




Enabling Markovian Representations under Imperfect Information

Francesco Belardinelli^{1,2}^a, Borja G. León¹^b, and Vadim Malvone³^c.

¹*Department of Computing, Imperial College London, London, United Kingdom*

²*IBISC, Université d'Evry, Evry, France*

³*INFRES, Télécom Paris, Paris, France*

{francesco.belardinelli, b.gonzalez-leon19}@imperial.ac.uk, vadim.malvone@telecom-paris.fr

Keywords: Markov Decision Processes, Partial Observability, Extended Partially Observable Decision Process, non-Markovian Rewards

Abstract: Markovian systems are widely used in reinforcement learning (RL), when the successful completion of a task depends exclusively on the last interaction between an autonomous agent and its environment. Unfortunately, real-world instructions are typically complex and often better described as non-Markovian. In this paper we present an extension method that allows solving partially-observable non-Markovian reward decision processes (PONMRDPs) by solving equivalent Markovian models. This potentially facilitates Markovian-based state-of-the-art techniques, including RL, to find optimal behaviours for problems best described as PONMRDP. We provide formal optimality guarantees of our extension methods together with a counterexample illustrating that naive extensions from existing techniques in fully-observable environments cannot provide such guarantees.


1 Introduction


One of the major long-term goals of artificial intelligence is to build autonomous agents that execute temporally extended human instructions (Oh et al., 2017; Lake, 2019; Hill et al., 2021; Abate et al., 2021; León et al., 2022). Markov Decision Processes (MDPs) are a widely-used mathematical model for sequential decision-making (Mnih et al., 2015; Bellemare et al., 2017; Hill et al., 2020). MDPs are particularly relevant for reinforcement learning (RL) (Sutton and Barto, 2018), where an agent attempts to maximise a reward signal given by the environment according to a predefined goal. RL has proved successful in solving various challenging real-world scenarios, after various breakthroughs (Silver et al., 2017; Vinyals et al., 2019; Bellemare et al., 2020). However, MDPs – and consequently RL – rely on the Markovian assumption, which intuitively says that the effects of an action depend exclusively on the state where it is executed. Unfortunately, many real-world instructions are naturally described as non-Markovian. For instance, we may ask our autonomous vehicle to *drive us home eventually, without hitting anything in the process*, which is


a temporally-extended, non-Markovian specification.

The problem of solving non-Markovian problems through Markovian techniques was first tackled in (Bacchus et al., 1996). Therein, the authors solved an MDP with non-Markovian rewards (NMRDP) by generating an equivalent MDP so that an optimal policy in the latter is also optimal for the former. Building on this, (Brafman et al., 2018; Giacomo et al., 2019; León and Belardinelli, 2020) extended the scope and applications of this method to solving temporally-extended goals. Still, this line of works focuses on fully observable environments, where the agent has perfect knowledge of the current state of the system. Assuming complete (state) knowledge of the environment is often unrealistic or computationally costly (Badia et al., 2020; Samvelyan et al., 2019), and while there exists successful empirical works applying (Bacchus et al., 1996) or similar extensions in scenarios of imperfect knowledge (Icarte et al., 2019; León et al., 2020), there is no theoretical analysis on which conditions these extensions should fulfill to guarantee solving the original non-Markovian model under imperfect state knowledge (also called partial observability or imperfect information).

Contribution. In this work we present a novel method to solve a partially-observable NMRDP (PONMRDP) by building an equivalent partially-observable MDP (POMDP), while providing formal

^a <https://orcid.org/0000-0002-7768-1794>

^b <https://orcid.org/0000-0002-5990-8684>

^c <https://orcid.org/0000-0001-6138-4229>

guarantees that any policy that is optimal in the latter has a corresponding policy in the former that is also optimal. We also prove that naive extensions of existing methods for fully-observable models can induce optimal policies that are not applicable in the original non-Markovian problem.

2 Partially Observable Models

In this section we recall standard notions on partially observable (p.o.) Markov decision processes and their non-Markovian version. Furthermore, for both models we present their belief-state counterparts, policies, and policy values. Given an element U in an MDP, \bar{U} denotes its non-Markovian version, and U_b denotes its belief version. Given a tuple \vec{w} , we denote its length as $|\vec{w}|$, and its i -th element as \vec{w}_{i-1} . Then, $\text{last}(\vec{w}) = \vec{w}_{|\vec{w}|-1}$ is the last element in \vec{w} . For $i \leq |\vec{w}| - 1$, let $\vec{w}_{\geq i}$ be the suffix $w_i, \dots, w_{|\vec{w}|-1}$ of \vec{w} starting at w_i and $\vec{w}_{\leq i}$ its prefix w_0, \dots, w_i . Moreover, we denote with $\vec{w} \cdot \vec{w}'$ the concatenation of tuples \vec{w} and \vec{w}' . Finally, given a set V , we denote with V^+ the set of all non-empty sequences on V .

Definition 1 (PONMRDP). A p.o. non-Markovian reward decision process is a tuple $\bar{\mathcal{M}} = \langle S, A, T, \bar{R}, Z, O, \gamma \rangle$, where:

- S is a finite set of states.
- A is the finite set of actions.
- $T : S \times A \times S \rightarrow [0, 1]$ is the transition probability function that returns the probability $T(s' | s, a)$ of transitioning to the successor state s' , given the previous state s and action a taken by the agent.
- \bar{R} is the reward function defined as $\bar{R} : (S \cdot A)^+ \cdot S \rightarrow \mathbb{R}$.
- Z is the set of observations.
- $O : S \times A \times S \rightarrow Z$ is the observation function that given the current state s and action a , draws an observation z , based on the successor state s' .
- $\gamma \in (0, 1]$ is the discount factor.

Note that, we consider a deterministic observation function for simplicity of presentation. However, the environment is still stochastic due to the transition function. We adopt this modelling since it is the standard approach for RL algorithms working in p.o. scenarios (Rashid et al., 2018; Icarte et al., 2019; Vinyals et al., 2019; Zhao et al., 2021)

A trajectory $s_0, a_1, \dots, s_n \in (S \cdot A)^+ \cdot S$, denoted as \vec{s} , is a finite (non-empty) sequence of states and actions, ending in a state, where for each $1 \leq i \leq n$, $T(s_i | s_{i-1}, a_i) > 0$. By Def. 1, a p.o. Markov decision process (POMDP) is a PONMRDP where the reward

function only depends on the last transition. That is, the reward function is a function $R : S \times A \times S \rightarrow \mathbb{R}$.

Because the agent cannot directly observe the state of the environment, she has to make decisions under uncertainty about its actual state. Then the agent updates her beliefs by interacting with the environment and receiving observations. A belief state b is a probability distribution over the set of states, that is, $b : S \rightarrow [0, 1]$ such that $\sum_{s \in S} b(s) = 1$.

Definition 2 (Belief update). Given a belief state b , action a , and observation z , we define an update operator ρ such that $b' = \rho(b, a, z)$ iff for every state $s' \in S$,

$$b'(s') = \begin{cases} \eta \sum_{s \in S} T(s' | s, a) b(s) & \text{if } O(s, a, s') = z; \\ 0 & \text{otherwise.} \end{cases}$$

where $b(s)$ denotes the probability that the environment is in state s ; $\eta = 1/\text{Pr}(z | b, a)$ is the normalization factor for $\text{Pr}(z | b, a) = \sum_{\{s' \in S | O(s, a, s') = z\}} \sum_{s \in S} T(s' | s, a) b(s)$.

Now, we have all the ingredients to define a belief non-Markovian reward decision process.

Definition 3 (BNMRDP). Given a PONMRDP $\bar{\mathcal{M}}$, the corresponding belief NMRDP is a tuple $\bar{\mathcal{M}}_b = \langle B, A, T_b, \bar{R}_b, \gamma \rangle$ where:

- A and γ are defined as in Def. 1.
- B is the set of belief states over the states of $\bar{\mathcal{M}}$.
- $T_b : B \times A \times B \rightarrow [0, 1]$ is the belief state transition function defined as:

$$T_b(b, a, b') = \sum_{z \in Z} \text{Pr}(b' | b, a, z) \text{Pr}(z | b, a)$$

$$\text{where } \text{Pr}(b' | b, a, z) = \begin{cases} 1 & \text{if } b' = \rho(b, a, z); \\ 0 & \text{otherwise.} \end{cases}$$

- the reward function $\bar{R}_b : (B \cdot A)^+ \cdot B \rightarrow \mathbb{R}$ is defined as:

$$\bar{R}_b(b_0, a_1, \dots, b_n) = \sum_{s_0, \dots, s_n \in S} b_1(s_0) \dots b_n(s_n) \bar{R}(s_0, a_1, \dots, s_n)$$

By Def. 3, given a POMDP \mathcal{M} , the corresponding belief MDP is a tuple $\mathcal{M}_b = \langle B, A, T_b, R_b, \gamma \rangle$, where The reward function $R_b : B \times A \times B \rightarrow \mathbb{R}$ is defined as

$$R_b(b, a, b') = \sum_{s, s' \in S} b(s) b'(s') R(s, a, s')$$

Remark 1. In what follows we assume w.l.o.g. an initial distribution b_0 . Such belief state b_0 can be generated by assuming an initial state s_0 and suitable auxiliary transitions over the states in b_0 .

We now define policies, policy values, and optimal policies in a similar fashion to (Sutton and Barto, 2018) for all frameworks described above.

Definition 4 (Non-Markovian Policy). A non-Markovian policy $\bar{\pi} : (S \cdot A)^+ \cdot S \rightarrow A$ is a function from trajectories to actions. The value $v^{\bar{\pi}}(\vec{s})$ of a trajectory \vec{s} following a non-Markovian policy $\bar{\pi}$ is defined as:

$$v^{\bar{\pi}}(\vec{s}) = \sum_{s' \in S} T(s' | \text{last}(\vec{s}), \bar{\pi}(\vec{s})) \left[\bar{R}(\vec{s}, \bar{\pi}(\vec{s}), s') + \gamma v^{\bar{\pi}}(\vec{s} \cdot s') \right]$$

An optimal non-Markovian policy $\bar{\pi}^*$ is one that maximizes the expected value for any given trajectory $\vec{s} \in (S \cdot A)^+ \cdot S$, that is, for all \vec{s} , $v^{\bar{\pi}^*}(\vec{s}) \doteq \max_{\bar{\pi}} v^{\bar{\pi}}(\vec{s})$.

By Def. 4, a Markovian policy $\pi : S \rightarrow A$ is a non-Markovian policy that only depends on the last visited state. Then, the value $v^\pi(s)$ of a Markovian policy π at state s is:

$$v^\pi(s) = \sum_{s' \in S} T(s' | s, \pi(s)) \left[R(s, \pi(s), s') + \gamma v^\pi(s') \right]$$

Finally, an optimal Markovian policy π^* is such that for all $s \in S$, $v^{\pi^*}(s) \doteq \max_{\pi} v^\pi(s)$.

Definition 5 (non-Markovian Belief Policy). A non-Markovian belief policy $\bar{\pi}_b : (B \cdot A)^+ \cdot B \rightarrow A$ is a function from belief trajectories to actions. The value $v^{\bar{\pi}_b}(\vec{b})$ of a non-Markovian belief policy $\bar{\pi}_b$ for a trajectory \vec{b} of belief states is defined as:

$$v^{\bar{\pi}_b}(\vec{b}) = \sum_{b' \in B} T_b(b' | \text{last}(\vec{b}), \bar{\pi}_b(\vec{b})) \left[\bar{R}_b(\vec{b}, \bar{\pi}_b(\vec{b}), b') + \gamma v^{\bar{\pi}_b}(\vec{b} \cdot b') \right]$$

An optimal non-Markovian belief policy in a p.o. model is one that achieves the maximum value for any trajectory of belief state $\vec{b} \in (B \cdot A)^+ \cdot B$: $v^{\bar{\pi}_b^*}(\vec{b}) \doteq \max_{\bar{\pi}_b} v^{\bar{\pi}_b}(\vec{b})$.

By Def. 5, a Markovian belief policy $\pi_b : B \rightarrow A$ is a non-Markovian belief policy that only depends on the last belief state. Then, the value $v^{\pi_b}(b)$ of a Markovian belief policy π_b at belief state b is given as:

$$v^{\pi_b}(b) = \sum_{b' \in B} T_b(b' | b, \pi_b(b)) \left[R_b(b, \pi_b(b), b') + \gamma v^{\pi_b}(b') \right]$$

Finally, an optimal Markovian belief policy is such that for all $b \in B$, $v^{\pi_b^*}(b) \doteq \max_{\pi_b} v^{\pi_b}(b)$.

Notice that, in a PONMRDP, i.e., a model where the rewards depend on trajectories of states and actions, optimal policies might also be Markovian as in the case of the counterexample in Sec 4.

In following sections, when not explicitly stated, we will assume an element to be Markovian, e.g. we will refer to $\bar{\pi}$ as a non-Markovian policy and π as a policy.

3 Extended POMDPs

In this section, we describe a state-space extension method to generate a correspondence between POMDPs and PONMRDPs. In particular, we extend the method originally proposed in (Bacchus et al., 1996) to partial observability.

First of all, we show how to generate a belief trajectory from a state trajectory.

Definition 6 (Belief trajectory). Given a model \mathcal{M} , let $s_0 \xrightarrow{a_1, p_1} s_1 \cdots s_{n-1} \xrightarrow{a_n, p_n} s_n$ be a trajectory, where $p_i = T(s_i | s_{i-1}, a_i)$, for all $1 \leq i \leq n$. We define the corresponding belief trajectory $b_0 \xrightarrow{a_1, p'_1} b_1 \cdots b_{n-1} \xrightarrow{a_n, p'_n} b_n$, where $p'_i = T_b(b_i | b_{i-1}, a_i)$ and $b_i = \rho(b_{i-1}, a_i, z_i)$ for all $0 < i \leq n$ ¹.

Note that, in Def. 6, we make use of observations. In particular, the observation z_i is generated by the observation function and the trajectory, i.e. $z_i = O(s_{i-1}, a_i, s_i)$, and it is used in the belief update function to generate the next belief state.

Now, we can define our notion of expansion.

Definition 7 (Expansion). A POMDP $\mathcal{M} = \langle ES, A, T_{ES}, R, EZ, O_{ES}, \gamma \rangle$ is an expansion of a PONMRDP $\bar{\mathcal{M}} = \langle S, A, T, \bar{R}, Z, O, \gamma \rangle$, if there exist functions $\tau : ES \rightarrow S$, $\sigma : S \rightarrow ES$, $\tau' : EZ \rightarrow Z$, and $\sigma' : Z \rightarrow EZ$ such that:

1. For all $s \in S$, $\tau(\sigma(s)) = s$.
2. For all $s_1, s_2 \in S$ and $es_1 \in ES$, if $T(s_2 | s_1, a) > 0$ and $\tau(es_1) = s_1$, then there exists a unique $es_2 \in ES$ such that $\tau(es_2) = s_2$ and $T_{ES}(es_2 | es_1, a) = T(s_2 | s_1, a)$.
3. For all $z \in Z$, $\tau'(\sigma'(z)) = z$.
4. For all $z \in Z$, $s, s' \in S$, and $es, es' \in ES$, if $O(s, a, s') = z$, $\tau(es) = s$, and $\tau(es') = s'$, then there exist a unique $ez \in EZ$ such that $\tau'(ez) = z$ and $O_{ES}(es, a, es') = ez$.
5. For any trajectory s_0, \dots, s_n in $\bar{\mathcal{M}}$ with corresponding belief trajectory b_0, \dots, b_n as per Def. 6, the trajectory es_0, \dots, es_n in \mathcal{M} that satisfies $\tau(es_i) = s_i$ for each $0 \leq i \leq n$ and $\sigma(s_0) = es_0$, generates a belief trajectory eb_0, \dots, eb_n such that $\bar{R}_b(b_0, \dots, b_{n-1}, a, b_n) = R_b(eb_{n-1}, a, eb_n)$.

Intuitively the extended state es and observation ez such that $\tau(es) = s$ and $\tau'(ez) = z$, can be thought of as labelled by $s \circ l$ and $z \circ l'$, where $s \in S$ is the base state (i.e. a state of $\bar{\mathcal{M}}$), $z \in Z$ is the base observation

¹As detailed in Remark 1, we define an auxiliary initial state s_0 with suitable auxiliary transitions. Consequently b_0 is assumed to be the distribution assigning 1 to s_0 and 0 to all other states.

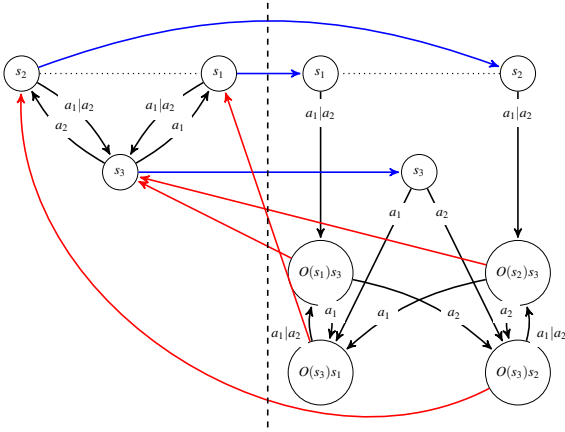


Figure 1: The models for the PONMRDP (left side) and an extended POMDP (right side) following Def. 7. Red lines denote some $\tau(es)$ transformations while blue lines we illustrate the cases in which $\sigma(s)$ is only the current state.

(i.e. an observation from $\overline{\mathcal{M}}$), and l and l' are labels that distinguish es and ez from other extensions of the same base elements. Figure 1 illustrates a POMDP and an extended POMDP that follow Def. 7.

The most important items in Def. 7 are points 2, 4 and 5. Points 2 and 4 ensure that both \mathcal{M} and $\overline{\mathcal{M}}$ are equivalent regarding their respective base elements in both state and observation dynamics. Point 5 asserts equivalence in reward structure from the agent's perspective. Note that, differently from the expansions in fully observable models (Bacchus et al., 1996; Brafman et al., 2018), it is not enough that equivalent trajectories have the same rewards. Here, we require that they induce equivalent *beliefs* rewards (for details see Sec. 4).

An optimal agent, i.e., an agent that always follows an optimal policy, working in $\overline{\mathcal{M}}$ would construct a BNMRDP, where it follows the policy that maximizes the value of each belief state according to \overline{R}_b . Similarly, an optimal agent working in \mathcal{M} would construct an extended BMDP maximizing the discounted expectation of R_b .

Since belief states are generated exclusively from transition probabilities and observations given states and actions, clauses 2 and 4 induce a similar behavior between equivalent trajectories of belief states to what is presented in (Bacchus et al., 1996) for trajectories of states.

Lemma 1. *Let the POMDP \mathcal{M} be an extension of the PONMRDP $\overline{\mathcal{M}}$. Given a belief trajectory \vec{b} in \mathcal{M} : $b_0 \xrightarrow{a_1, p_{b_1}} b_1 \cdots b_{n-1} \xrightarrow{a_n, p_{b_n}} b_n$, there is a trajectory \vec{eb} in $\overline{\mathcal{M}}$ defined as: $eb_0 \xrightarrow{a_1, p_{b_1}} eb_1 \cdots eb_{n-1} \xrightarrow{a_n, p_{b_n}} eb_n$ where $eb_i(es_i) = b(\tau(es_i))$ for all $0 \leq i \leq n$.*

We say that \vec{b} and \vec{eb} are weakly corresponding belief trajectories.

Proof. From clause 2 immediately follows that for any trajectory \vec{s} in $\overline{\mathcal{M}}$

$$\vec{s} = s_0 \xrightarrow{a_1, p_1} s_1 \cdots s_n \xrightarrow{a_n, p_n} s_n$$

there is a trajectory \vec{es} in \mathcal{M} of similar structure

$$\vec{es} = es_0 \xrightarrow{a_1, p_{e_1}} s_1 \cdots s_n \xrightarrow{a_n, p_{e_n}} s_n$$

where $p_{ei} = p_i$ and $\tau(es_i) = s_i$, for all $0 \leq i \leq n$. In this case we say that \vec{s} and \vec{es} are *weakly corresponding* trajectories. Symmetrically, clause 4 assures that these weakly correspondent real states trajectories will generate as well weakly correspondent observation trajectories. As a consequence, it follows immediately that these trajectories will generate weakly correspondent belief trajectories. \square

Given Lemma 1 we can define strong correspondence.

Definition 8. *Let \vec{b} and \vec{eb} be weakly corresponding trajectories with initial belief states b_0 and eb_0 , respectively. We say that \vec{b} and \vec{eb} are strongly correspondent when eb_0 is an initial belief state.*

This means that when the transformation induced by σ , i.e., $eb_i(\sigma(s_0)) = b_i$ is a trajectory containing only the current belief state, i.e, the first state of the trajectory in the non-Markovian model is the first state in the environment, we have strongly correspondent belief trajectories. Note that clause 5 requires that strongly correspondent belief trajectories have the same rewards.

Now we can introduce corresponding policies.

Definition 9 (Corresponding Policy). *Let π_b be a belief policy for expansion $\overline{\mathcal{M}}$. The corresponding belief policy $\overline{\pi}_b$ for the PONMRDP \mathcal{M} is defined as $\overline{\pi}_b(\vec{b}) = \pi'_b(\text{last}(\vec{eb}))$, where \vec{eb} is the strongly corresponding trajectory for \vec{b} .*

Given the expanded MDP from Def. 7 and corresponding policies as in Def. 9, we can now present the following result.

Proposition 1. *For every policy π_b in expansion $\overline{\mathcal{M}}$, corresponding policy $\overline{\pi}_b$ in PONMRDP \mathcal{M} , we have that $v^{\overline{\pi}_b}(\vec{b}) = v^{\pi_b}(\text{last}(\vec{eb}))$, where \vec{eb} and \vec{b} are strongly corresponding trajectories.*

Proof. This is evident since corresponding policies will generate the same actions for any correspondent belief trajectory and predict the same expected discounted return given the equivalent transition and observation dynamics from clauses 1-4 in Def. 7 and that clause 5 imposes the same belief rewards. \square

Consequently we can find optimal policies for the PONMRDP by working on the POMDP instead.

Corollary 1. *Let π_b an optimal policy for expansion \mathcal{M} . Then the corresponding policy $\bar{\pi}_b$ is optimal for the PONMRDP $\bar{\mathcal{M}}$.*

Thus, given an optimal policy in the extended POMDP one can easily obtain an optimal solution for the original PONMRDP. As in previous fully-observable approaches, there is no need to generate $\bar{\pi}_b$ explicitly and, instead, the agent can run with π_b while assuming that the underlying model is \mathcal{M} .

4 A Counterexample to a Naive Extension

As anticipated in the introduction, we illustrate now the relevance of item 5 in Def. 7. In particular, equivalences under perfect knowledge (Bacchus et al., 1996; Brafman et al., 2018) require only that strongly correspondent trajectories of states have the same rewards. Here we show that a naive extension of the definition of expansion in (Bacchus et al., 1996), i.e., one that requires equivalent dynamics on states and observations, and enforces equivalence between rewards on state trajectories only, may induce policies that are unfeasible in the non-Markovian model. Formally, consider a variant of Def. 7, where item 5 only is replaced as follows:

5'. For every trajectory s_0, \dots, s_n in $\bar{\mathcal{M}}$ and es_0, \dots, es_n in \mathcal{M} such that $\tau(es_i) = s_i$ for each $0 \leq i \leq n$ and $\sigma(s_0) = es_0$, we have $\bar{R}(s_0, \dots, s_{n-1}, a_{n-1}, s_n) = R(es_{n-1}, a_{n-1}, es_n)$.

We also introduce the notion of feasible policies:

Definition 10 (Feasible Policy). *Given a PONMRDP $\bar{\mathcal{M}}$ with an extended POMDP \mathcal{M} . We say that a policy π in \mathcal{M} is not feasible in $\bar{\mathcal{M}}$ if there exist a pair of states s in $\bar{\mathcal{M}}$ and es in \mathcal{M} , where \vec{s} is a trajectory in $\bar{\mathcal{M}}$ ending in s , $\tau(es) = s$ and $v^\pi(es) \neq v^\pi(\vec{s})$ for every policy $\bar{\pi}$ in $\bar{\mathcal{M}}$.*

Now, we can prove the following theorem.

Theorem 1. *There exist a PONMRDP $\bar{\mathcal{M}}$ with expansion \mathcal{M} given as in Def. 7 with item 5 replaced by item 5', where the optimal policy in \mathcal{M} is not feasible in $\bar{\mathcal{M}}$.*

Proof. Consider the PONMRDP $\bar{\mathcal{M}} = \langle S, A, T, \bar{R}, Z, O, \gamma \rangle$ depicted in Fig. 2, such that:

1. $S = \{s_1, s_2, s_3\}$;
2. $A = \{a_1, a_2\}$;

3. $T^{a_1} = \{[0, 0, 1], [0, 0, 1], [1, 0, 0]\}$ and $T^{a_2} = \{[0, 0, 1], [0, 0, 1], [0, 1, 0]\}$, where $T^{a_1}[1] = T^{a_1}(s_1)[s_1, s_2, s_3]$, $T^{a_1}[2] = T^{a_1}(s_2)[s_1, s_2, s_3], \dots$;
4. the reward function \bar{R} is given as

$$\bar{R}(\vec{w}) = \begin{cases} 1 & \text{if } \vec{w} = w_0, w_1, w_2 \text{ and } w_0 \neq w_2; \\ 0 & \text{otherwise.} \end{cases}$$

5. the observation function O is such that $O(s_1) = O(s_2) = z_1$ and $O(s_3) = z_2$ for $Z = \{z_1, z_2\}$ (we assume that observations are independent from the action taken);
6. $\gamma = 1$.

Finally, we assume an initial distribution $b_0 = [0.3, 0.7, 0]$. Note that, we consider time horizons of 3 time steps only.

Since both initial states s_1 and s_2 are indistinguishable (i.e., they return the same observation), but there is a higher chance of starting in s_2 , an optimal policy in $\bar{\mathcal{M}}$ is taking action a_1 for every state.

$$\pi^*(b_i) = [p(a_1) = 0, p(a_2) = 1] \text{ for any } b_i \quad (1)$$

Now consider the expansion \mathcal{M} , where states and observations are extended with a labelling that tells the agent the previous state visited. Formally, $\mathcal{M} = \langle ES, A, T_{ES}, R, EZ, O_{ES}, \gamma \rangle$, where:

- A and γ are the same as in $\bar{\mathcal{M}}$;
- $ES = \{s_1, s_2, s_1s_3, s_2s_3, s_3s_1, s_3s_2\}$;
- $EZ = \{ez_1, ez_2, ez_3, ez_4, ez_5\}$;
- $O_{ES}(\emptyset, \emptyset, s_1) = O_{ES}(\emptyset, \emptyset, s_2) = ez_1$ and for all $a \in A$, $es \in ES$, $O_{ES}(es, a, s_3s_1) = ez_2$, $O_{ES}(es, a, s_2s_1) = ez_3$, $O_{ES}(es, a, s_3s_1) = ez_4$, $O_{ES}(es, a, s_3s_2) = ez_5$.
- The transition function is as depicted in Fig ??.
- The reward function is defined as:

$$R(es_i, a_i, es_{i+1}) = \begin{cases} 1 & \text{if } es_i = s_1s_3, a_i = a_2, es_{i+1} = s_3s_2; \\ 1 & \text{if } es_i = s_2s_3, a_i = a_1, es_{i+1} = s_3s_1; \\ 0 & \text{otherwise.} \end{cases}$$

Now, functions $\tau : ES \rightarrow S$, $\sigma : S \rightarrow ES$, $\tau' : EZ \rightarrow Z$, and $\sigma' : Z \rightarrow EZ$ are intuitively defined as follows: τ and τ' add the information of the previous state visited to construct the extended state or observation respectively, while σ and σ' remove that information. The POMDP \mathcal{M} satisfies the requirements in Def. 7 with condition 5' to be an expansion of $\bar{\mathcal{M}}$. Figure 2 illustrates \mathcal{M} and $\bar{\mathcal{M}}$.

We now show that the optimal policies in this model is not feasible in the original. In the expansion POMDP \mathcal{M} an optimal policy is one that:

$$\pi_b^*(eb_i) = \begin{cases} a_2 & \text{if } ez_i = ez_2; \\ a_1 & \text{if } ez_i = ez_3; \\ a_1 \text{ or } a_2 & \text{otherwise.} \end{cases} \quad (2)$$

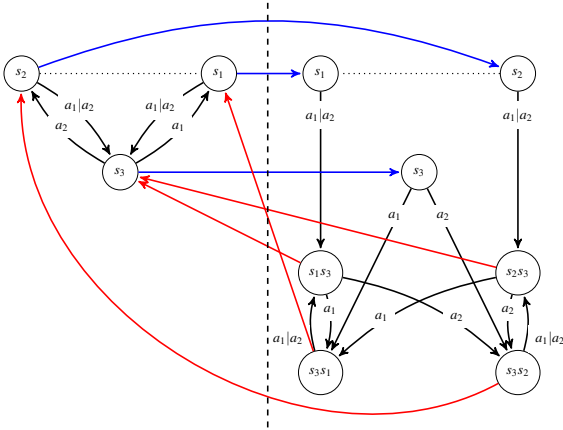


Figure 2: Counterexample to naive extension. The models for the PONMRDP (left side) and its extended POMDP (right side), satisfying item 5' instead of the proposed item 5 from Def. 7. Red lines denote some $\tau(es)$ transformations while blue lines illustrate the cases in which $\sigma(s)$ is only the current state.

where ez_i is the last observation used to obtain eb_i . So, we have that $v^{e\pi^*}(s_1s_3) = v^{e\pi^*}(s_1s_2) = 1$.

However, in the original model $\overline{\mathcal{M}}$ the maximum value we can obtain for $\tau(s_1s_3)$ is $v^{\pi^*}(s_3) = 0.7$. Thus we have that $v^{e\pi^*} \neq v^{\pi^*}$. \square

Intuitively, an expansion with condition 5' does not prevent generating an equivalent model where the dynamics are the same, but information differ. In models with knowledge of the state, the expansion method must ensure that the agent has access to the same information available in the trajectories of the original system.

5 Related Work

The intersection between formal methods and reinforcement learning has lead to a growing interest and demand of reinforcement learning agents solving temporally extended instructions that are naturally described as non-Markovian. Early work (Bacchus et al., 1996) focus on facilitating the application of RL algorithms to non-Markovian problems by introducing the concept of extended MDP as a minimal equivalent Markovian construction that allows RL agents to tackle a problem where rewards are naturally conceived as non-Markovian. Later literature (Toro Icarte et al., 2018; Giacomo et al., 2019; Illanes et al., 2020) has applied similar constructions with increasingly complex benchmarks, e.g., the Minecraft-inspired navigation environment (Andreas et al., 2017) or Min-

iGrid (Chevalier-Boisvert et al., 2018), and expressive languages such as co-safe linear-time temporal logic (co-safe LTL) (Kupferman and Vardi, 2001) or linear dynamic logic over finite traces (LDL_f) (Brafman et al., 2018). This combination of temporal logic and RL has also sparked interest in multi-agent systems in a comparable extending-the-non-Markovian-model fashion with examples such as the extended Markov games (León and Belardinelli, 2020) and the product Markov games (Hammond et al., 2021). Some of the latest contributions (Toro Icarte et al., 2019) have empirically applied this kind of extensions to partially observable environments. However, to the best of our knowledge, no previous work has provided theoretical studies about how a PONMRDP should be extended to ensure finding an optimal policy that is guaranteed to be optimal and applicable to the original non-Markovian problem at hand.

6 Conclusions

When tackling real world problems with autonomous agents it is common to face settings that are naturally described as non-Markovian (i.e. relying on the past) and partially observable. We presented an expansion method for extending p.o. non-Markovian reward decision processes, so as to obtain an equivalent POMDP, where a Markovian agent can find an optimal policy that is guaranteed to be optimal for the original non-Markovian model. We also provided proof that naive expansions from existing methods for fully observable models might find solutions that are not applicable in the original problem. Note that, in this work we considered the setting in which the observation function is deterministic to simplify the presentation. Nonetheless, a counterexample for the naive extension in this setting is a counterexample for the general case. Our work provides theoretical ground for research lines solving complex instructions, such as complex temporal logic formulas, through RL in environments with imperfect knowledge of the state of the system.

REFERENCES

- Abate, A., Gutierrez, J., Hammond, L., Harrenstein, P., Kwiatkowska, M., Najib, M., Perelli, G., Steeples, T., and Wooldridge, M. (2021). Rational verification: game-theoretic verification of multi-agent systems. *Applied Intelligence*, 51(9):6569–6584.

- Andreas, J., Klein, D., and Levine, S. (2017). Modular multitask reinforcement learning with policy sketches. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 166–175. JMLR. org.
- Bacchus, F., Boutilier, C., and Grove, A. (1996). Rewarding behaviors. In *Proceedings of the National Conference on Artificial Intelligence*, pages 1160–1167.
- Badia, A. P., Piot, B., Kapturowski, S., Sprechmann, P., Vitvitskiy, A., Guo, D., and Blundell, C. (2020). Agent57: Outperforming the atari human benchmark. *arXiv preprint arXiv:2003.13350*.
- Bellemare, M. G., Candido, S., Castro, P. S., Gong, J., Machado, M. C., Moitra, S., Ponda, S. S., and Wang, Z. (2020). Autonomous navigation of stratospheric balloons using reinforcement learning. *Nature*, 588(7836):77–82.
- Bellemare, M. G., Dabney, W., and Munos, R. (2017). A distributional perspective on reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 449–458. JMLR. org.
- Brafman, R., De Giacomo, G., and Patrizi, F. (2018). Ltlf/ldlf non-markovian rewards. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.
- Chevalier-Boisvert, M., Willems, L., and Pal, S. (2018). Minimalistic gridworld environment for openai gym. <https://github.com/maximecb/gym-minigrid>.
- Giacomo, G. D., Iocchi, L., Favorito, M., and Patrizi, F. (2019). Foundations for restraining bolts: Reinforcement learning with LTLf/LDLf restraining specifications. In Benton, J., Lipovetzky, N., Onaindia, E., Smith, D. E., and Srivastava, S., editors, *Proceedings of the Twenty-Ninth International Conference on Automated Planning and Scheduling, ICAPS 2018, Berkeley, CA, USA, July 11-15, 2019*, pages 128–136. AAAI Press.
- Hammond, L., Abate, A., Gutierrez, J., and Wooldridge, M. (2021). Multi-agent reinforcement learning with temporal logic specifications. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, pages 583–592.
- Hill, F., Lampinen, A., Schneider, R., Clark, S., Botvinick, M., McClelland, J. L., and Santoro, A. (2020). Environmental drivers of systematicity and generalization in a situated agent. In *International Conference on Learning Representations*.
- Hill, F., Tieleman, O., von Glehn, T., Wong, N., Merzic, H., and Clark, S. (2021). Grounded language learning fast and slow. In *International Conference on Learning Representations*.
- Icarte, R. T., Waldie, E., Klassen, T., Valenzano, R., Castro, M., and McIlraith, S. (2019). Learning reward machines for partially observable reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 15497–15508.
- Illanes, L., Yan, X., Icarte, R. T., and McIlraith, S. A. (2020). Symbolic plans as high-level instructions for reinforcement learning. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 30, pages 540–550.
- Kupferman, O. and Vardi, M. Y. (2001). *Formal Methods in System Design*, 19(3):291–314.
- Lake, B. M. (2019). Compositional generalization through meta sequence-to-sequence learning. In *Advances in Neural Information Processing Systems*, pages 9788–9798.
- León, B. G. and Belardinelli, F. (2020). Extended markov games to learn multiple tasks in multi-agent reinforcement learning. In Giacomo, G. D., Catalá, A., Dilkina, B., Milano, M., Barro, S., Bugarín, A., and Lang, J., editors, *ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020)*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 139–146. IOS Press.
- León, B. G., Shanahan, M., and Belardinelli, F. (2020). Systematic generalisation through task temporal logic and deep reinforcement learning. *CoRR*, abs/2006.08767.
- León, B. G., Shanahan, M., and Belardinelli, F. (2022). In a nutshell, the human asked for this: Latent goals for following temporal specifications. *International Conference on Learning Representations (to appear)*.
- Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540):529.
- Oh, J., Singh, S., Lee, H., and Kohli, P. (2017). Zero-shot task generalization with multi-task deep reinforcement learning. In *Proceedings of the 34th*

- International Conference on Machine Learning-Volume 70*, pages 2661–2670. JMLR. org.
- Rashid, T., Samvelyan, M., Schroeder, C., Farquhar, G., Foerster, J., and Whiteson, S. (2018). Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 4295–4304. PMLR.
- Samvelyan, M., Rashid, T., Schroeder de Witt, C., Farquhar, G., Nardelli, N., Rudner, T. G., Hung, C.-M., Torr, P. H., Foerster, J., and Whiteson, S. (2019). The starcraft multi-agent challenge. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, pages 2186–2188. International Foundation for Autonomous Agents and Multiagent Systems.
- Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., et al. (2017). Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*.
- Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.
- Toro Icarte, R., Klassen, T. Q., Valenzano, R., and McIlraith, S. A. (2018). Teaching multiple tasks to an RL agent using LTL. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 452–461.
- Toro Icarte, R., Waldie, E., Klassen, T., Valenzano, R., Castro, M., and McIlraith, S. (2019). Learning reward machines for partially observable reinforcement learning. volume 32, pages 15523–15534.
- Vinyals, O., Babuschkin, I., Chung, J., Mathieu, M., Jaderberg, M., Czarnecki, W. M., Dudzik, A., Huang, A., Georgiev, P., Powell, R., et al. (2019). Alphastar: Mastering the real-time strategy game starcraft ii. *DeepMind blog*, page 2.
- Zhao, M., Liu, Z., Luan, S., Zhang, S., Precup, D., and Bengio, Y. (2021). A consciousness-inspired planning agent for model-based reinforcement learning. In *Advances in Neural Information Processing Systems*.