrow \frac{N(hao)}{N(h)} e^{\widehat{\mathcal{R}}(ha)} \mathcal{W}^\gamma(hao)$
    end if
    $\mathcal{D} \leftarrow \mathcal{D} + \widehat{\mathbb{U}}^*(hao)$
  end for
  for $a \in \mathcal{A}$ **do**
    $\Pi(a) \leftarrow \frac{N(hao)}{N(ha)} \log \left[\frac{N(hao)\mathcal{D}}{N(ha)\widehat{\mathbb{U}}^*(hao)} \right]$
  end for
  $\mathfrak{u}^* \leftarrow \{a : \exp[-\Pi(a)]/\mathcal{D}^{\Pi}\}$
  return Random-Sample$(\mathfrak{u}^*)$
end procedure

8.8 Details of Navigation1 Scenario

The robot can move in the four cardinal directions with 0.1 probability of actuator failure. If the realised movement leads to a collision with an obstacle or the edge of the map, no movement occurs and the robot remains in its current position. If the robot's true state is a landmark, the robot receives a position reading uniformly in the 9×9 grid around the robot's true state. Outside the landmarks, the robot receives no observation. The robot receives a penalty of -100 for entering a danger zone and a reward of +300 for entering a goal state. In both cases, the problem terminates. Every other state incurs a reward of -1. The discount parameter was 0.99. The robot's initial belief was equally distributed between two initial positions that were uniformly sampled from the southern-most row of the map.

8.9 Details of Navigation2 Scenario

Similar to Navigation1, the robot's action space consists of moves anywhere in the four cardinal directions NORTH, SOUTH, EAST, WEST. To simulate noise in the robot's actuator's, actions fail with 0.1 probability, and if this occurs, the robot moves randomly in a direction orthogonal to the one specified. If the realised movement leads to a collision with an obstacle or the edge of the map, no movement occurs and the robot remains in its current position. If the robot's true state is a landmark, the robot receives a position reading uniformly in the 9×9 grid around the robot's true state. Otherwise, the robot receives no observation. The robot receives a reward of +600 for being in a goal state, and -3 for being in any other state. The discount parameter was $\gamma = 0.99$. The robot's initial belief was equally distributed between two initial positions that were uniformly sampled from the southern-most row of the map.

8.10 Source code

We also include the source codes for REFSOLVER, which is developed on top of pomdp_py.