

Constructing Non-Markovian Decision Process via History Aggregator

Yongyi Wang^{1,2}, Wenxin Li^{1,2}

¹AILab, School of Computer Science, Peking University, Beijing 100871, China.

²National Key Laboratory for Multimedia Information Processing, School of Computer Science, Peking University, Beijing 100871, China.
{wangyongyi, lwx}@pku.edu.cn

Abstract

In the domain of algorithmic decision-making, non-Markovian dynamics manifest as a significant impediment, especially for paradigms such as Reinforcement Learning (RL), thereby exerting far-reaching consequences on the advancement and effectiveness of the associated systems. Nevertheless, the existing benchmarks are deficient in comprehensively assessing the capacity of decision algorithms to handle non-Markovian dynamics. To address this deficiency, we have devised a generalized methodology grounded in category theory. Notably, we established the category of Markov Decision Processes (MDP) and the category of non-Markovian Decision Processes (NMDP), and proved the equivalence relationship between them. This theoretical foundation provides a novel perspective for understanding and addressing non-Markovian dynamics. We further introduced non-Markovianity into decision-making problem settings via the History Aggregator for State (HAS). With HAS, we can precisely control the state dependency structure of decision-making problems in the time series. Our analysis demonstrates the effectiveness of our method in representing a broad range of non-Markovian dynamics. This approach facilitates a more rigorous and flexible evaluation of decision algorithms by testing them in problem settings where non-Markovian dynamics are explicitly constructed.

environments are prevalent in real-world applications, including human physiology, biological systems, and material science [Gupta *et al.*, 2021].

For the characterization and resolution of non-Markovian decision problems, in addition to NMDP, several theoretical models have emerged, including linear-dynamic-logic-based Regular Decision Process (RDP) [Brafman *et al.*, 2019; Abadi and Brafman, 2020], automaton-based Reward Machine (RM) [Camacho *et al.*, 2019; Rens and Raskin, 2020], and Markov Abstraction [Hutter, 2009; Maillard *et al.*, 2011; Veness *et al.*, 2011; Nguyen *et al.*, 2013; Lattimore *et al.*, 2013; Majeed *et al.*, 2018; Ronca *et al.*, 2022], among others. These models are utilized either to model the state transition dynamics or the reward mechanisms of non-Markovianity, but the way they model non-Markovianity varies, thereby causing inconvenience to the research of decision algorithms. In response, we establish the MDP category and the NMDP category, providing a unified and rigorous view within the framework of category theory. Under this framework, any general approach that converts NMDPs into MDPs while preserving the transition dynamics can be seen as a functor from the NMDP category to the MDP category. This framework not only encapsulates the essence of preceding research but also paves the way for devising novel strategies to tackle non-Markovian decision problems. Moreover, the utilization of the category theory perspective facilitates a deeper understanding of the relationship between MDP and NMDP. We prove that the MDP category and the NMDP category are equivalent categories based on the intuition that any NMDP can be transformed into an MDP, given that the entire history can be regarded as a state.

1 Introduction

Decision-making algorithms, such as RL, generally demand that the problem satisfy the Markov assumption, which posits that the current observation encapsulates all accessible information for decision-making, rendering past observations irrelevant. However, many real-world systems do not adhere to this assumption. For example, control systems often exhibit non-Markovianity due to limited sensor information about internal states [Whitehead and Lin, 1995]. Similarly, open quantum systems display non-Markovian characteristics due to complex interactions with their external environment [de Vega and Alonso, 2017]. Such non-Markovian

Based on the theoretical models mentioned above, numerous algorithms have emerged to solve non-Markovian decision problems. However, an algorithm that is both commonly accepted and highly effective for NMDPs remains elusive. The evaluation of such algorithms relies on non-Markovian decision problems, which poses a set of challenges. For instance, Gaon and Brafman [2020] utilizes a multi-armed bandit and robot grid world with non-Markovian rewards; Gupta *et al.* [2021] employs a blood glucose control simulation environment; Ronca *et al.* [2022] uses Rotating MAB, Malfunction MAB, Enemy Corridor, Reset - Rotating MAB, and Flickering grid; Dohmen *et al.* [2022] uses an office grid-world scenario; Qin *et al.* [2024] uses modified Mujoco envi-

ronments; Chandak *et al.* [2024] uses modified Cartpole and mountain Car, along with a controlled non-Markovian random walk. The non-Markovian environments described in the literature, which are scattered and unsystematic, originate from diverse fields. Moreover, most of these environments are either obtained through simple construction or derived from masking some part of the observation in existing MDP, that is, by transforming it into a Partially Observable MDP (POMDP), which may change the internal difficulty because of information loss in the observation. Their simplicity and lack of standardization impede the development of a universal benchmark, thereby limiting the ability to comprehensively evaluate how effectively algorithms can manage non-Markovian dynamics.

To address the issue, we have developed a method for constructing NMDPs from MDPs. We use the History Aggregator for State (HAS) and History Aggregator for Reward (HAR) to introduce non-Markovianity into MDPs’ state transition dynamics and reward mechanisms. Given that transition dynamics is more essential in non-Markovianity, this paper focuses on HAS. When HAS meets the reversibility condition, the original MDP’s essence is preserved during transformation. Specifically, this condition enables the original state to be decoded from the NMDP’s complete history without information loss. Using reversible HAS allows the constructed NMDP to more precisely evaluate decision-making algorithms’ ability to remember and decode history to obtain the original state. Since handling complex temporal dependencies through memorization and decoding is crucial for addressing non-Markovianity, our method, using different HAS, constructs diverse temporal dependency structures, providing greater flexibility and applicability.

We propose two special kinds of easily-implementable reversible HAS. One aggregates history by applying a binary group operator between states, and the other by using auxiliary sequences and ring operators. The idea behind the first one is that directly adding vector states raises the order of Markov dependence by one. For the second, its intuition is derived from the convolution operation in signal processing. We use an auxiliary sequence to assign weights to states and then aggregate them. In actual problem scenarios, states are generally representable as real vectors. Thus, these two types of HAS can be straightforwardly implemented using linear algebra operators. Moreover, the non-Markovian environments constructed by these two methods share the same interface as the original Markovian environments. This characteristic enables the direct application of the same decision-making algorithm to solve problems in both types of environments without modification. Consequently, it becomes possible to conduct a pure assessment of the performance degradation of decision-making algorithms resulting from the introduction of non-Markovianity. There is no need to consider changes in the inherent complexity of the decision-making problem itself. This is because the construction based on the HAS only imposes requirements on the decision-making algorithm’s capacity for history memorization and decoding.

Building on the challenges associated with constructing non-Markovian decision problems as discussed above, the subsequent chapters of our work are structured to provide so-

lutions. In Chapter 2, we introduce the fundamentals of category theory, MDP, and NMDP, along with the basics of algebra. This serves as essential background for understanding our work. In Chapter 3, we formally describe and prove the equivalence of the MDP and NMDP categories. This not only unifies the two concepts but also naturally leads to the method of constructing NMDPs in the subsequent chapter. Chapter 4 presents the definition of HAS, two construction methods using reversible HAS, and an analysis of their properties. These methods innovatively introduce non-Markovianity while preserving the original nature of the problem. Although the main contribution of our work is theoretical, we convert some classical MDPs into NMDPs and conduct several empirical experiments using some RL algorithms in Chapter 5. This validates our theoretical framework and demonstrates its applicability. The Appendix provides mathematical proofs, illustrative examples, intuitive explanations of key concepts, and links to the code. This allows the main text to focus on the key ideas without being cluttered with details. Overall, our work offers a comprehensive, novel, and practical solution for constructing non-Markovian decision problems.

2 Preliminaries

2.1 Notations

In the following, we denote by $\Delta_X := \{p \mid p : X \rightarrow [0, \infty), \sum_{x \in X} p(x) = 1\}$ the set of all probability distributions on the set X , denote by $f^{(n)}$ an n -variable function created from the unary function f in the manner of $f^{(n)}(x_1, x_2, \dots, x_n) := (f(x_1), f(x_2), \dots, f(x_n))$, and denote by $x_{m:n}$ ($m, n \in \mathbb{N}, m \leq n$) a finite sequence $\{x_i\}_{i=m}^n$. We do not distinguish between $x_{n:n}$ and x_n in the following.

2.2 Category Theory

A category is a system of related objects. The objects do not live in isolation: there is some notion of morphism, or equivalently, map, between objects, binding them together [Leinster, 2014]. Category theory delves into the relationships between objects and morphisms. Objects can be of any nature, and morphisms denote the relations or transformations linking these objects. For example, all groups together with all group homomorphisms form a category of groups, and any partially ordered set can also form a category. This theory offers a potent framework for apprehending mathematical structures. By distilling common traits, category theory allows for a unified analysis across diverse mathematical domains. It uncovers profound connections and streamlines intricate concepts, presenting a more coherent view of mathematics.

Definition 1 (Category). A category \mathcal{C} consists of:

- a collection $ob(\mathcal{C})$ of **objects**;
- for each $A, B \in ob(\mathcal{C})$, a collection $\mathcal{C}(A, B)$ of **morphisms** from A to B ;
- for each $A, B, C \in ob(\mathcal{C})$, a function

$$\begin{aligned} \mathcal{C}(A, B) \times \mathcal{C}(B, C) &\rightarrow \mathcal{C}(A, C) \\ (g, f) &\mapsto g \circ f \end{aligned}$$

called **composition**;

- for each $\mathcal{A} \in \text{ob}(\mathfrak{C})$, an element $1_{\mathcal{A}} \in \mathfrak{C}(\mathcal{A}, \mathcal{A})$, called the **identity** on \mathcal{A} ,

satisfying the following axioms:

- **associativity**: for each $f \in \mathfrak{C}(\mathcal{A}, \mathcal{B})$, $g \in \mathfrak{C}(\mathcal{B}, \mathcal{C})$, $h \in \mathfrak{C}(\mathcal{C}, \mathcal{D})$ we have $(h \circ g) \circ f = h \circ (g \circ f)$;
- **identity**: for each $f \in \mathfrak{C}(\mathcal{A}, \mathcal{B})$, we have $f = f \circ 1_{\mathcal{A}} = 1_{\mathcal{B}} \circ f$.

Just as morphisms convey intra-category object relations or transformations, functors represent inter-category ones.

Definition 2 (Functor). Let $\mathfrak{A}, \mathfrak{B}$ be categories. A **functor** $\mathbf{F} : \mathfrak{A} \rightarrow \mathfrak{B}$ consists of:

- a function $\text{ob}(\mathfrak{A}) \rightarrow \text{ob}(\mathfrak{B})$ written as $\mathcal{A} \mapsto \mathbf{F}(\mathcal{A})$;
- for each $\mathcal{A}, \mathcal{A}' \in \mathfrak{A}$, a function

$$\mathfrak{A}(\mathcal{A}, \mathcal{A}') \rightarrow \mathfrak{B}(\mathbf{F}(\mathcal{A}), \mathbf{F}(\mathcal{A}'))$$

written as $f \mapsto \mathbf{F}(f)$;

satisfying the following axioms:

- $\mathbf{F}(f' \circ f) = \mathbf{F}(f') \circ \mathbf{F}(f)$ if $\mathcal{A} \xrightarrow{f} \mathcal{A}' \xrightarrow{f'} \mathcal{A}''$ in \mathfrak{A} ;
- $\mathbf{F}(1_{\mathcal{A}}) = 1_{\mathbf{F}(\mathcal{A})}$ if $\mathcal{A} \in \mathfrak{A}$.

The collection of functors from \mathfrak{A} to \mathfrak{B} also forms a category $\mathbf{Fun}(\mathfrak{A}, \mathfrak{B})$.

2.3 Algebra

Definition 3 (Semigroup). Let S be a non-empty set, $\cdot : S \times S \rightarrow S$ is a binary operator, if the following axioms hold:

- **closure**: $\forall a, b \in S, a \cdot b \in S$;
- **associativity**: $\forall a, b, c \in S, a \cdot (b \cdot c) = (a \cdot b) \cdot c$.

Then (S, \cdot) forms a semigroup.

If there exists a **unit** $e \in S$ in a semigroup (S, \cdot) such that $\forall a \in S, a \cdot e = e \cdot a = a$, then (S, \cdot) becomes a **monoid**.

Definition 4 (Group). Let (G, \cdot) be a monoid, e is the unit, if the following axiom holds:

- **inverse**: $\forall a \in G, \exists a^{-1} \in G, a^{-1} \cdot a = a \cdot a^{-1} = e$.

Then (G, \cdot) forms a group.

If the operator in a group (G, \cdot) is commutative, i.e. $\forall a, b \in G, a \cdot b = b \cdot a$, then (G, \cdot) is an **abelian group**.

Definition 5 (Ring). Let R be a non-empty set, if (R, \cdot) forms a semigroup, $(R, +)$ forms an abelian group with unit $0 \in R$, and the following axioms hold:

- **left distributivity**: $\forall a, b, c \in R, a \cdot (b + c) = a \cdot b + a \cdot c$;
- **right distributivity**: $\forall a, b, c \in R, (b + c) \cdot a = b \cdot a + c \cdot a$.

Then $(R, +, \cdot)$ forms a ring.

If the operator \cdot in ring $(R, +, \cdot)$ is commutative, then $(R, +, \cdot)$ becomes a **commutative ring**, and if (R, \cdot) is a monoid with unit 1, then $(R, +, \cdot)$ becomes a **unital ring** with unit 1.

Definition 6 (Left R -Module). Let $(R, +, \cdot)$ be a ring, (M, \oplus) be an abelian group, if there exists a map $R \times M \rightarrow M$ under which the image of any $(r, m) \in R \times M$ is denoted as $rm \in M$, and the following axioms hold:

- **R -distributivity**:

$$\forall r \in R, \forall m, m' \in M, r(m \oplus m') = rm \oplus rm'$$

- **M -distributivity**:

$$\forall r, r' \in R, \forall m \in M, (r + r')m = rm \oplus r'm$$

- **associativity**:

$$\forall r, r' \in R, \forall m \in M, r(r'm) = (r \cdot r')m$$

- **identity**: if $(R, +, \cdot)$ is a unital ring with unit 1, then $\forall m \in M, 1m = m$.

Then M is called a left R -module.

A left R -module is a generalization of a linear space. When R becomes a field, left R -module M becomes a linear space.

2.4 MDP and NMDP

Before setting up the MDP and NMDP categories, we pin down their definitions.

Definition 7 (Markov Decision Process (MDP)). An MDP \mathcal{M} is a tuple $\langle \rho_0, S, A, \{T_t\}_{t=0}^{\infty} \rangle$ consisting of:

- Initial state distribution $\rho_0 \in \Delta_S$;
- State set S ;
- Action set A ;
- Transition dynamics at time t :

$$T_t : S \times A \rightarrow \Delta_{S \times \mathbb{R}}$$

A **policy** for an MDP at time t is defined as $\pi_t : S \rightarrow \Delta_A$. An MDP is called **degenerate** if $\exists s, s' \in S, \forall t \in \mathbb{N}, \forall a \in A, T_t(s, a) = T_t(s', a)$, which means there are indistinguishable states in S .

Note that, by definition, an MDP is not necessarily time-homogeneous. However, any non-time-homogeneous MDP can be converted into a time-homogeneous MDP by extending the state set S to be a subset of $\mathbb{N} \times S$, where each state is associated with its timestep, i.e., $s_t \mapsto (t, s_t)$ and $T : ((t, s_t), a) \mapsto T_t(s_t, a)$.

Definition 8 (Non-Markov Decision Process (NMDP)). An NMDP \mathcal{N} is a tuple $\langle \rho_0, S, A, \{T_t\}_{t=0}^{\infty} \rangle$ consisting of:

- Initial state distribution $\rho_0 \in \Delta_S$;
- State set S ;
- Action set A ;
- Transition dynamics at time t :

$$T_t : S^{t+1} \times A^t \times \mathbb{R}^t \times A \rightarrow \Delta_{S \times \mathbb{R}}$$

A **policy** for an NMDP at time t is $\pi_t : S^{t+1} \times A^t \times \mathbb{R}^t \rightarrow \Delta_A$. The **history set at time t** for an MDP or an NMDP is $H_t := S^{t+1} \times A^t \times \mathbb{R}^t$, with which transition dynamics and policy of an NMDP can be rewritten as $T_t : H_t \times A \rightarrow \Delta_{S \times \mathbb{R}}, \pi_t : H_t \rightarrow \Delta_A$. The **history set H** for an MDP or an NMDP is defined as the disjoint union of all H_t , i.e., $H := \bigsqcup_{t=0}^{\infty} H_t$. The **proper prefix** relation between h and h' is denoted as $h \prec h'$, indicating that h is a proper prefix of h' , i.e. $h = (s_{0:t}, a_{0:t-1}, r_{0:t-1})$ and $h' = (s_{0:t+k}, a_{0:t+k-1}, r_{0:t+k-1}), k \in \mathbb{N}^+$.

With the concept of the history set for both MDPs and NMDPs, we can define operators that extract states, actions, and rewards from a history.

Definition 9 (States, Actions, and Rewards Extraction Operators). The states, actions, and rewards extraction operators are:

- $\mathcal{E}_S : h_t \mapsto s_{0:t} \ (\forall t \in \mathbb{N}, \forall h_t \in H_t);$
- $\mathcal{E}_A : h_t \mapsto a_{0:t-1} \ (\forall t \in \mathbb{N}^+, \forall h_t \in H_t);$
- $\mathcal{E}_R : h_t \mapsto r_{0:t-1} \ (\forall t \in \mathbb{N}^+, \forall h_t \in H_t).$

Definition 10 (Latest State, Action, and Reward Extraction Operators). The latest state, action, and reward extraction operators at time t are:

- $\mathcal{L}_{S,t} : h_t \mapsto s_t \ (\forall t \in \mathbb{N}, \forall h_t \in H_t);$
- $\mathcal{L}_{A,t} : h_{t+1} \mapsto a_t \ (\forall t \in \mathbb{N}, \forall h_{t+1} \in H_{t+1});$
- $\mathcal{L}_{R,t} : h_{t+1} \mapsto r_t \ (\forall t \in \mathbb{N}, \forall h_{t+1} \in H_{t+1}).$

3 Equivalence of MDP and NMDP Category

In this chapter, we present the definitions of the MDP category and the NMDP category, together with the theorem regarding their equivalence relationship. The relevant proofs are provided in the Appendix.

Definition 11 (MDP Category). \mathfrak{M} is a category where:

- **Objects:** The objects are MDPs in the form of $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$;
- **Morphisms:** A morphism $\phi = (\phi_S, \phi_A, \phi_R)$ from $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$ to $\mathcal{M}' = \langle \rho'_0, S', A', \{T'_t\}_{t=0}^\infty \rangle$ where $\phi_S : S \rightarrow S'$, $\phi_A : A \rightarrow A'$, $\phi_R : \mathbb{R} \rightarrow \mathbb{R}$, satisfies the following properties:
 - $\rho_0 = \rho'_0 \circ \phi_S$;
 - $T_t(s, a) = ((T'_t \circ (\phi_S, \phi_A))(s, a)) \circ (\phi_S, \phi_R)$, $(\forall t \in \mathbb{N}, \forall s \in S, \forall a \in A)$.

Identity: $1_{\mathcal{M}} = (1_S, 1_A, 1_R)$;

Composition: composition of $(\phi'_S, \phi'_A, \phi'_R)$ and (ϕ_S, ϕ_A, ϕ_R) is $(\phi'_S \circ \phi_S, \phi'_A \circ \phi_A, \phi'_R \circ \phi_R)$.

Definition 12 (NMDP Category). \mathfrak{N} is a category where:

- **Objects:** The objects are NMDPs in the form of $\mathcal{N} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$;
- **Morphisms:** A morphism $\psi = (\psi_S, \psi_A, \psi_R)$ from $\mathcal{N} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$ to $\mathcal{N}' = \langle \rho'_0, S', A', \{T'_t\}_{t=0}^\infty \rangle$ where $\psi_S : S \rightarrow S'$, $\psi_A : A \rightarrow A'$, $\psi_R : \mathbb{R} \rightarrow \mathbb{R}$, satisfies the following properties:
 - $\rho_0 = \rho'_0 \circ \psi_S$;
 - $T_t(h_t, a) = ((T'_t \circ (\psi_{H_t}, \psi_A))(h_t, a)) \circ (\psi_S, \psi_R)$, $(\forall t \in \mathbb{N}^+, \forall h_t \in H_t, \forall a \in A)$,
where $\psi_{H_t} = (\psi_S^{(t+1)}, \psi_A^{(t)}, \psi_R^{(t)})$.

Identity: $1_{\mathcal{N}} = (1_S, 1_A, 1_R)$;

Composition: composition of $(\psi'_S, \psi'_A, \psi'_R)$ and (ψ_S, ψ_A, ψ_R) is $(\psi'_S \circ \psi_S, \psi'_A \circ \psi_A, \psi'_R \circ \psi_R)$.

A morphism in either category represents that the transition dynamics of the source decision process can be "simulated" by the target.

As discussed in Chapter 1, previous algorithmic works have sought functors in $\mathbf{Fun}(\mathfrak{N}, \mathfrak{M})$, which can generally be summarized as approximations of the Markov abstraction

functor $\mathbf{M} \in \mathbf{Fun}(\mathfrak{N}, \mathfrak{M})$, utilizing various techniques to compress history. In contrast to what is described in the existing literature, our definition of Markov abstraction retains the entire history without any compression.

Definition 13 (Markov Abstraction). Markov abstraction is a functor $\mathbf{M} : \mathfrak{N} \rightarrow \mathfrak{M}$. For any $\mathcal{N} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle \in \mathfrak{N}$ there exists $\mathcal{M}' = \mathbf{M}(\mathcal{N}) = \langle \rho'_0, S', A', \{T'_t\}_{t=0}^\infty \rangle \in \mathfrak{M}$, $\rho'_0 = \rho_0$, $A' = A$, which satisfies the following properties:

- $S' = H$, where H is the history set of \mathcal{N} ;
- $T'_t : \begin{matrix} S' \times A' & \rightarrow & \Delta_{S' \times \mathbb{R}} \\ (h_t, a) & \mapsto & (T_t(h_t, a)) \circ (\mathcal{L}_{S,t+1}, 1_{\mathbb{R}}) \end{matrix}$.

Note that an MDP is a special case of an NMDP, which naturally induces the non-Markov embedding functor denoted as $\mathbf{N} \in \mathbf{Fun}(\mathfrak{M}, \mathfrak{N})$.

Definition 14 (Non-Markov Embedding). Non-Markov embedding is a functor $\mathbf{N} : \mathfrak{M} \rightarrow \mathfrak{N}$. For any $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle \in \mathfrak{M}$, there exists $\mathcal{N}' = \mathbf{N}(\mathcal{M}) = \langle \rho'_0, S', A', \{T'_t\}_{t=0}^\infty \rangle \in \mathfrak{N}$, $\rho'_0 = \rho_0$, $S' = S$, $A' = A$, which satisfies the following property:

- $T'_t : \begin{matrix} H'_t \times A' & \rightarrow & \Delta_{S' \times \mathbb{R}} \\ (h'_t, a) & \mapsto & (T_t \circ (\mathcal{L}_{S,t}, 1_A))(h'_t, a) \end{matrix}$.

Although \mathfrak{M} and \mathfrak{N} have distinct properties by definition, we discover that they are equivalent categories. This leads to the following theorem.

Theorem 1 (Equivalence of \mathfrak{M} and \mathfrak{N}). Category \mathfrak{M} and \mathfrak{N} are equivalent through functor \mathbf{M} and \mathbf{N} , i.e. $\mathbf{M} \circ \mathbf{N} \cong 1_{\mathfrak{M}}$, $\mathbf{N} \circ \mathbf{M} \cong 1_{\mathfrak{N}}$.

Inspired by the equivalence of \mathfrak{M} and \mathfrak{N} , we believe that by exploring functors in $\mathbf{Fun}(\mathfrak{M}, \mathfrak{N})$, our work will also contribute to the study of algorithms that address non-Markovianity, i.e., identifying more computationally efficient functors in $\mathbf{Fun}(\mathfrak{N}, \mathfrak{M})$.

4 NMDP Constructed from MDP via HAS

Unlike many algorithmic studies that focus on developing functors in $\mathbf{Fun}(\mathfrak{N}, \mathfrak{M})$ to handle NMDP's history more effectively, our work focuses on constructing functors in $\mathbf{Fun}(\mathfrak{M}, \mathfrak{N})$ for transforming MDPs into NMDPs. While \mathbf{N} is such a functor, it is trivial since, in the constructed NMDP, the states prior to the current time are essentially irrelevant to decision-making. However, our interest lies in functors that can substantially incorporate non-Markovian transition dynamics into MDPs. To this end, we first introduce the History Aggregator for State (HAS) which enables us to construct such functors. Similarly, we can also define the History Aggregator for Reward (HAR), which substantially incorporates non-Markovian reward mechanisms into MDPs. As HAR is not the main content of this paper, we discuss it in the Appendix. Examples and intuitive explanations of the concepts presented in this chapter are also provided in the Appendix.

Definition 15 (History Aggregator for State (HAS)). For any MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$, an HAS $\mathcal{A}_S := \{\mathcal{A}_{S,t}\}_{t=0}^\infty$ is a series of maps in which $\mathcal{A}_{S,t} : H_t \rightarrow \mathcal{A}(H)$, $(\forall t \in \mathbb{N})$ where $\mathcal{A}(H)$ represents a set called the target state set of \mathcal{A}_S .

As the name implies, the HAS constructs the NMDP state by aggregating the MDP history. Before detailing our method, we first require that the aggregators be reversible.

Definition 16 (Reversibility of HAS on MDP). An HAS $\mathcal{A}_S = \{\mathcal{A}_{S,t}\}_{t=0}^\infty$ on MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$ is reversible iff there exists a series of maps, denoted as $\mathcal{A}_S^* := \{\mathcal{A}_{S,t}^*\}_{t=0}^\infty$, which satisfy:

$$\mathcal{A}_{S,t}^*(\{\mathcal{A}_{S,\tau}(h_\tau)\}_{\tau=0}^t) = s_t, \quad (\forall t \in \mathbb{N})$$

where $h_\tau \in H_\tau$, $h_0 \prec h_1 \prec \dots \prec h_t$, $s_t = \mathcal{L}_{S,t}(h_t)$.

The reversibility condition of HAS ensures that a decision-making algorithm can reconstruct the original MDP state from the history of the NMDP. A reversible HAS-based transformation neither alters the transition dynamics of the original MDP nor increases its inherent complexity. Consequently, the performance upper bound for the NMDP is identical to that of the original MDP. Thus, once we know the best achievable performance of the original MDP, we can set the same expectations for the constructed NMDP. By ensuring that the transformation from the MDP to the NMDP does not increase the inherent difficulty of the original but only requires effective management of the NMDP's history, we can more accurately evaluate a decision-making algorithm's ability to handle non-Markovianity through encoding and memorization of NMDP's history.

Noticing that the latest state extraction operators constitute reversible HAS, we have the following corollary.

Corollary 1. The series of latest state extraction operators, $\mathcal{L}_S := \{\mathcal{L}_{S,t}\}_{t=0}^\infty$, constitutes a reversible HAS on MDP with a target state set identical to the state set.

When any reversible HAS is applied to an MDP, it generates an NMDP. The NMDP's state set encompasses all aggregated histories of the original MDP. Its transition dynamics take the NMDP's history as input, decode it into an MDP state, and then apply the MDP's transition dynamics.

Definition 17 (Application of Reversible HAS to MDP). The application of a reversible HAS $\mathcal{A}_S = \{\mathcal{A}_{S,t}\}_{t=0}^\infty$ to an MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$ is an NMDP $\mathcal{N} = \langle \rho'_0, S', A', \{T'_t\}_{t=0}^\infty \rangle$, $A' = A$ in which $\rho'_0 = \rho_0 \circ \mathcal{A}_{S,0}^*$, $S' = \mathcal{A}(H)$, where H is the history set of \mathcal{M} , H' is the history set of \mathcal{N} ;

$$T'_t : H'_t \times A' \rightarrow \Delta_{S' \times \mathbb{R}} \\ (h'_t, a) \mapsto ((T_t \circ (\mathcal{A}_{S,t}^* \circ \mathcal{E}_{S'}(1_A)))(h'_t, a)) \circ (G_{S,h'_t}, 1_{\mathbb{R}})$$

$$\text{where } G_{S,h'_t} : S' \rightarrow S \\ s'_{t+1} \mapsto \mathcal{A}_{S,t+1}^*(\mathcal{E}_{S'}(h'_t), s'_{t+1})$$

The decoding process, denoted as $\mathcal{A}_{S,t}^*$ is the source of non-Markovianity. This is because, in the general case, decision-making algorithms have to remember the NMDP's entire history to decode the MDP's current state.

Although reversible HAS naturally induces functors in $\text{Fun}(\mathfrak{M}, \mathfrak{N})$, the definition of reversible HAS alone does not directly produce implementations, necessitating further specification.

For simplicity and practicality in implementation, we restrict our construction to using states alone, not the entire history. Given this constraint, we examine two types of HAS

constructions: one using just states, and the other relying on auxiliary sequences.

4.1 HAS Constructed Using Just States

One straightforward way to construct HAS using just states is introducing a binary operator on the state set.

Definition 18 (HAS Induced by Binary Operator). The HAS $\mathcal{B}_S := \{\mathcal{B}_{S,t}\}_{t=0}^\infty$ on MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$ induced by a binary operator $\otimes : S \times S \rightarrow S$ is:

$$\mathcal{B}_{S,t} : \begin{array}{ll} H_t & \rightarrow S \\ h_t & \mapsto \bigotimes_{\tau=0}^t s_\tau \end{array}$$

where $\{s_\tau\}_{\tau=0}^t = \mathcal{E}_S(h_t)$, $\bigotimes_{\tau=0}^0 s_\tau := s_0$, and $\bigotimes_{\tau=0}^{k+1} s_\tau := (\bigotimes_{\tau=0}^k s_\tau) \otimes s_{k+1}$, $(\forall k \in [0, t) \cap \mathbb{N})$.

According to the above definition, for the HAS induced by operator \otimes to be reversible, it is sufficient for \otimes to be a group operator, meaning (S, \otimes) forms a group. This approach is quite general since any state set can be extended to form a group, like a free group. Moreover, in practical problems, states are typically represented as vectors in \mathbb{R}^n , and \mathbb{R}^n forms a group under vector addition.

Theorem 2 (Reversibility of HAS Induced by Group Operator). The HAS \mathcal{G}_S on MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$ induced by binary operator \otimes is reversible if (S, \otimes) is a group.

Let $\mathbf{G} \in \text{Fun}(\mathfrak{M}, \mathfrak{N})$ denote the functor induced by \mathcal{G}_S . The proof of Theorem 2 in the Appendix shows that \mathbf{G} generates an NMDP. Unlike non-Markov embedding, the HAS \mathcal{G}_S does introduce non-Markovianity. It transforms an MDP, where decisions rely only on the current state, into an NMDP that requires knowledge of both the current and previous one-step state for decision-making. However, this state dependency structure is inadequate. Later, we'll show that repeated application of this HAS can yield a more complex state dependency structure.

Definition 19 (State Dependency Structure of NMDP History). The state dependency structure D_{h_t} of a history $h_t = (s_{0:t}, a_{0:t-1}, r_{0:t-1})$ of NMDP $\mathcal{N} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$ is a subset of \mathbb{N}

$$D_{h_t} := \{i \mid \exists s \in S, \exists a \in A, T_t(\sigma_i(h_t, s), a) \neq T_t(h_t, a)\}$$

where $\sigma_i(h_t, s) = ((s_{0:i-1}, s, s_{i+1:t}), a_{0:t-1}, r_{0:t-1})$.

According to the above definition, $i \in D_{h_t}$ implies that s_i has an impact on the state transition. Replacing it leads to a change in the subsequent transition probabilities. The concept of state dependency structure can also be extended to MDPs by defining it in terms of the state dependency structure of the MDP's non-Markov embedding. It has been observed that applying \mathbf{G} once to an MDP can increase the number of aggregated histories required to reconstruct the original MDP state in the resulting NMDP, thereby expanding the cardinality of the state dependency structure. Consequently, we explore applying \mathbf{G} multiple times to the MDP to generate NMDPs with more complex temporal dependencies. To facilitate the application of reversible HAS to NMDPs, and without significant loss of generality, we assume that the transition dynamics of the NMDP depend solely on the state component of the history. The extension required to apply the

reversible HAS to the NMDP is presented in the Appendix. With a slight abuse of notation, we let $\mathbf{G} \in \mathbf{Fun}(\mathfrak{M}, \mathfrak{N})$ also denote the functor induced by HAS.

Theorem 3 (Impact of \mathbf{G} on State Dependency Structure). *For any MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$, if it is non-degenerate, then the functor \mathbf{G} induced by some group operator on S ensures that the dependency structure $D_{h_t^n}$ of any history $h_t^n \in H_t^n$ of $\mathbf{G}^n(\mathcal{M})$ satisfies $D_{h_t^n} = [t - n, t] \cap \mathbb{N}$.*

The significance of Theorem 3 lies in its provision of a general method for designing a transition mechanism for an NMDP that relies on the states of the previous $n + 1$ steps. Specifically, it is sufficient to apply \mathbf{G} functor n times to a non-degenerate MDP.

4.2 HAS Constructed Using Auxiliary Sequences

Another approach to aggregating states is using auxiliary sequences to assign importance weights to these states. In this way, a binary operator is required to connect an auxiliary sequence with the sequence of states.

Definition 20 (HAS Induced by Auxiliary Sequences and Binary Operator). *The HAS $\mathcal{W}_S := \{\mathcal{W}_{S,t}\}_{t=0}^\infty$ on MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$ induced by a series of auxiliary sequences $W := \{W_t\}_{t=0}^\infty := \{\{w_\tau^t\}_{\tau=0}^t\}_{t=0}^\infty$ and a binary operator $*$: $W \times \mathcal{E}_S(H) \rightarrow S$ is:*

$$\mathcal{W}_{S,t} : \begin{array}{ll} H_t & \rightarrow S \\ h_t & \mapsto \{w_\tau^t\}_{\tau=0}^t * \{s_\tau\}_{\tau=0}^t \end{array}$$

where $\{s_\tau\}_{\tau=0}^t = \mathcal{E}_S(h_t)$.

The binary operator $*$ does not specify how the auxiliary sequence elements interact with the state sequence elements. Therefore, we define two operators to determine this interaction, one operator \cdot : $\bigsqcup_{t=0}^\infty W_t \times S \rightarrow S$ to apply an item in the auxiliary sequence to a state, the other operator \oplus : $S \times S \rightarrow S$ to aggregate two states. Without too much loss of generality, we restrict our discussion to left R-modules, where (S, \oplus) forms an Abelian group, \cdot is a scalar multiplication operator, and $\bigsqcup_{t=0}^\infty W_t$ becomes a unital ring with operators defined. We use \mathcal{R}_S to denote such an HAS.

Additionally, the correspondence between the elements of the auxiliary sequence and the elements of the state sequence needs to be specified. For the sake of simplicity in the subsequent discussion, we consider only two types of correspondence, where the auxiliary sequences are prefixes of a given sequence $\{w_\tau\}_{\tau=0}^\infty$:

$$\begin{array}{l} \vec{*} : (\{w_\tau\}_{\tau=0}^t, \{s_\tau\}_{\tau=0}^t) \mapsto \bigoplus_{\tau=0}^t (w_\tau \cdot s_\tau) \\ \overleftarrow{*} : (\{w_\tau\}_{\tau=0}^t, \{s_\tau\}_{\tau=0}^t) \mapsto \bigoplus_{\tau=0}^t (w_\tau \cdot s_{t-\tau}) \end{array}$$

where $\vec{*}$ is referred to as correlation operator, and $\overleftarrow{*}$ is referred to as convolution operator.

We examine the reversibility of HAS in these two cases separately. For the correlation operator case, we have the following theorem:

Theorem 4 (Reversibility of HAS induced by Auxiliary Sequence and Correlation Operator). *The HAS $\mathcal{R}_S := \{\mathcal{R}_{S,t}\}_{t=0}^\infty$ induced by prefixes of auxiliary sequence $\{w_t\}_{t=0}^\infty$ and operator $\vec{*}$ is reversible if each w_t is invertible in the ring.*

Based on the proof in the Appendix, it is obvious that the correlation operator only results in each set of the dependency structure of the constructed NMDP having a potential of no more than 2, which is similar to the effect of a group operator. Therefore, we focus on discussing the properties of the convolution operator.

For the convolution operator case, we have the following theorem:

Theorem 5 (Reversibility of HAS induced by Auxiliary Sequence and Convolution Operator). *The HAS $\mathcal{R}_S := \{\mathcal{R}_{S,t}\}_{t=0}^\infty$ induced by prefixes of auxiliary sequence $\{w_t\}_{t=0}^\infty$ and operator $\overleftarrow{*}$ is reversible if w_0 is invertible in the ring.*

We use $\mathbf{R} \in \mathbf{Fun}(\mathfrak{M}, \mathfrak{N})$ to denote the functor induced by reversible \mathcal{R}_S and operator $\overleftarrow{*}$, then we have the following theorem about its impact on the state dependency structure of the NMDP.

Theorem 6 (Impact of \mathbf{R} on State Dependency Structure). *For any MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$, if it is non-degenerate, then the functor \mathbf{R} induced by prefixes of auxiliary sequence $\{w_\tau\}_{\tau=0}^\infty$ and operator $\overleftarrow{*}$ ensures that the dependency structure $D_{h_t'}^t$ of any history $h_t' \in H_t'$ of $\mathbf{R}(\mathcal{M})$ satisfies $D_{h_t'}^t = \{t - \tau \mid (\mathbf{w}^{-1})_{0,\tau} \neq 0\}$, where \mathbf{w}^{-1} is the inverse matrix of \mathbf{w} in the ring, 0 is the zero element of the ring.*

The proof of Theorem 6 indicates that it is possible to achieve any specific state dependency structure in the history of the constructed NMDP by modifying the elements of the upper triangular band matrix \mathbf{w}^{-1} . This provides a general method for designing NMDPs with desired state dependency structures.

5 Experiments

Although the main contribution of this paper is theoretical, we have designed and implemented several highly versatile environment wrappers based on the two methods proposed in Chapter 4 for constructing functors in $\mathbf{Fun}(\mathfrak{M}, \mathfrak{N})$: the group-operator-based and the convolution-based method. These wrappers are implemented using a computationally efficient approach with a time complexity of $O(n)$ and can be applied to environments that conform to the Gymnasium [Farama-Foundation, 2023] interface.

We use the PPO algorithm implemented in the Stable-Baselines3 library [DLR-RM, 2024b] and the LSTM-PPO algorithm of the Stable-Baselines3-Contrib library [Stable-Baselines-Team, 2024] as standard reinforcement learning algorithms. These algorithms are tested in Gymnasium environments wrapped with our wrappers, with reinforcement learning training experiments conducted using the framework provided by the RL-Baselines3-Zoo library [DLR-RM, 2024a].

We assume $s_{-1} := \mathbf{0}$, $\lambda \in [0, 1]$, $\exists k \forall t, s_t \in \mathbb{R}^k$, and employ the following functor to implement wrappers:

- **S** induced by HAS $\mathcal{S}_{S,t} : h_t \mapsto \sum_{\tau=0}^t s_\tau$;
- **D** induced by HAS $\mathcal{D}_{S,t} : h_t \mapsto s_t - s_{t-1}$;
- **S $_\lambda$** induced by HAS $\mathcal{S}_{S,t}^\lambda : h_t \mapsto \sum_{\tau=0}^t \lambda^\tau s_{t-\tau}$;

- D_λ induced by HAS $\mathcal{D}_{S,t}^\lambda : h_t \mapsto s_t - \lambda s_{t-1}$.

From the definitions of S , D , S_λ , D_λ , we can derive the state dependency structure of histories in the NMDP under their application. S , S_λ introduce dependencies of s_t , s_{t-1} , and D , D_λ introduce dependencies of $s_{0:t}$. The state dependency weights induced by S_λ and D_λ exhibit an exponential decay with respect to the temporal distance from the current time step, governed by the factor λ , while S and D introduce uniformly distributed dependency.

Our experiments use CartPole-v1 and Pendulum-v1 as the base MDP environments, and we employ the following functors to obtain the NMDP environments:

$$S^n, D^n \text{ for } n \in \{0, 1, 2, 3, 4, 5\}$$

$$S_\lambda, D_\lambda \text{ for } \lambda \in \{\frac{0}{5}, \frac{1}{5}, \frac{2}{5}, \frac{3}{5}, \frac{4}{5}, \frac{5}{5}\}$$

All hyperparameters required for training were kept at their default settings from the RL-Baselines3-Zoo library, and the training process was initiated using the library’s provided command-line instructions. Each combination of environment, wrapper, and algorithm was trained 3 times, 10^6 timesteps for each time, evaluated every 2×10^5 timesteps, and the checkpoints with the best average episode reward were selected.

Theoretically, an increase in n of S^n and D^n or λ of S_λ and D_λ makes the non-Markov environment more challenging for RL algorithms to solve, as it introduces more complex temporal dependencies. As shown in 1, the results of our experiments also indicate this property: the average episode reward, plotted as a function of n or λ , generally shows a declining trend for each combination of environment, algorithm, and wrapper.

Another conclusion drawn from the experimental results is that, for any given combination of environment and algorithm, the performance degradation caused by increasing λ from 0 to 1 is comparable to the degradation resulting from increasing n from 0 to 1. This observation is consistent with our theoretical analysis, which shows that increasing λ from 0 to 1 progressively introduces a greater dependence on previous states, eventually matching the dependence introduced by a wrapper with $n = 1$.

By comparing the performance of the PPO and LSTM-PPO algorithms in environments wrapped with varying degrees of dependency complexity, we observe that LSTM-PPO outperforms PPO when the dependencies are more complex. This is due to LSTM-PPO’s ability to recall past states, which also aligns with our intuition.

6 Summary and Future Work

We propose a general and effective method to construct NMDPs from MDPs using techniques from category theory and algebra. Our method provides a comprehensive way to evaluate decision-making, especially RL algorithms’ ability to handle non-Markovianity through encoding and memorization.

Theoretical analysis demonstrates that our method possesses strong expressive power and can represent a wide variety of temporal state dependency structures. Using this theoretical framework, we transformed classical Markovian environments into non-Markovian ones and tested them with

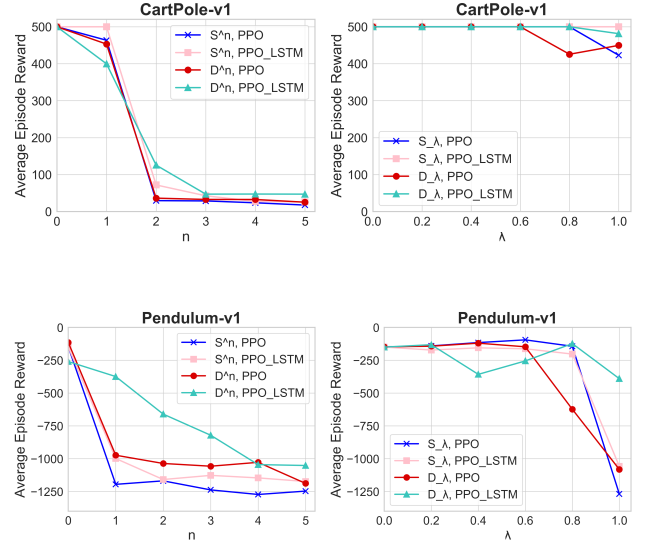


Figure 1: Experimental results for each combination of environment, algorithm, and wrapper. The x-axis represents the wrapper parameter, while the y-axis shows the average episode reward. Lines of different colors indicate different combinations of algorithms and wrapper types.

both PPO algorithms, with and without LSTM. This validation confirms our method’s effectiveness in introducing varying degrees of non-Markovian characteristics into Markovian environments.

The examples tested in our experiments represent only a small subset of the cases within the theoretical framework. The functor forms that exhibit desirable properties through our theory are highly diverse, extending beyond those tested. This is due to our proofs being conducted within abstract algebraic structures, which encompass numerous notable specific cases.

Our study also has limitations that warrant further investigation. For instance, we did not explore the introduction of randomness into HAS using random sequences or mappings. Additionally, we did not address the theory of HAR, which remains relevant for reward shaping. Moreover, our theoretical analysis is restricted to reversible HAS, and we have not developed a theory for non-reversible cases. For example, a common method to create non-Markovian environments is by obscuring information in the observations of an MDP to transform it into a POMDP. Combining this method with our approach and exploring their combined expressive power presents a promising avenue for future research.

Ethical Statement

There are no ethical issues.

Acknowledgments

The authors have no acknowledgments to report.

References

- Eden Abadi and Ronen I Brafman. Learning and solving regular decision processes. *arXiv preprint arXiv:2003.01008*, 2020.
- Ronen I Brafman, Giuseppe De Giacomo, et al. Regular decision processes: A model for non-markovian domains. In *IJCAI*, pages 5516–5522, 2019.
- Alberto Camacho, Rodrigo Toro Icarte, Torny Q Klassen, Richard Anthony Valenzano, and Sheila A McIlraith. Ltl and beyond: Formal languages for reward function specification in reinforcement learning. In *IJCAI*, volume 19, pages 6065–6073, 2019.
- Siddharth Chandak, Pratik Shah, Vivek S Borkar, and Parth Dohia. Reinforcement learning in non-markovian environments. *Systems & Control Letters*, 185:105751, 2024.
- Inés de Vega and Daniel Alonso. Dynamics of non-markovian open quantum systems. *Rev. Mod. Phys.*, 89:015001, Jan 2017.
- DLR-RM. Rl-baselines3-zoo, 2024.
- DLR-RM. Stable-baselines3, 2024.
- Taylor Dohmen, Noah Topper, George Atia, Andre Beckus, Ashutosh Trivedi, and Alvaro Velasquez. Inferring probabilistic reward machines from non-markovian reward signals for reinforcement learning. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 32, pages 574–582, 2022.
- Farama-Foundation. Gymnasium, 2023.
- Maor Gaon and Ronen Brafman. Reinforcement learning with non-markovian rewards. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 3980–3987, 2020.
- Gaurav Gupta, Chenzhong Yin, Jyotirmoy V Deshmukh, and Paul Bogdan. Non-markovian reinforcement learning using fractional dynamics. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 1542–1547. IEEE, 2021.
- Marcus Hutter. Feature reinforcement learning: Part i. unstructured mdps. *Journal of Artificial General Intelligence*, 1(1):3, 2009.
- Tor Lattimore, Marcus Hutter, and Peter Sunehag. The sample-complexity of general reinforcement learning. In *International Conference on Machine Learning*, pages 28–36. PMLR, 2013.
- Tom Leinster. *Basic category theory*, volume 143. Cambridge University Press, 2014.
- Odalric-Ambrym Maillard, Daniil Ryabko, and Rémi Munos. Selecting the state-representation in reinforcement learning. *Advances in Neural Information Processing Systems*, 24, 2011.
- Sultan Javed Majeed, Marcus Hutter, et al. On q-learning convergence for non-markov decision processes. In *IJCAI*, volume 18, pages 2546–2552, 2018.
- Phuong Nguyen, Odalric-Ambrym Maillard, Daniil Ryabko, and Ronald Ortner. Competing with an infinite set of models in reinforcement learning. In *Artificial Intelligence and Statistics*, pages 463–471. PMLR, 2013.
- Aoyang Qin, Feng Gao, Qing Li, Song-Chun Zhu, and Sirui Xie. Learning non-markovian decision-making from state-only sequences. *Advances in Neural Information Processing Systems*, 36, 2024.
- Gavin Rens and Jean-François Raskin. Learning non-markovian reward models in mdps. *arXiv preprint arXiv:2001.09293*, 2020.
- Alessandro Ronca, Gabriel Paludo Licks, and Giuseppe De Giacomo. Markov abstractions for pac reinforcement learning in non-markov decision processes. *arXiv preprint arXiv:2205.01053*, 2022.
- Stable-Baselines-Team. Stable-baselines3-contrib, 2024.
- Joel Veness, Kee Siong Ng, Marcus Hutter, William Uther, and David Silver. A monte-carlo aixo approximation. *Journal of Artificial Intelligence Research*, 40:95–142, 2011.
- Steven D Whitehead and Long-Ji Lin. Reinforcement learning of non-markov decision processes. *Artificial intelligence*, 73(1-2):271–306, 1995.

Appendix: Constructing Non-Markovian Decision Process via History Aggregator

Yongyi Wang^{1,2}, Wenxin Li^{1,2}

¹AILab, School of Computer Science, Peking University, Beijing 100871, China.

²National Key Laboratory for Multimedia Information Processing, School of Computer Science, Peking University, Beijing 100871, China.

{wangyongyi, lwx}@pku.edu.cn

Code: Considering anonymity, the code on Github will be made public after this article is accepted.

1 Proof of Well-Definedness of MDP Category

Lemma 1. *MDP Category \mathfrak{M} is well-defined.*

Proof. It is sufficient to prove that the composition of morphisms remains a morphism.

For any MDP $\mathcal{M}, \mathcal{M}', \mathcal{M}'' \in \mathfrak{M}$ and any morphism: $\phi : \mathcal{M} \rightarrow \mathcal{M}'$, $\phi' : \mathcal{M}' \rightarrow \mathcal{M}''$, by definition of composition of MDP category we have

$$\phi' \circ \phi = (\phi'_S \circ \phi_S, \phi'_A \circ \phi_A, \phi'_R \circ \phi_R)$$

Since

$$\rho_0 = \rho'_0 \circ \phi_S$$

$$\rho'_0 = \rho''_0 \circ \phi'_S$$

we have

$$\rho_0 = (\rho''_0 \circ \phi'_S) \circ \phi_S = \rho''_0 \circ (\phi'_S \circ \phi_S) = \rho''_0 \circ (\phi' \circ \phi)_S$$

similarly, since

$$T_t(s, a) = ((T'_t \circ (\phi_S, \phi_A))(s, a)) \circ (\phi_S, \phi_R)$$

$$T'_t(s', a') = ((T''_t \circ (\phi'_S, \phi'_A))(s', a')) \circ (\phi'_S, \phi'_R)$$

take $(s', a') = (\phi_S, \phi_A)(s, a)$, we have

$$T_t(s, a) = ((T'_t \circ (\phi_S, \phi_A))(s, a)) \circ (\phi_S, \phi_R)$$

$$= (T'_t((\phi_S, \phi_A)(s, a))) \circ (\phi_S, \phi_R)$$

$$= (T'_t(s', a')) \circ (\phi_S, \phi_R)$$

$$= (((T''_t \circ (\phi'_S, \phi'_A))(s', a')) \circ (\phi'_S, \phi'_R)) \circ (\phi_S, \phi_R)$$

$$= ((T''_t \circ (\phi'_S, \phi'_A))(s', a')) \circ ((\phi' \circ \phi)_S, (\phi' \circ \phi)_R)$$

$$= ((T''_t \circ (\phi'_S, \phi'_A))((\phi_S, \phi_A)(s, a))) \circ ((\phi' \circ \phi)_S, (\phi' \circ \phi)_R)$$

$$= ((T''_t \circ ((\phi' \circ \phi)_S, (\phi' \circ \phi)_A))(s, a)) \circ ((\phi' \circ \phi)_S, (\phi' \circ \phi)_R)$$

therefore $\phi' \circ \phi$ is still a morphism, i.e. \mathfrak{M} is a well-defined category. \square

2 Proof of Well-Definedness of NMDP Category

Lemma 2. *NMDP Category \mathfrak{N} is well-defined.*

Proof. It is sufficient to prove that the composition of morphisms remains a morphism.

For any NMDP $\mathcal{N}, \mathcal{N}', \mathcal{N}'' \in \mathfrak{N}$ and any morphism: $\psi : \mathcal{N} \rightarrow \mathcal{N}'$, $\psi' : \mathcal{N}' \rightarrow \mathcal{N}''$, by definition of composition of NMDP category we have

$$\psi' \circ \psi = (\psi'_S \circ \psi_S, \psi'_A \circ \psi_A, \psi'_R \circ \psi_R)$$

Since

$$\rho_0 = \rho'_0 \circ \psi_S$$

$$\rho'_0 = \rho''_0 \circ \psi'_S$$

we have

$$\rho_0 = (\rho''_0 \circ \psi'_S) \circ \psi_S = \rho''_0 \circ (\psi'_S \circ \psi_S) = \rho''_0 \circ (\psi' \circ \psi)_S$$

similarly, we have

$$T_t(h_t, a) = ((T'_t \circ (\psi_{H_t}, \psi_A))(h_t, a)) \circ (\psi_S, \psi_R)$$

$$T'_t(h'_t, a') = ((T''_t \circ (\psi'_{H_t}, \psi'_A))(h'_t, a')) \circ (\psi'_S, \psi'_R)$$

take $(h'_t, a') = (\psi_{H_t}, \psi_A)(h_t, a)$, we have

$$T_t(h_t, a) = ((T'_t \circ (\psi_{H_t}, \psi_A))(h_t, a)) \circ (\psi_S, \psi_R)$$

$$= (T'_t((\psi_{H_t}, \psi_A)(h_t, a))) \circ (\psi_S, \psi_R)$$

$$= (T'_t(h'_t, a')) \circ (\psi_S, \psi_R)$$

$$= (((T''_t \circ (\psi'_{H_t}, \psi'_A))(h'_t, a')) \circ (\psi'_S, \psi'_R)) \circ (\psi_S, \psi_R)$$

$$= ((T''_t \circ (\psi'_{H_t}, \psi'_A))(h'_t, a')) \circ ((\psi' \circ \psi)_S, (\psi' \circ \psi)_R)$$

$$= ((T''_t \circ (\psi'_{H_t}, \psi'_A))((\psi_{H_t}, \psi_A)(h_t, a))) \circ ((\psi' \circ \psi)_S, (\psi' \circ \psi)_R)$$

$$= ((T''_t \circ (\psi'_{H_t} \circ \psi_{H_t}, (\psi' \circ \psi)_A))(h_t, a)) \circ ((\psi' \circ \psi)_S, (\psi' \circ \psi)_R)$$

since

$$\psi_{H_t} = (\psi_S^{(t+1)}, \psi_A^{(t)}, \psi_R^{(t)})$$

$$\psi'_{H_t} = (\psi_S'^{(t+1)}, \psi_A'^{(t)}, \psi_R'^{(t)})$$

we have

$$\psi'_{H_t} \circ \psi_{H_t} = (\psi_S'^{(t+1)}, \psi_A'^{(t)}, \psi_R'^{(t)}) \circ (\psi_S^{(t+1)}, \psi_A^{(t)}, \psi_R^{(t)})$$

$$= (\psi_S'^{(t+1)} \circ \psi_S^{(t+1)}, \psi_A'^{(t)} \circ \psi_A^{(t)}, \psi_R'^{(t)} \circ \psi_R^{(t)})$$

$$= ((\psi' \circ \psi)_S^{(t+1)}, (\psi' \circ \psi)_A^{(t)}, (\psi' \circ \psi)_R^{(t)})$$

$$= (\psi' \circ \psi)_{H_t}$$

therefore

$$T_t(h_t, a) = ((T''_t \circ ((\psi' \circ \psi)_{H_t}, (\psi' \circ \psi)_A))(h_t, a)) \circ ((\psi' \circ \psi)_S, (\psi' \circ \psi)_R)$$

which means $\psi' \circ \psi$ is still a morphism, i.e. \mathfrak{N} is a well-defined category. \square

3 Proof of Well-Definedness of Non-Markov Embedding as a Functor

Lemma 3. *Non-Markov embedding \mathbf{N} is a well-defined functor from \mathfrak{M} to \mathfrak{N} if $\forall \mathcal{M}, \mathcal{M}' \in \mathfrak{M}, \forall \phi \in \mathfrak{M}(\mathcal{M}, \mathcal{M}'), \mathbf{N}(\phi) = \phi$.*

Proof. For any $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$, $\mathcal{M}' = \langle \rho'_0, S', A', \{T'_t\}_{t=0}^\infty \rangle$, $\mathcal{M}'' = \langle \rho''_0, S'', A'', \{T''_t\}_{t=0}^\infty \rangle \in \mathfrak{M}$ and any morphism $\phi \in \mathfrak{M}(\mathcal{M}, \mathcal{M}')$, $\phi' \in \mathfrak{M}(\mathcal{M}', \mathcal{M}'')$, denote $\mathcal{N} := \mathbf{N}(\mathcal{M}) = \langle \rho_0, S, A, \{J_t\}_{t=0}^\infty \rangle$ and $\mathcal{N}' := \mathbf{N}(\mathcal{M}') = \langle \rho'_0, S', A', \{J'_t\}_{t=0}^\infty \rangle$.

If the condition $\forall \phi, \mathbf{N}(\phi) = \phi$ holds, then

$$\mathbf{N}(1_{\mathcal{M}}) = 1_{\mathcal{M}} = (1_S, 1_A, 1_{\mathbb{R}}) = 1_{\mathbf{N}(\mathcal{M})}$$

and

$$\mathbf{N}(\phi' \circ \phi) = \phi' \circ \phi = \mathbf{N}(\phi') \circ \mathbf{N}(\phi)$$

which means the identity morphisms and associativity of morphisms in \mathfrak{M} is preserved by \mathbf{N} .

Therefore, for the well-definedness of \mathbf{N} , we only need to prove that the condition $\forall \phi, \mathbf{N}(\phi) = \phi$ is compatible with the action of \mathbf{N} on $ob(\mathfrak{M})$ which is given by the definition of \mathbf{N} , i.e. $\mathbf{N}(\phi) = \phi \in \mathfrak{N}(\mathcal{N}, \mathcal{N}')$.

Since \mathbf{N} does nothing to $\rho_0, \rho'_0; S, S'; A, A'$, we only need to prove that the condition is compatible with the action of \mathbf{N} on T_t, T'_t .

By the definition of morphisms in \mathfrak{M} , we have

$$T_t(s, a) = ((T'_t \circ (\phi_S, \phi_A))(s, a)) \circ (\phi_S, \phi_{\mathbb{R}})$$

By the definition of \mathbf{N} , we have

$$J_t(h_t, a) = (T_t \circ (\mathcal{L}_{S,t}, 1_A))(h_t, a)$$

$$J'_t(h'_t, a') = (T'_t \circ (\mathcal{L}'_{S',t}, 1_{A'}))(h'_t, a')$$

denote $s := \mathcal{L}_{S,t}(h_t)$, $s' := \mathcal{L}'_{S',t}(h'_t)$ then

$$\begin{aligned} J_t(h_t, a) &= (T_t \circ (\mathcal{L}_{S,t}, 1_A))(h_t, a) \\ &= T_t((\mathcal{L}_{S,t}, 1_A)(h_t, a)) \\ &= T_t(s, a) \\ &= ((T'_t \circ (\phi_S, \phi_A))(s, a)) \circ (\phi_S, \phi_{\mathbb{R}}) \\ &= (T'_t((\phi_S, \phi_A)(s, a))) \circ (\phi_S, \phi_{\mathbb{R}}) \\ &= (T'_t(s', a')) \circ (\phi_S, \phi_{\mathbb{R}}) \\ &= (T'_t((\mathcal{L}'_{S',t}, 1_{A'})(h'_t, a'))) \circ (\phi_S, \phi_{\mathbb{R}}) \\ &= ((T'_t \circ (\mathcal{L}'_{S',t}, 1_{A'}))(h'_t, a')) \circ (\phi_S, \phi_{\mathbb{R}}) \\ &= (J'_t(h'_t, a')) \circ (\phi_S, \phi_{\mathbb{R}}) \\ &= (J'_t((\phi_{H_t}, \phi_A)(h_t, a))) \circ (\phi_S, \phi_{\mathbb{R}}) \\ &= ((J'_t \circ (\phi_{H_t}, \phi_A))(h_t, a)) \circ (\phi_S, \phi_{\mathbb{R}}) \end{aligned}$$

therefore, by definition of morphisms in \mathfrak{N} , $\phi \in \mathfrak{N}(\mathcal{N}, \mathcal{N}')$, which means $\forall \phi, \mathbf{N}(\phi) = \phi$ is compatible with the action of \mathbf{N} on T_t, T'_t , i.e. \mathbf{N} is a well-defined functor from \mathfrak{M} to \mathfrak{N} . \square

4 Proof of Well-Definedness of Markov Abstraction as a Functor

Lemma 4. *Markov abstraction \mathbf{M} is a well-defined functor from \mathfrak{N} to \mathfrak{M} if $\forall \mathcal{N}, \mathcal{N}' \in \mathfrak{N}, \forall \psi = (\psi_S, \psi_A, \psi_{\mathbb{R}}) \in \mathfrak{N}(\mathcal{N}, \mathcal{N}'), \mathbf{M}(\psi) = (\psi_H, \psi_A, \psi_{\mathbb{R}})$, where*

$$\begin{aligned} H &:= \bigsqcup_{t=0}^\infty H_t = \bigsqcup_{t=0}^\infty S^{t+1} \times A^t \times \mathbb{R}^t \\ H' &:= \bigsqcup_{t=0}^\infty H'_t = \bigsqcup_{t=0}^\infty S'^{t+1} \times A'^t \times \mathbb{R}^t \end{aligned}$$

and $\forall t \in \mathbb{N}$

$$\psi_H : (s_{0:t}, a_{0:t-1}, r_{0:t-1}) \mapsto \psi_{H_t}(s_{0:t}, a_{0:t-1}, r_{0:t-1})$$

Proof. For any $\mathcal{N} = \langle \rho_0, S, A, \{J_t\}_{t=0}^\infty \rangle$, $\mathcal{N}' = \langle \rho'_0, S', A', \{J'_t\}_{t=0}^\infty \rangle$, $\mathcal{N}'' = \langle \rho''_0, S'', A'', \{J''_t\}_{t=0}^\infty \rangle \in \mathfrak{N}$ and any morphism $\psi \in \mathfrak{N}(\mathcal{N}, \mathcal{N}')$, $\psi' \in \mathfrak{N}(\mathcal{N}', \mathcal{N}'')$, denote $\mathcal{M} := \mathbf{M}(\mathcal{N}) = \langle \rho_0, H, A, \{T_t\}_{t=0}^\infty \rangle$ and $\mathcal{M}' := \mathbf{M}(\mathcal{N}') = \langle \rho'_0, H', A', \{T'_t\}_{t=0}^\infty \rangle$, where $H := \bigsqcup_{t=0}^\infty H_t = \bigsqcup_{t=0}^\infty S^{t+1} \times A^t \times \mathbb{R}^t$, $H' := \bigsqcup_{t=0}^\infty H'_t = \bigsqcup_{t=0}^\infty S'^{t+1} \times A'^t \times \mathbb{R}^t$.

If the condition $\forall \psi, \mathbf{M}(\psi) = (\psi_H, \psi_A, \psi_{\mathbb{R}})$ holds, then

$$\mathbf{M}(1_{\mathcal{N}}) = 1_{\mathcal{N}} = (1_H, 1_A, 1_{\mathbb{R}}) = 1_{\mathbf{M}(\mathcal{N})}$$

and

$$\begin{aligned} \mathbf{M}(\psi' \circ \psi) &= \mathbf{M}((\psi' \circ \psi)_S, (\psi' \circ \psi)_A, (\psi' \circ \psi)_{\mathbb{R}}) \\ &= ((\psi' \circ \psi)_H, (\psi' \circ \psi)_A, (\psi' \circ \psi)_{\mathbb{R}}) \\ &= (\psi'_H \circ \psi_H, \psi'_A \circ \psi_A, \psi'_{\mathbb{R}} \circ \psi_{\mathbb{R}}) \\ &= (\psi'_H, \psi'_A, \psi'_{\mathbb{R}}) \circ (\psi_H, \psi_A, \psi_{\mathbb{R}}) \\ &= \mathbf{M}(\psi') \circ \mathbf{M}(\psi) \end{aligned}$$

which means the identity morphisms and associativity of morphisms in \mathfrak{N} is preserved by \mathbf{M} .

Therefore, for the well-definedness of \mathbf{M} , we only need to prove that the condition $\forall \psi, \mathbf{M}(\psi) = (\psi_H, \psi_A, \psi_{\mathbb{R}})$ is compatible with the action of \mathbf{M} on $ob(\mathfrak{N})$ which is given by the definition of \mathbf{M} , i.e. $\mathbf{M}(\psi) = (\psi_H, \psi_A, \psi_{\mathbb{R}}) \in \mathfrak{M}(\mathcal{M}, \mathcal{M}')$. Since \mathbf{M} does nothing to $\rho_0, \rho'_0; A, A'$, we only need to prove that the condition is compatible with the action of \mathbf{M} on $S, S'; J_t, J'_t$.

For any $s \in S$, let $s' = \psi_S(s) \in S'$. and $h_t = (s_{0:t}, a_{0:t-1}, r_{0:t-1}) \in H$, if $\forall i, s'_i = \psi_S(s_i), a'_i = \psi_A(a_i), r'_i = \psi_{\mathbb{R}}(r_i)$, then $h'_t = (s'_{0:t}, a'_{0:t-1}, r'_{0:t-1}) \in H'$. We have

$$\begin{aligned} h'_t &= (s'_{0:t}, a'_{0:t-1}, r'_{0:t-1}) \\ &= (\psi_S^{(t+1)}(s_{0:t}), \psi_A^{(t)}(a_{0:t-1}), \psi_{\mathbb{R}}^{(t)}(r_{0:t-1})) \\ &= (\psi_S^{(t+1)}, \psi_A^{(t)}, \psi_{\mathbb{R}}^{(t)})(s_{0:t}, a_{0:t-1}, r_{0:t-1}) \\ &= \psi_H(h_t) \end{aligned}$$

therefore the condition is compatible with the action of \mathbf{M} on S, S' .

By the definition of morphisms in \mathfrak{N} , we have

$$J_t(h_t, a) = ((J'_t \circ (\psi_{H_t}, \psi_A))(h_t, a)) \circ (\psi_S, \psi_{\mathbb{R}})$$

By the definition of \mathbf{M} we have

$$T_t(h_t, a) = (J_t(h_t, a)) \circ (\mathcal{L}_{S,t+1}, 1_{\mathbb{R}})$$

$$T'_t(h'_t, a') = (J'_t(h'_t, a')) \circ (\mathcal{L}_{S',t+1}, 1_{\mathbb{R}})$$

For any $h_{t+1} := (s_{0:t+1}, a_{0:t}, r_{0:t}) \in H_{t+1} \subset H$, we have

$$(\psi_S \circ \mathcal{L}_{S,t+1})(h_{t+1}) = \psi_S(s_{t+1})$$

and

$$\begin{aligned} (\mathcal{L}_{S',t+1} \circ \psi_H)(h_{t+1}) &= \mathcal{L}_{S',t+1}(\psi_{H_{t+1}}(h_{t+1})) \\ &= \mathcal{L}_{S',t+1}(\psi_S^{(t+2)}(s_{0:t+1}), \psi_A^{(t+1)}(a_{0:t}), \psi_{\mathbb{R}}^{(t+1)}(r_{0:t})) \\ &= \psi_S(s_{t+1}) \end{aligned}$$

thus $\psi_S \circ \mathcal{L}_{S,t+1} = \mathcal{L}_{S',t+1} \circ \psi_H$, which implies that

$$\begin{aligned} T_t(h_t, a) &= (J_t(h_t, a)) \circ (\mathcal{L}_{S,t+1}, 1_{\mathbb{R}}) \\ &= ((J'_t \circ (\psi_{H_t}, \psi_A))(h_t, a)) \circ (\psi_S, \psi_{\mathbb{R}}) \circ (\mathcal{L}_{S,t+1}, 1_{\mathbb{R}}) \\ &= ((J'_t \circ (\psi_{H_t}, \psi_A))(h_t, a)) \circ (\psi_S \circ \mathcal{L}_{S,t+1}, \psi_{\mathbb{R}}) \\ &= ((J'_t \circ (\psi_{H_t}, \psi_A))(h_t, a)) \circ (\mathcal{L}_{S',t+1} \circ \psi_H, 1_{\mathbb{R}} \circ \psi_{\mathbb{R}}) \\ &= (((J'_t \circ (\psi_{H_t}, \psi_A))(h_t, a)) \circ (\mathcal{L}_{S',t+1}, 1_{\mathbb{R}})) \circ (\psi_H, \psi_{\mathbb{R}}) \\ &= ((J'_t((\psi_{H_t}, \psi_A)(h_t, a))) \circ (\mathcal{L}_{S',t+1}, 1_{\mathbb{R}})) \circ (\psi_H, \psi_{\mathbb{R}}) \\ &= ((J'_t(h'_t, a')) \circ (\mathcal{L}_{S',t+1}, 1_{\mathbb{R}})) \circ (\psi_H, \psi_{\mathbb{R}}) \\ &= (T'_t(h'_t, a')) \circ (\psi_H, \psi_{\mathbb{R}}) \\ &= (T'_t((\psi_H, \psi_A)(h_t, a))) \circ (\psi_H, \psi_{\mathbb{R}}) \\ &= ((T'_t \circ (\psi_H, \psi_A))(h_t, a)) \circ (\psi_H, \psi_{\mathbb{R}}) \end{aligned}$$

therefore, by definition of morphisms in \mathfrak{M} , $(\psi_H, \psi_A, \psi_{\mathbb{R}}) \in \mathfrak{M}(\mathcal{M}, \mathcal{M}')$, which means $\forall \psi, \mathbf{M}(\psi) = (\psi_H, \psi_A, \psi_{\mathbb{R}})$ is compatible with the action of \mathbf{M} on J_t, J'_t , i.e. \mathbf{M} is a well-defined functor from \mathfrak{N} to \mathfrak{M} . \square

5 Proof of Theorem 1

Definition 1 (Isomorphism). A morphism $f : A \rightarrow B$ in a category \mathfrak{A} is an isomorphism if there exists a map $g : B \rightarrow A$ in \mathfrak{A} such that $g \circ f = 1_A$, $f \circ g = 1_B$.

Definition 2 (Natural Transformation). Let \mathfrak{A} and \mathfrak{B} be categories and let $\mathfrak{A} \xrightarrow[\mathbf{G}]{\mathbf{F}} \mathfrak{B}$ be functors. A natural transformation $\alpha : \mathbf{F} \rightarrow \mathbf{G}$ is a family $(\mathbf{F}(A) \xrightarrow{\alpha_A} \mathbf{G}(A))_{A \in \mathfrak{A}}$ of

morphisms in \mathfrak{B} such that for every morphism $A \xrightarrow{f} A'$ in \mathfrak{A} , the square

$$\begin{array}{ccc} \mathbf{F}(A) & \xrightarrow{\mathbf{F}(f)} & \mathbf{F}(A') \\ \downarrow \alpha_A & & \downarrow \alpha_{A'} \\ \mathbf{G}(A) & \xrightarrow{\mathbf{G}(f)} & \mathbf{G}(A') \end{array}$$

commutes. The maps α_A are called the components of α .

Definition 3 (Functor Category). Let \mathfrak{A} and \mathfrak{B} be categories. Functor category $\mathbf{Fun}(\mathfrak{A}, \mathfrak{B})$ is a category whose objects are all the functors from \mathfrak{A} to \mathfrak{B} and morphisms are all the natural transformations between these functors.

Definition 4 (Natural Isomorphism). Let \mathfrak{A} and \mathfrak{B} be categories. A natural isomorphism between functors \mathbf{F} and \mathbf{G} from \mathfrak{A} to \mathfrak{B} is an isomorphism in category $\mathbf{Fun}(\mathfrak{A}, \mathfrak{B})$. Functors $\mathbf{F}, \mathbf{G} \in \mathbf{Fun}(\mathfrak{A}, \mathfrak{B})$ are isomorphic (denoted as $\mathbf{F} \cong \mathbf{G}$) if there exists a natural isomorphism between them.

Definition 5 (Category Equivalence). A equivalence between categories \mathfrak{A} and \mathfrak{B} consists of a pair of functors together with natural isomorphisms $\eta : 1_{\mathfrak{A}} \rightarrow \mathbf{G} \circ \mathbf{F}$, $\epsilon : \mathbf{F} \circ \mathbf{G} \rightarrow 1_{\mathfrak{B}}$. If there exists an equivalence between \mathfrak{A} and \mathfrak{B} , we say that \mathfrak{A} and \mathfrak{B} are equivalent through \mathbf{F} and \mathbf{G} .

Theorem 1 (Equivalence of \mathfrak{M} and \mathfrak{N}). Category \mathfrak{M} and \mathfrak{N} are equivalent through functor \mathbf{M} and \mathbf{N} , i.e. $\mathbf{M} \circ \mathbf{N} \cong 1_{\mathfrak{M}}$, $\mathbf{N} \circ \mathbf{M} \cong 1_{\mathfrak{N}}$.

Proof. For any

$$\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^{\infty} \rangle \in \mathfrak{M}$$

where

$$T_t : (s_t, a_t) \mapsto ((s_{t+1}, r_t) \mapsto \mathbb{P}_{\mathcal{M}}^t(s_t, a_t; s_{t+1}, r_t))$$

we have

$$(\mathbf{M} \circ \mathbf{N})(\mathcal{M}) = \langle \rho'_0, S', A', \{T'_t\}_{t=0}^{\infty} \rangle$$

where

$$\begin{aligned} \rho'_0 &= \rho_0, S' = H, A' = A \\ T'_t &: (h'_t, a_t) \mapsto ((h'_{t+1}, r_t) \mapsto \mathbb{P}_{\mathcal{M}}^t(s_t, a_t; s_{t+1}, r_t)) \\ h'_t &:= (\{h_{\tau}\}_{\tau=0}^t, \{a_{\tau}\}_{\tau=0}^{t-1}, \{r_{\tau}\}_{\tau=0}^{t-1}) \\ h_t &:= (\{s_{\tau}\}_{\tau=0}^t, \{a_{\tau}\}_{\tau=0}^{t-1}, \{r_{\tau}\}_{\tau=0}^{t-1}) \end{aligned}$$

therefore, we can reconstruct the representation of T_t using T'_t without loss of information and vice versa,

$$(T_t(s_t, a_t))(s_{t+1}, r_t) = (T'_t(\mathcal{L}_{S,t}^{-2}(s_t), a_t))(\mathcal{L}_{S,t+1}^{-2}(s_{t+1}), r_t)$$

$$(T'_t(h'_t, a_t))(h'_{t+1}, r_t) = (T_t(\mathcal{L}_{S,t}^2(h'_t), a_t))(\mathcal{L}_{S,t+1}^2(h'_{t+1}), r_t)$$

which means $\mathbf{M} \circ \mathbf{N} \cong 1_{\mathfrak{M}}$.

Similarly, for any

$$\mathcal{N} \in \langle \rho_0, S, A, \{J_t\}_{t=0}^{\infty} \rangle \in \mathfrak{N}$$

where

$$J_t : (h_t, a_t) \mapsto ((s_{t+1}, r_t) \mapsto \mathbb{P}_{\mathcal{N}}^t(h_t, a_t; s_{t+1}, r_t))$$

we have

$$(\mathbf{N} \circ \mathbf{M})(\mathcal{N}) = \langle \rho'_0, S', A', \{J'_t\}_{t=0}^{\infty} \rangle$$

where

$$\begin{aligned} \rho'_0 &= \rho_0, S' = H, A' = A \\ J'_t &: (h'_t, a_t) \mapsto ((h'_{t+1}, r_t) \mapsto \mathbb{P}_{\mathcal{N}}^t(h_t, a_t; s_{t+1}, r_t)) \\ h'_t &:= (\{h_{\tau}\}_{\tau=0}^t, \{a_{\tau}\}_{\tau=0}^{t-1}, \{r_{\tau}\}_{\tau=0}^{t-1}) \\ h_t &:= (\{s_{\tau}\}_{\tau=0}^t, \{a_{\tau}\}_{\tau=0}^{t-1}, \{r_{\tau}\}_{\tau=0}^{t-1}) \end{aligned}$$

therefore we can reconstruct the representation of J_t using J'_t without loss of information and vice versa,

$$(J_t(h_t, a_t))(s_{t+1}, r_t) = (J'_t(\mathcal{L}_{S,t}^{-1}(h_t), a_t))(\mathcal{L}_{S,t+1}^{-1}(s_{t+1}), r_t)$$

$$(J'_t(h'_t, a_t))(h'_{t+1}, r_t) = (J_t(\mathcal{L}_{S,t}(h'_t), a_t))(\mathcal{L}_{S,t+1}(h'_{t+1}), r_t)$$

which means $\mathbf{N} \circ \mathbf{M} \cong 1_{\mathfrak{N}}$. \square

6 Proof of Theorem 2

Theorem 2 (Reversibility of HAS Induced by Group Operator). *The HAS \mathcal{G}_S on MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$ induced by binary operator \otimes is reversible if (S, \otimes) is a group.*

Proof. Define \mathcal{G}_S^* as follows which satisfy the reversibility condition of \mathcal{G}_S

$$\mathcal{G}_{S,t}^*(\{\mathcal{G}_{S,\tau}(h_\tau)\}_{\tau=0}^t) := (\mathcal{G}_{S,t-1}(h_{t-1}))^{-1} \otimes \mathcal{G}_{S,t}(h_t) = s_t$$

□

Note: Actually, the condition that (S, \otimes) forms a group can be relaxed to (S', \otimes) forms a group where $S \subseteq S'$.

7 Proof of Theorem 3

Theorem 3 (Impact of G on State Dependency Structure). *For any MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$, if it is non-degenerate, then the functor \mathbf{G} induced by some group operator on S ensures that the dependency structure $D_{h_t^n}$ of any history $h_t^n \in H_t^n$ of $\mathbf{G}^n(\mathcal{M})$ satisfies $D_{h_t^n} = [t - n, t] \cap \mathbb{N}$.*

Proof. Based on the closure property of the group operator \otimes , we repeatedly apply \mathbf{G} to MDP $\mathcal{N}_0 := \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$ to acquire a series of NMDPs $\{\mathcal{N}_i\}_{i=1}^n := \{\langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle\}_{i=1}^n$.

$$\mathcal{N} \ni \mathcal{N}_0 \xrightarrow{\mathbf{G}} \mathcal{N}_1 \xrightarrow{\mathbf{G}} \mathcal{N}_2 \xrightarrow{\mathbf{G}} \dots \xrightarrow{\mathbf{G}} \mathcal{N}_n \in \mathfrak{N}$$

Consider history $h_t^i \in H_{\mathcal{N}_i}$, $(\forall i \in [0, n] \cap \mathbb{N})$ with $\mathcal{E}_S(h_t^i) = s_{0:t}^i$. By the definition of \mathbf{G} we have

$$\{\mathcal{G}_{S,\tau}(h_\tau^i)\}_{\tau=0}^t = \{s_\tau^{i+1}\}_{\tau=0}^t = \mathcal{E}_S(h_t^{i+1}) \quad (1)$$

and

$$\{(\mathcal{G}_{S,\tau}(h_\tau^i))^{-1} \otimes \mathcal{G}_{S,\tau+1}(h_{\tau+1}^i)\}_{\tau=0}^{t-1} = \{s_{\tau+1}^i\}_{\tau=0}^{t-1} \quad (2)$$

substituting Equation 1 into Equation 2 yields

$$s_{\tau+1}^i = (s_\tau^{i+1})^{-1} \otimes s_{\tau+1}^{i+1} \quad (3)$$

Starting from s_t^0 , repeatedly applying Equation 3 by substituting the left side into the right side yields an expression of s_t^0 in terms of $s_t^n, s_{t-1}^n, \dots, s_{t-n}^n$:

$$\begin{aligned} s_t^0 &= (s_{t-1}^1)^{-1} s_t^1 \\ &= ((s_{t-2}^2)^{-1} s_{t-1}^2)^{-1} ((s_{t-1}^2)^{-1} s_t^2) \\ &= (s_{t-1}^2)^{-1} s_{t-2}^2 (s_{t-1}^2)^{-1} s_t^2 \\ &= ((s_{t-2}^3)^{-1} s_{t-1}^3)^{-1} ((s_{t-3}^3)^{-1} s_{t-2}^3) ((s_{t-2}^3)^{-1} s_{t-1}^3)^{-1} ((s_{t-1}^3)^{-1} s_t^3) \\ &= (s_{t-1}^3)^{-1} s_{t-2}^3 (s_{t-3}^3)^{-1} s_{t-2}^3 (s_{t-1}^3)^{-1} s_{t-2}^3 (s_{t-1}^3)^{-1} s_t^3 \\ &= \dots \end{aligned}$$

By mathematical induction, it can be shown that

$$s_t^0 = \bigotimes_{i=1}^{2^n} (s_{t_i}^n)^{2^{\chi_{2\mathbb{Z}}(i)-1}} \quad (4)$$

where $\{t_i\}_{i=1}^{2^n} \subseteq [t - n, t] \cap \mathbb{Z}$, $s_{t_i}^n := e$, $(\forall t_i < 0)$, e is the unit element of the group.

Because s_t^0 has the above form, it is sufficient to specify the group operator as the multiplication in the free group to obtain the theorem. □

Note: Free group is not the only form in which (S, \otimes) can be. For example, the theorem still holds when the group is $(\mathbb{R}^k, +)$, $\forall k \in \mathbb{N}^+$.

8 Proof of Theorem 4

Theorem 4 (Reversibility of HAS induced by Auxiliary Sequence and Convolution Operator). *The HAS $\mathcal{R}_S := \{\mathcal{R}_{S,t}\}_{t=0}^\infty$ induced by prefixes of auxiliary sequence $\{w_t\}_{t=0}^\infty$ and operator \star is reversible if w_0 is invertible in the ring.*

Proof. Define \mathcal{R}_S^* as follows which satisfy the reversibility condition of \mathcal{R}_S if w_t is invertible.

$$\mathcal{R}_{S,t}^*(\{\mathcal{R}_{S,\tau}(h_\tau)\}_{\tau=0}^t) := w_t^{-1} \cdot (\mathcal{R}_{S,t}(h_t) \oplus (-\mathcal{R}_{S,t-1}(h_{t-1}))) = s_t$$

□

9 Proof of Theorem 5

Theorem 5 (Reversibility of HAS induced by Auxiliary Sequence and Convolution Operator). *The HAS $\mathcal{R}_S := \{\mathcal{R}_{S,t}\}_{t=0}^\infty$ induced by prefixes of auxiliary sequence $\{w_t\}_{t=0}^\infty$ and operator \star is reversible if w_0 is invertible in the ring.*

Proof. By the definition of convolution operator \star , we have

$$\mathcal{R}_{S,\tau}(h_\tau) = \bigoplus_{i=0}^{\tau} w_i \cdot s_{\tau-i}$$

the following equation comes from taking $\tau \in [0, t] \cap \mathbb{N}$

$$\begin{bmatrix} w_0 & w_1 & w_2 & \dots & w_t \\ 0 & w_0 & w_1 & \dots & w_{t-1} \\ 0 & 0 & w_0 & \dots & w_{t-2} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & w_0 \end{bmatrix} \cdot \begin{bmatrix} s_t \\ s_{t-1} \\ s_{t-2} \\ \dots \\ s_0 \end{bmatrix} = \begin{bmatrix} \mathcal{R}_{S,t}(h_t) \\ \mathcal{R}_{S,t-1}(h_{t-1}) \\ \mathcal{R}_{S,t-2}(h_{t-2}) \\ \dots \\ \mathcal{R}_{S,0}(h_0) \end{bmatrix} \quad (5)$$

If w_0 is invertible in the ring, the Gaussian elimination method can be used to solve Equation 5 and obtain the expression for s_t , which also means the matrix \mathbf{w} is invertible in the ring. □

10 Proof of Theorem 6

Theorem 6 (Impact of R on State Dependency Structure). *For any MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$, if it is non-degenerate, then the functor \mathbf{R} induced by prefixes of auxiliary sequence $\{w_\tau\}_{\tau=0}^\infty$ and operator \star ensures that the dependency structure $D_{h_t'}$ of any history $h_t' \in H_t'$ of $\mathbf{R}(\mathcal{M})$ satisfies $D_{h_t'} = \{t - \tau \mid (\mathbf{w}^{-1})_{0,\tau} \neq 0\}$, where \mathbf{w}^{-1} is the inverse matrix of \mathbf{w} in the ring, 0 is the zero element of the ring.*

Proof. As is proved in Theorem 5, $\mathbf{s} = \mathbf{w}^{-1} \mathbf{r}$, where

$$\mathbf{w}^{-1} = \begin{bmatrix} w_0^{-1} & -w_0^{-1} \cdot w_1 \cdot w_0^{-1} & \dots & \dots & \dots \\ 0 & w_0^{-1} & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & w_0^{-1} & -w_0^{-1} \cdot w_1 \cdot w_0^{-1} \\ 0 & \dots & \dots & 0 & w_0^{-1} \end{bmatrix} \quad (6)$$

is an upper-triangular matrix, therefore

$$s_t = \bigoplus_{\tau=0}^t (\mathbf{w}^{-1})_{0,\tau} \cdot \mathcal{R}_{S,t-\tau}(h_{t-\tau}) \quad (7)$$

If $(\mathbf{w}^{-1})_{0,\tau}$ is not zero, then there exists some aggregated state $s'_{t-\tau} \in \mathcal{R}(H)$ that will change s_t when using it to replace $\mathcal{R}_{S,t-\tau}(h_{t-\tau})$ in Equation 7, which means $t - \tau \in D_{h'_t}$. \square

11 History Aggregator for Reward (HAR)

Unlike HAS, which introduces non-Markovianity into the MDP's transition dynamics, HAR does so in the MDP's reward mechanisms. HAR is similar to reward shaping. However, instead of simplifying RL training as reward shaping does, it is used to construct NMDPs with non-Markovian reward mechanisms.

Definition 6 (History Aggregator for Reward (HAR)). For any MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$, an HAR $\mathcal{A}_{\mathbb{R}} := \{\mathcal{A}_{\mathbb{R},t}\}_{t=0}^\infty$ is a series of maps in which $\mathcal{A}_{\mathbb{R},t} : H_{t+1} \rightarrow \mathbb{R}$, $(\forall t \in \mathbb{N})$.

Definition 7 (Reversibility of HAR on MDP). An HAR $\mathcal{A}_{\mathbb{R}} = \{\mathcal{A}_{\mathbb{R},t}\}_{t=0}^\infty$ on MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$ is reversible iff there exists a series of maps denoted as $\mathcal{A}_{\mathbb{R}}^* := \{\mathcal{A}_{\mathbb{R},t}^*\}_{t=0}^\infty$ which satisfy:

$$\mathcal{A}_{\mathbb{R},t}^*(\{\mathcal{A}_{\mathbb{R},\tau}(h_{\tau+1})\}_{\tau=0}^t) = r_t, \quad (\forall t \in \mathbb{N})$$

where $h_\tau \in H_\tau$, $h_1 \prec h_2 \prec \dots \prec h_{t+1}$, $r_t = \mathcal{L}_{\mathbb{R},t}(h_{t+1})$.

Definition 8 (Application of Reversible HAR to MDP). The application of a reversible HAR $\mathcal{A}_{\mathbb{R}} = \{\mathcal{A}_{\mathbb{R},t}\}_{t=0}^\infty$ to an MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$ is an NMDP $\mathcal{N}' = \langle \rho'_0, S', A', \{T'_t\}_{t=0}^\infty \rangle$, $\rho'_0 = \rho$, $S' = S$, $A' = A$ in which:

$$T'_t : H'_t \times A' \rightarrow \Delta_{S' \times \mathbb{R}} \\ (h'_t, a) \mapsto ((T_t \circ (\mathcal{L}_{S',t}, 1_A))(h'_t, a)) \circ (1_{S'}, G_{\mathbb{R},h'_t})$$

$$\text{where } G_{\mathbb{R},h'_t} : \mathbb{R} \rightarrow \mathbb{R} \\ r'_t \mapsto \mathcal{A}_{\mathbb{R},t}^*(\mathcal{E}_{\mathbb{R}}(h'_t), r'_t) \cdot$$

Corollary 1. The series of latest reward extraction operators, $\mathcal{L}_{\mathbb{R}} := \{\mathcal{L}_{\mathbb{R},t}\}_{t=0}^\infty$, constitutes a reversible HAR on MDP.

12 Extending Reversible HAS to be Applicable on NMDP

- **HAS:** Although HAS was previously described as a structure established on MDP, due to the formal similarity between MDP and NMDP, we can directly replace all occurrences of "MDP" in this definition with "MDP or NMDP". This still constitutes a valid definition.
- **Reversibility of HAS:** Unlike HAS on MDP, to ensure that the target NMDP can utilize the transition function from the original NMDP, it must be possible to reconstruct the complete state sequence from the aggregated history sequence, rather than just the current state. Therefore, a reverse $\mathcal{A}_S^* = \{\mathcal{A}_{S,t}^*\}_{t=0}^\infty$ of an HAS $\mathcal{A}_S = \{\mathcal{A}_{S,t}\}_{t=0}^\infty$ on NMDP $\mathcal{N} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$ should satisfy:

$$\mathcal{A}_{S,t}^*(\{\mathcal{A}_{S,\tau}(h_\tau)\}_{\tau=0}^\infty) = \hat{h}_t \in \mathcal{E}_S^{-1}(\mathcal{E}_S(h_t)) \subseteq H_t, \quad (\forall t \in \mathbb{N})$$

- **Application of Reversible HAS:** The application of reversible HAS on NMDP is similar to that on MDP, with G_{S,h'_t} replaced by the following:

$$G_{S,h'_t} : S' \rightarrow S \\ s'_{t+1} \mapsto (\mathcal{L}_{S,t+1} \circ \mathcal{A}_{S,t+1}^*)(\mathcal{E}_{S'}(h'_t), s'_{t+1})$$

The use of $\mathcal{L}_{S,t+1}$ is to extract the latest state from the history returned by $\mathcal{A}_{S,t+1}^*$, according to the definition of reversed HAS on NMDP.

- **HAS Induced by Binary Operator:** Similar to extending the concept of HAS from MDP to NMDP, in this definition, "MDP" can also be replaced with "MDP or NMDP", which still constitutes a valid definition.

13 Intuitive Explanation and Examples of Key Concepts

13.1 MDP and NMDP

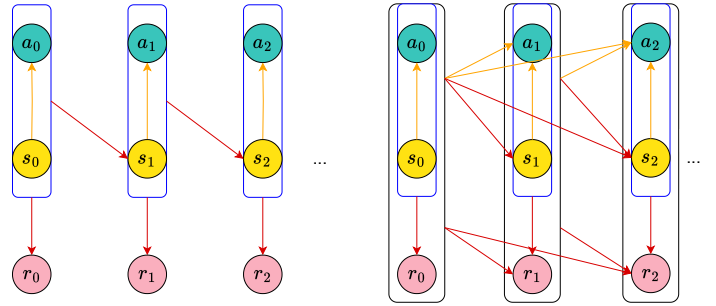


Figure 1: Probabilistic graphical model of policy and transitions in a standard MDP (left) and an NMDP (right). red arrows indicate the transition dynamics of the environment, while orange arrows represent the policy. Arrows from a frame indicate probabilistic dependencies between all the framed variables and the variable at the head of the arrow.

Figure.1 illustrates the probabilistic dependencies between states, actions, and rewards in the definitions of MDP and NMDP. Note that, by definition, the transition dynamics of MDPs and NMDPs return a joint distribution over the next state and reward. For clarity, Figure 1 simplifies this representation to emphasize the main conditional independence relationships.

As illustrated, the key difference between MDP and NMDP lies in the factors determining state transitions and rewards. In an MDP, both the transition to the next state and the received reward rely solely on the current state. For example, in a simple robotic movement model, the robot's next position and the reward it gets are only determined by its current location. Conversely, in an NMDP, the transition to the next state and the associated reward are influenced by previous states, actions, and rewards. This means that historical information plays a crucial role in the decision-making process.

A typical example of an NMDP is stock price prediction in the financial market. Consider a particular stock. The "state"

here can be defined by various factors such as the current stock price, trading volume, and relevant market indices. The "action" could be decisions made by investors, like buying, selling, or holding the stock. The future stock price (the next state) is not simply determined by the current stock price (the current state) and the current investment actions (the current action). Instead, it is highly correlated with historical data. For instance, if a stock has shown a consistent upward trend over the past few quarters due to strong company fundamentals and positive market sentiment, this historical pattern is a significant factor in predicting the stock price in the next period. Also, past events such as quarterly earnings announcements, major product launches, or changes in industry regulations can all leave an imprint on the stock's price movement history and have a substantial impact on future price predictions.

13.2 The Category-Theoretic Outlook on MDP and NMDP

- A **category** is composed of a collection of objects (items) and a collection of morphisms (also known as maps or arrows). Each morphism has a domain and a codomain, that is, each arrow has a source and a target.
- **Objects** in a category can be anything. For example, all sets form the category of sets, all groups form the category of groups, and all topological spaces form the category of topological spaces.
- **Morphisms** represent the connections between pairs of objects. These connections take different forms in different categories. In the category of sets, it is a function; in the category of partially ordered sets, it is a partial order relation; in a category constructed from a set, it is an equivalence relation; in the category of groups, it is a homomorphism; in the category of linear spaces, it is a linear map; or it can be any connection that conforms to the essential characteristics of the objects.

In the MDP category \mathfrak{M} , a morphism $\phi = (\phi_S, \phi_A, \phi_R)$ from \mathcal{M} to \mathcal{M}' bears the following intuitive meaning: Map the states, actions, and rewards of \mathcal{M} separately using (ϕ_S, ϕ_A, ϕ_R) . Thus, we obtain the MDP \mathcal{M}' , whose states, actions, and rewards are images under ϕ and the transition dynamics (along with the reward mechanics) are identical to those of \mathcal{M} , except for the different representations of states, rewards, and actions. The intuitive explanation of morphisms in the NMDP category \mathfrak{N} is similar to that in the MDP category \mathfrak{M} .

- **Functors**, on the other hand, represent the connections between pairs of categories. A functor preserves the structure of each category, mapping objects to objects and morphisms to morphisms while maintaining the structure unchanged. For example, the fundamental group functor maps a topological space to a group. Since it preserves the structure of the topological space, the study of the topological space using geometric

methods is transformed into the study of the fundamental group using algebraic methods.

- Both **category isomorphism** and **category equivalence** capture a sense of "similarity" between categories, but they diverge in key aspects. Category isomorphism represents a special and highly restrictive form of categorical equivalence. Isomorphic categories are, by default, equivalent. They both signify a profound structural and property-based equivalence between categories. Category equivalence, on the other hand, offers more flexibility. It amounts to an "approximate one-to-one correspondence" and is realized through the interaction of two functors. While it preserves many crucial properties, it does so less rigidly than category isomorphism. Take, for example, the category of finite-dimensional vector spaces and the category of matrices. Finite-dimensional vector spaces can be related to matrices, with linear transformations corresponding to matrix multiplications. Although this is not a one-to-one mapping, the two categories are equivalent.

The equivalence between MDP category \mathfrak{M} and NMDP category \mathfrak{N} implies the following: For any MDP, the non-Markov embedding functor \mathbf{N} designates an NMDP. In the NMDP category, this NMDP can maintain the structure created by the morphisms related to the original MDP in the MDP category. Conversely, for any NMDP, the Markov abstraction functor \mathbf{M} determines an MDP. In the MDP category, this MDP can preserve the structure formed by the morphisms associated with the original NMDP in the NMDP category. Notably, the non-Markov embedding functor \mathbf{N} is not the sole means of choosing an NMDP for an MDP, and the Markov abstraction functor \mathbf{M} isn't the only way to select an MDP for an NMDP either.

13.3 The Reversibility Condition of HAS

As the name implies, the History aggregator for State (HAS) constructs the NMDP state by aggregating the MDP history. The reversibility condition of HAS ensures that a decision-making algorithm can reconstruct the original MDP state from the history of the NMDP.

Example 1. We have a non-degenerate MDP $\mathcal{M} = \langle \rho_0, S, A, \{T_t\}_{t=0}^\infty \rangle$, where $S = \mathbb{R}^n$, and a reversible HAS $\mathcal{S}_{S,t} : h_t \mapsto \sum_{\tau=0}^t s_\tau$ together with its reverse:

$$\begin{aligned} \mathcal{S}_{S,t}^*(\{\mathcal{S}_{S,\tau}(h_\tau)\}_{\tau=0}^t) &:= \mathcal{S}_{S,t}(h_t) - \mathcal{S}_{S,t-1}(h_{t-1}) \\ &= s'_t - s'_{t-1} \\ &= \sum_{\tau=0}^t s_\tau - \sum_{\tau=0}^{t-1} s_\tau \\ &= s_t \end{aligned}$$

which recover the MDP state s_t from the NMDP state series $s'_{0:t} = \{\mathcal{S}_{S,\tau}(h_\tau)\}_{\tau=0}^t$.

Applying $\mathcal{S}_{S,t}$ to \mathcal{M} , we get an NMDP $\mathcal{N} = \langle \rho'_0, S', A', \{T'_t\}_{t=0}^\infty \rangle$, where $\rho'_0 = \rho_0$, $S' = S$, $A' = A$,

$$(T'_t(h'_t, a_t))(s'_{t+1}, r_t) = (T_t(s_t, a_t))(s_{t+1}, r_t)$$

For any history $h_t = (s_{0:t}, a_{0:t-1}, r_{0:t-1})$ of \mathcal{M} , the corresponding history of \mathcal{N} is

$$\begin{aligned} h'_t &= (\{\mathcal{S}_{S,\tau}(h_\tau)\}_{\tau=0}^t, a_{0:t-1}, r_{0:t-1}) \\ &= (s'_{0:t}, a_{0:t-1}, r_{0:t-1}) \\ &= (\{\sum_{i=0}^{\tau} s_i\}_{\tau=0}^t, a_{0:t-1}, r_{0:t-1}) \end{aligned}$$

The state dependency structure of h'_t is $\{t, t-1\}$ because calculating $T'_t(h'_t, a_t)$ which is equivalent to calculating $T_t(s_t, a_t)$ or s_t only requires $s'_t = \sum_{\tau=0}^t s_\tau$ and $s'_{t-1} = \sum_{\tau=0}^{t-1} s_\tau$.

The reversibility condition is essential for reconstructing an MDP's state from an NMDP's history. When this condition is violated, consider the use of the non-reversible HAS $\mathcal{B}_{S,t} : h_t \mapsto s_t \odot e_1$, where $e_1 = (1, 0, \dots, 0) \in \mathbb{R}^n$, and the operator \odot is defined as

$$\odot : \begin{array}{ccc} \mathbb{R}^n \times \mathbb{R}^n & \rightarrow & \mathbb{R}^n \\ ((x_1, \dots, x_n), (y_1, \dots, y_n)) & \mapsto & (x_1 y_1, \dots, x_n y_n) \end{array}$$

In such a scenario, the non-reversible HAS retains only the first-dimension value of s_t and discards all others. For any decision-making algorithm designed to solve the NMDP, it is impossible to reconstruct the MDP state s_t using the NMDP history h'_t . Although the functor induced by \mathcal{B}_S can be easily implemented as an environment wrapper, it transforms an MDP into a partially-observable NMDP instead of an NMDP. The absence of necessary information thus increases the complexity of the original MDP.

13.4 Reversible HAS Induced by Group Operator

The intuition for the reversible HAS induced by a group operator is derived from the concept of the prefix-sum of a sequence. It can be shown that the prefix-sum of a time-homogeneous first-order Markov chain is a time-homogeneous second-order Markov chain.

Theorem 7. Let $\{X_n\}_{n=0}^\infty$ be a time-homogeneous first-order Markov chain, then $\{S_n\}_{n=0}^\infty := \{\sum_{i=0}^n X_i\}_{n=0}^\infty$ is a time-homogeneous second-order Markov chain.

Proof. Since $\{X_n\}_{n=0}^\infty$ is a time-homogeneous first-order Markov chain, we have

$$\begin{aligned} \mathbb{P}(X_{n+1} = x_{n+1} \mid X_n = x_n, \dots, X_1 = x_1) \\ = \mathbb{P}(X_{n+1} = x_{n+1} \mid X_n = x_n) \end{aligned}$$

Then

$$\begin{aligned} \mathbb{P}(S_{n+1} = s_{n+1} \mid S_n = s_n, \dots, S_1 = s_1) \\ = \mathbb{P}(S_{n+1} - S_n = s_{n+1} - s_n \mid S_n = s_n, \dots, S_1 = s_1) \\ = \mathbb{P}(X_{n+1} = s_{n+1} - s_n \mid S_n = s_n, \dots, S_1 = s_1) \\ = \mathbb{P}(X_{n+1} = s_{n+1} - s_n \mid X_n = s_n - s_{n-1}, \dots, X_1 = s_1) \\ = \mathbb{P}(X_{n+1} = s_{n+1} - s_n \mid X_n = s_n - s_{n-1}) \end{aligned}$$

And

$$\begin{aligned} \mathbb{P}(S_{n+1} = s_{n+1} \mid S_n = s_n, S_{n-1} = s_{n-1}) \\ = \mathbb{P}(S_{n+1} - S_n = s_{n+1} - s_n \mid S_n = s_n, S_{n-1} = s_{n-1}) \\ = \mathbb{P}(X_{n+1} = s_{n+1} - s_n \mid X_n = s_n - s_{n-1}, S_{n-1} = s_{n-1}) \\ = \mathbb{P}(X_{n+1} = s_{n+1} - s_n \mid \{X_n = s_n - s_{n-1}\} \cap \{ \sum_{i=1}^{n-1} X_i = s_{n-1} \}) \\ = \mathbb{P}(X_{n+1} = s_{n+1} - s_n \mid \{X_n = s_n - s_{n-1}\} \cap \{ \sum_{i=1}^{n-1} x_i = s_{n-1} \mid X_{n-1} = x_{n-1}, \dots, X_1 = x_1 \}) \\ = \mathbb{P}(X_{n+1} = s_{n+1} - s_n \mid \{ \sum_{i=1}^{n-1} x_i = s_{n-1} \mid X_n = s_n - s_{n-1}, X_{n-1} = x_{n-1}, \dots, X_1 = x_1 \}) \\ = \mathbb{P}(X_{n+1} = s_{n+1} - s_n \mid X_n = s_n - s_{n-1}) \end{aligned}$$

Therefore we have

$$\begin{aligned} \mathbb{P}(S_{n+1} = s_{n+1} \mid S_n = s_n, \dots, S_1 = s_1) \\ = \mathbb{P}(S_{n+1} = s_{n+1} \mid S_n = s_n, S_{n-1} = s_{n-1}) \end{aligned}$$

which means $\{S_n\}_{n=0}^\infty$ is a time-homogeneous second-order Markov chain. \square

Similarly, it can be shown that the prefix-sum of a time-homogeneous n -th order Markov chain is a time-homogeneous $n+1$ -th order Markov chain. By repeatedly applying the prefix-sum operation to a time-homogeneous first-order Markov chain, we can obtain a series of Markov chains with increasing orders. We generalize the prefix-sum operation to aggregation by using the group operator, with MDPs/NMDPs instead of Markov chains as operands. This leads to the development of the reversible HAS induced by the group operator.

13.5 Reversible HAS Induced by Auxiliary Sequence and Convolution Operator

The convolution operation in signal processing serves as the source of inspiration for reversible HAS induced by auxiliary sequence and convolution operator. The convolution of two sequence $\{x_n\}_{n=0}^\infty$ and $\{y_n\}_{n=0}^\infty$ is

$$\{x_n\}_{n=0}^\infty * \{y_n\}_{n=0}^\infty = \left\{ \sum_{i=0}^n x_i y_{n-i} \right\}_{n=0}^\infty$$

In the above equation, we fix $\{x_n\}_{n=0}^\infty$ as a constant sequence, let $\{y_n\}_{n=0}^\infty$ be the sequence of states in the history of the MDP, and define multiplication and addition appropriately, thus forming the expression form of HAS induced by auxiliary sequence and convolution operator.

14 The General Applicability of HAS

14.1 Reversible HAS Induced by Group Operator

As is proved in Theorem 3, if the group is a free group or $(\mathbb{R}^k, +)$ for some k , then applying functor \mathbf{G} n times ensures that the state dependency structure is $[t-n, t] \cap \mathbb{N}$, which means for all $i \in [t-n, t] \cap \mathbb{N}$, the NMDP state s'_i is essential for the decoding of MDP state s_t from the NMDP history.

Although there are multiple ways to extend S into a group, as can be seen from the proof of Theorem 3, not every approach can ensure that all elements in the above-mentioned state dependency structure $[t-n, t] \cap \mathbb{N}$ play a role in the decoding process. Therefore, using other extension methods may result in the actually obtained state dependency structure being “sparser” compared to the case of extending S to $(\mathbb{R}^k, +)$ for some k .

The reversible HAS induced by group operator can only create state dependency structure in the form of $[t-n, t] \cap \mathbb{N}$, and it fails to provide a method for specifying the “degree” of each item's dependency. This is also a drawback of this approach.

14.2 Reversible HAS Induced by Auxiliary Sequence and Convolution Operator

As proven in Theorem 6, in this case, the state dependency structure is given by $\{t-\tau \mid (\mathbf{w}^{-1})_{0,\tau} \neq 0\}$, and the "degree" of dependency of $t-\tau$ is $(\mathbf{w}^{-1})_{0,\tau}$. Thus, we can assign any zero or non-zero, large or small values to the entries of the upper-triangular matrix \mathbf{w}^{-1} to control the dependency structure and the "degree" of dependency, making it a flexible way of constructing dependency structure into MDPs.

However, the HAS induced by the auxiliary sequence and convolution operator treats all MDP histories of the same length equally, which means that it assigns the same weights to histories of the same length.

14.3 Further Generalization

Within the framework of reversible HAS, the constructions using group operators and auxiliary sequences are merely two conveniently implementable special cases. There are clear theoretical guarantees for their properties and well-defined methods to analyze their dependency structures. However, they are by no means the only way of constructing reversible HAS. We can also consider non-linear sequence processing methods and sequence processing methods that are sensitive to different histories, etc.

When the reversibility condition is abandoned, HAS becomes a general theoretical model for constructing general partially observable NMDPs. We can use methods such as information hiding and introducing randomness to ensure that the resulting partially observable NMDPs possess the desired properties.

In summary, HAS offers a general theoretical model for transforming MDPs into NMDPs. The reversibility condition captures the key to ensuring the essential equivalence between the pre-transformation and post-transformation MDPs. The HAS framework still holds vast exploration potential and promising research prospects in future studies.