Deep reinforcement learning for controlled piecewise deterministic Markov process in cancer treatment follow-up

Alice Cleynen, Benoîte de Saporta, Orlane Rossini, Régis Sabbadin, Meritxell Vinyals

Deep reinforcement learning for controlled piecewise deterministic Markov process in cancer treatment follow-up.

Alice Cleynen [1] & Benoîte de Saporta [2] & Orlane Rossini [3] & Régis Sabbadin [4] & Meritxell Vinyals [5]

[1] *John Curtin School of Medical Research, Australian National University, Canberra, ACT, Australia and IMAG, Univ Montpellier, CNRS, Montpellier, France alice.cleynen@umontpellier.fr*
[2] *IMAG, Univ Montpellier, CNRS, Montpellier, France benoite.de-saporta@umontpellier.fr*
[3] *IMAG, Univ Montpellier, CNRS, Montpellier, France orlane.rossini@umontpellier.fr*
[4] *Univ Toulouse, INRAE-MIAT, Toulouse, France regis.sabbadin@inrae.fr*
[5] *Univ Toulouse, INRAE-MIAT, Toulouse, France meritxell.vinyals@inrae.fr*

Résumé. Les maladies humaines telles que le cancer impliquent un suivi à long terme. Un·e patient·e alterne des phases de rémission et de rechutes. Un biomarqueur est monitoré tout au long du suivi. Sa dynamique est modélisée par un processus de Markov déterministe par morceaux (PDMP) caché et contrôlé. Le PDMP évolue en temps et en espace continus, le processus est observé à travers un bruit et le modèle est partiellement connu,ce qui rend le problème du contrôle particulièrement difficile. À notre connaissance, il n'existe pas de méthode pour contrôler un tel PDMP, c'est-à-dire pour maximiser la vie du·de la patient·e tout en minimisant le coût du traitement et les effets secondaires. Nous considérons des dates discrètes uniquement pour les décisions, transformant ainsi le PDMP contrôlé en un processus de décision markovien partiellement observé (POMDP). L'algorithme deep Q-network (DQN) permet de résoudre le problème de contrôle. Une des limitation de DQN est de ne pas prendre en compte l'historique complet des observations, ce qui est pourtant une caractéristique clé des POMDP. Ce constat nous conduit à traduire le POMDP en un MDP défini sur l'espace des historiques et à appliquer l'algorithme DQN à ce nouveau modèle. Par le biais de simulations, nous comparons les deux méthodes de résolution. Ces analyses visent à éclairer les avantages et les limites de chaque approche dans le contexte du contrôle de PDMP pour une gestion optimale des maladies chroniques.

Mots-clés. Processus markovien déterministe par morceaux, états cachés, processus de décision markovien, contrôle stochastique, apprentissage par renforcement profond, optimisation de traitement

Abstract. Human diseases such as cancer involve long-term follow-up. A patient alternates between phases of remission with relapses. A biomarker is monitored throughout the follow-up. Its dynamic is modelled by a controlled piecewise deterministic Markov process (PDMP). The PDMP evolves in continuous time and space, the process is observed through noise and some of its parameters are unknown, making the control problem especially difficult. To our knowledge, there is no method to control such a PDMP, i.e. to maximize the life of the patient while minimizing the treatment cost and side effects. We consider discrete

dates only for the decisions, thus turning the controlled PDMP into a partially observable Markov decision process (POMDP). The deep Q-network (DQN) algorithm solves the control problem. A constraint associated with DQN is its inability to consider the entire historical sequence of observations, a crucial aspect in the context of POMDPs. This drawback led us to translate the POMDP into an MDP defined on the space of histories and to apply the DQN algorithm to this new model. Through simulation, we compare the two resolution methods. These analyses aim to shed light on the advantages and limitations of each approach in the context of POMDP control for optimal chronic disease management.

Keywords. Piecewise deterministic Markov process, hidden state, Markov decision process, stochastic control, deep reinforcement learning, treatment optimisation

1 Introduction

Numerous challenges can be characterized as problems of sequential decision-making under uncertainty, including medical treatment design [Wu+23]. In the field of medical decision-making, the treatment of cancer patients emerges as an intricate challenge. Physicians aim to adapt treatments to uphold the patient's quality of life and life expectancy over time. The primary objective is to formulate optimal strategies for cancer treatment follow-up, acknowledging the continuous nature of the patient's state and its partial observability.

Our focus centres on the computational resolution of a specific category of impulse control problems for piecewise deterministic Markov processes (PDMPs). Impulse control for PDMPs involves selecting actions and intervention dates, as initially explored in [CD89]. Approximating solutions to continuous-time and continuous-state impulse control problems, when the process is only partially observed, jump times remain hidden and the underlying model is partially unknown, presents a challenge. Previous approaches [CS18; CS23], propose to express the controlled PDMP into a partially observable Markov decision process (POMDP). Then they resort to discretizing the state space and employing dynamic programming to approximate the value function effectively addressing problems of continuous state space and partial observability. While effective, these methods are constrained by their reliance on explicit model knowledge and the discretization process. An alternative strategy [Cle+24], adopts a simulation-based approach similar to the partially observable Monte-Carlo planning (POMCP) algorithm [SV10]. This method was essentially developed to deal with the continuous state problem. However, it does not require explicit model information.

In this paper, we propose a resolution method leveraging neural networks. While offering generalization capabilities, our approach aims to approximate the value function directly from a simulator of data. As in previous work, we transform the controlled continuous-time PDMP problem into a discrete-time POMDP. While conventional POMDP solutions often operate over history, deep learning methods frequently focus only on current observations. Hence, a compelling direction emerges in adapting POMDPs to Markov decision processes (MDPs) over history. Our primary conjecture is that this paradigm shift will yield enhanced decision-making policies, optimizing cancer treatment strategies.

The paper is organized as follows. In section 2 we state our optimization problem and turn it into a POMDP. In section 3 we give our resolution strategy and our main assumption. Numerical experiments are also described whereas numerical results are postponed to the upcoming conference.

2 Problem statement

In our illustrative medical scenario, a patient enrols in a clinical trial at the onset of a remission phase. Throughout remission, the biomarker hovers at the nominal threshold ζ_0. In the absence of treatment, a relapse triggers an exponential surge in the biomarker level, culminating in the patient's death upon reaching the critical value of D. Treatment interventions succeed in lowering the biomarker level, yet with each relapse, the probability of treatment resistance escalates. This intricate interplay involving phases of remission, relapse, and treatment response constitutes the fundamental essence of our impulse control problem. Our investigation starts with delineating a specialized class of impulse control problems designed for piecewise deterministic Markov processes (PDMPs). We describe the translation of our control problem into a partially observable Markov decision process (POMDP) framework.

2.1 PDMP

We consider an impulse control problem for hidden piecewise deterministic Markov processes (PDMPs.) We introduce four variables m, k, ζ, u where the mode (m, k) corresponds to the patient's overall state of health ($m = 0$: remission, $m = 1$: relapse, $m = 2$: untreatable relapse, $m = 3$: death) and $k \in \mathbb{N}$ (the number of curable relapses). The biological marker level is denoted by $\zeta \in [\zeta_0, D]$ with ζ_0 the nominal value and D the death level and $u \in [0, H]$ is the sojourn time in a health' state (added for technical reasons to deal with semi-Markov condition), where H corresponds to the end of the patient's follow-up. The complete state of the patient is denoted by $x = (m, k, \zeta, u)$ in E the state space. Let the state space E be an open subset of $\mathbb{R}^4$ such that : $E \subset \{0, 1, 2\} \times \mathbb{N} \times [\zeta_0, D] \times [0, H] \cup \{3\}$.

Decisions are made throughout a patient's trajectory. Let $\mathbb{D}$ be the space of decisions such that $\mathbb{D} = \mathcal{L} \times \mathcal{R} \cup \{\Delta\}$. Control is expressed as a decision pair: $d = (\ell, r)$, where $r \in \mathcal{R} = \{15, 30, 60\}$ is the delay before the next visit. Visits correspond to the measurement of the biomarker level and the adjustment of the treatment according to results. The therapeutic choice is $\ell \in \mathcal{L} = \{\emptyset, a, b\}$ ($\ell = \emptyset$: *no treatment*, $\ell = a$: *chemotherapy* and $\ell = b$: *palliative care*). The decision $d = \Delta$ corresponds to the action *do nothing* and applies when the patient is dead.

A PDMP on the state space E is defined by three local characteristics $(\mathbf{\Phi}, \lambda, \mathcal{Q})$. The flow $\mathbf{\Phi}$ describes the deterministic trajectory of the process between jumps. The jump intensity λ characterizes the frequency of jumps. The Markov kernel Q provides a probabilistic mapping from the pre-jump state to the post-jump state.

The flow depends on the control applied and in particular on the treatment: $\mathbf{\Phi}^{\ell}(x,t) = (m, k, \Phi^{\ell}_{m,k}(\zeta,t), u+t)$, where $\Phi^{\ell}_{m,k}(\zeta,t)$ describes only the trajectory of the biological marker between jumps. When the patient is dead, no treatment is applied and the flow is $\mathbf{\Phi}^{\Delta}(x,t) = (m)$. The biomarker evolution (summarized in Table 1) depends on the therapy choice, the disease regimen and the number of relapses.

Let $t^{\ell\star}(x)$ be the deterministic time the flow takes to reach the boundary of the state space E. Let $\partial E = \{1,2\} \times \mathbb{N} \times \{\zeta_0, D\} \times (0, H]$ be the boundary on E. The time $t^{\ell\star}(x)$ also depends on the treatment and the disease regimen: $t^{\ell\star}_{m,k}(\zeta) = \inf\{t > 0 : \Phi^{\ell}_{m,k}(\zeta,t) \in \partial E\}$. This function is detailed in table 2.

m/ℓ	$\emptyset$	a	b
0	ζ_0		
1	$\zeta e^{v_1 t}$	$\zeta e^{-\frac{v_1}{k}t}$	$\zeta e^{v_1 t}$
2	$\zeta e^{v_2 t}$		

Table 1: **Flow.** $\Phi^{\ell}_{m,k}(\zeta,t)$, where v_1 and v_2 are constants.

m/ℓ	$\emptyset$	a	b
0	$+\infty$		
1	$\frac{1}{v_1}\log(\frac{D}{\zeta})$	$\frac{k}{v_1}\log(\frac{\zeta}{\zeta_0})$	$\frac{1}{v_1}\log(\frac{D}{\zeta})$
2	$\frac{1}{v_2}\log(\frac{D}{\zeta})$		

Table 2: **Boundary jump.** $t^{\ell\star}_{m,k}(\zeta)$,, where v_1 and v_2 are constants.

Treatment also influences the risk function $\lambda^{\ell}(x) = \lambda^{\ell}_{m,k}(\zeta,u)$. Notably, there are two distinctive types of relapse scenarios considered: standard relapses occurring during remission phases and relapses indicative of therapeutic escape. For standard relapses, the probability of occurrence increases with the duration of time spent in remission. On the other hand, the risk of relapses associated with therapeutic escape is influenced by the biomarker level. In light of these considerations, we choose Weibull distributions of the form: $\mu_i(u) = (\alpha_i u)^{\beta_i}$ and $\mu'_2(\zeta) = (\alpha'\zeta)^{\beta'}$. Details of jump intensity are available in table 3.

m / ℓ	$\emptyset$	a	b
0	$(\mu_1+\mu_2)(u)$	$\mu_2(u)$	$(\mu_1+\mu_2)(u)$
1	$\mu'_2(\zeta)$		
2	0		
3	0		

Table 3: **Jump intensity.** $\lambda^{\ell}_{m,k}(x)$

In remission, the patient may transition to either a curable relapse in the absence of chemotherapy or an incurable relapse. In the case of relapse and without treatment, the biomarker increases to a critical value D, leading to the patient's death. When chemotherapy is administered, the biomarker decreases to ζ_0 and returns to remission. Regardless of treatment chosen, therapeutic escape may occur at any time. In the case of therapeutic escape, the biomarker increases, regardless of the administered treatment, toward the D threshold, ultimately resulting in the patient's death. We define the Markov kernel $\mathbf{Q}(x,\ell)(x')$ in table 4, for all $h : E \to \mathbb{R}$ a bounded measurable test function. Case $m = 3$ is omitted as no jumps are allowed when patients are dead.

Let $\mathcal{P}(x,d)(x')$ be the transition kernel associated with the continuous time PDMP, for a time period r such that $\mathcal{P}h(x,d) = \mathbb{E}[h(X_r)|X_0 = x, d = (\ell, r)]$. The transition kernel of the PDMP combines the deterministic flow, the jump intensity and the Markov kernel. However,

	$\ell \in \{\emptyset, b\}$
$m=0$	$h(1,k+1,\zeta_0,0)\frac{\mu_1(u)}{(\mu_1+\mu_2)(u)} + h(2,k,\zeta_0,0)\frac{\mu_2(u)}{\mu_1+\mu_2)(u)}$
$m=1$	$h(2,k,\zeta,0)\mathbb{1}_{D>\zeta} + h(3)\mathbb{1}_{\zeta=D}$
$m=2$	$h(3)\mathbb{1}_{\zeta=D}$
	$\ell = a$
$m=0$	$h(2,k,\zeta_0,0)$
$m=1$	$h(2,k,\zeta,0)\mathbb{1}_{\zeta>\zeta_0} + h(0,k,\zeta_0,0)\mathbb{1}_{\zeta=\zeta_0}$
$m=2$	$h(3)\mathbb{1}_{\zeta=D}$

Table 4: **Markov kernel.** $\mathbf{Q}(x,\ell)(x')$

due to its extensive nature, detailed analytic formulas will not be included in this paper, but it is worth noting that they allow the kernel to be simulated easily.

2.2 Partially observed Markov decision process

The trajectory of the process defined above depends on the sequence of decisions and the dates on which the decisions are made. The visit dates take place at discrete dates $n_0 = 0, n_1, ..., n_k$, where the time lapse between two visits can be 15, 30 or 60 days. At most, $N = \frac{H}{15}$ visits can occur. The impulse control problem described above can be formalized as a discrete-time partially observed Markov decision process (POMDP).

A POMDP is a tuple $(\mathbb{S}, \Omega, \mathbb{D}, \mathbb{K}, \mathcal{T}, C)$, where $\mathbb{S}$ corresponds to the state space, which corresponds to the PDMP state space E, Ω corresponds to the observation space, $\mathbb{D}$ to the decision space, which remains unchanged, $\mathbb{K}(\omega) \subseteq \Omega \times \mathbb{D}$ is the space of admissible decisions in observation ω, $\mathcal{T}(s,\omega,d)(s',\omega')$ is the transition kernel of a state-observation tuple $(s,\omega) \in \mathbb{S}\times\Omega$ to state-observation tuple $(s',\omega') \in \mathbb{S}\times\Omega$ when action $d \in \mathbb{K}(\omega)$ is taken, $c(s,d)$ is the cost incurred in state $s \in S$ when decision $d \in D$ is made.

Blood measurements are intrinsically subject to variations independent of the medical condition. These fluctuations can be attributed to measurement errors, natural variations, and external influences. The biomarker is thus observed through a multiplicative noise as the biomarker is growing exponentially. Let $y = \zeta e^{\epsilon}$ with $\epsilon \sim \mathcal{N}(0,1)$ be the noisy biomarker. In addition, the patient's overall health is not observed, except when the patient is deceased. Let $z = \mathbb{1}_{(m=3)}$ be the death indicator. Decision-related constraints then appear. The last visit must take place at the end H of the follow-up. The variable $t \in [0,H]$ indicates the time elapsed since the start of the trajectory. In addition, treatment must be applied for a minimum of 45 days. The variable $\tau \in [0,H]$ corresponds to the time since treatment (chemotherapy or palliative care) was administered. At a given time t, the observation of a patient's condition is $\omega = (\tau, t, y, z)$ with $\omega \in \Omega$. The observation space is $\Omega \subset [0,H]^2 \times \mathbb{R}_+ \times \{0\} \cup [0,H] \times \{1\}$.

Let $\mathbb{K} \subseteq \Omega \times \mathbb{D}$, be the constraint space. It is used to specify all allowed actions state by state: $\mathbb{K}(\omega) = \{d \in \mathbb{D}; (\omega,d) \in \mathbb{K}\} \neq \emptyset$. Constraints are only defined by observations.

$$\mathbb{K}(\omega) = \begin{cases} \{\Delta\} & \text{if } z = 1 \text{ or } t = H \\ (l, r) \in \{a, b\} \times \mathcal{R} & \text{if } 0 < \tau < 45 \text{ and } t + r \leq H \\ (l, r) \in \mathcal{L} \times \mathcal{R} & \text{such that } t + r \leq H \end{cases}$$

The POMDP joint transition-observation function can be expressed as a function of $\mathcal{P}(x, d)(x')$ the piecewise deterministic Markov process (PDMP) transition kernel. For all $g : \mathbb{S} \times \Omega \to \mathbb{R}$ be a bounded measurable test function, $\mathbb{1}_z(m, k, \zeta, u, \tau, t, y, z) = \mathbb{1}_{z=1}$ and f_ϵ is the probability density function of ϵ . Let $\mathcal{T}g(s, \omega, d) = \mathbb{E}[g(S_{t+r}, \omega_{t+r})|S_t = (m, k, \zeta, u), \omega_t = (\tau, t, y, z), d]$.

$$\mathcal{T}g(s, \omega, d) = \begin{cases} g(3, H, 1) & \text{if } d = \Delta \\ \mathcal{P}g(m, k, \zeta, u, 0, t + r, \zeta\epsilon, 0) & \text{if } \ell = \emptyset \\ \mathcal{P}\mathbb{1}_{z=1}g(m, k, \zeta, u, H, 1) + \int \mathcal{P}\mathbb{1}_{z\neq 1}g(m, k, \zeta, u, \tau + r, t + r, \zeta e^{f_\epsilon(\xi)}, 0)\mathrm{d}\xi & \text{else} \end{cases}$$

Let C be the non-negative cost-per-stage function such that $C : \mathbb{D} \times \mathbb{S} \to \mathbb{R}_+$. In POMDPs, the cost function quantifies the cost associated with different decisions per stage.

A history is a sequence of observations and decisions $h_n = \{\omega_0, d_0, \omega_1, \cdots, \omega_n\}$ and H is the set of histories. Along a trajectory, the agent applies decision rules which map a history to an appropriate decision. Let $f_k : H_k \to \mathbb{K}(\omega_k)$ be a decision rule for the kth visit. We define an admissible policy π as a sequence of decision rules $\pi = (f_k)_{0:N-1}$ and Π the set of all admissible policies. Then, the total cumulated cost from visit k is defined as follows $C_k = \sum_{n=k}^{N-1} C(D_n, S_{n+1})$ for all $h \in H$.

The value function $V^\pi(h_k) = \mathbb{E}_\pi[C_k|h = h_k]$ is the expected return from history h when following policy π. Our next aim is to obtain an optimal policy $\pi^\star$ such that the value function V is optimal: $V^\star(h) = \min_{\pi\in\Pi} V^\pi(h)$ for all $h \in H$.

3 Resolution strategy

In the next section, we proceed to an exploration of the deep Q-network (DQN) algorithm. We then move on to the translation of the partially observable Markov decision process (POMDP) into a Markov decision process (MDP) on the history. This allows us to discuss the two main strategies proposed.

3.1 Deep Q-Network algorithm

Reinforcement learning methodologies can be broadly categorized into two principal approaches: value learning and policy learning. These approaches diverge in their strategies for addressing sequential decision problems. Value learning focuses on assessing and enhancing the value function associated with a given policy, aiming to identify the optimal value for each state. On the other hand, policy learning directly updates the policy, determining the optimal sequence of actions for each state. Notably, policy iteration often achieves convergence in

fewer iterations, yet value iteration assures convergence to the optimal policy. Furthermore, the value iteration approach ensures a deterministic policy, a crucial characteristic in cancer monitoring and treatment.

Deep Q-network (DQN) is a value learning algorithm developed in [Mni+13]. DQN uses a deep neural network to approximate the action-state function Q rather than the value function V, where the Q-function corresponds to the expected return starting from state $s \in \mathbb{S}$, taking the decision $d \in \mathbb{D}$: $Q^{\pi}(h_k, d_k) = \mathbb{E}_{\pi}[C_k | h = h_k, d = d_k]$. The optimisation problem is then $V^{\star}(h) = min_{d \in \mathbb{D}} Q(h, d)$. This choice aims to mitigate the overestimation of Q-values, thereby contributing to faster and more stable learning during training. DQN facilitates optimal decision-making in intricate and dynamic environments by focusing on the Q function.

The double deep Q-network (DDQN), introduced in [HGS16], represents an enhancement of the DQN. In contrast to DQN, DDQN employs two neural networks: a target network and a primary network, as illustrated in Figure 1. The primary network is responsible for action selection, while the target network is utilized to compute target values for the primary network. The loss function calculation depends on the weights θ of each network: $L(\theta) = \left[(c + \gamma \max_{d_{t+1}} Q(\omega_{t+1}, d_{t+1}; \theta^{-})) - Q(\omega, d^{\star}; \theta)\right]^2$, where $\gamma \in [0, 1]$ denotes a discount factor. The training objective is to minimize this loss function to improve the predictive capabilities of the neural network. By segregating action selection and Q-value estimation processes, DDQN mitigates Q-value overestimation, resulting in expedited training, improved learning stability, and more effective policies.

DQN has introduced methodologies like experience replay to enhance network updates and model training [Mni+13]. Experience replay involves the utilization of a replay buffer, as depicted in Figure 1. The replay buffer is a repository of past experiences, storing transitions consisting of state-action pairs, the next state, and their corresponding costs. During iterations, a random batch of experiences is sampled from the buffer, diminishing the inherent correlation in sequential data and breaking temporal dependencies between successive observations. This detachment of experiences contributes to a more resilient and stable training procedure, preventing the algorithm from being overly affected by the immediate consequences of recent actions. Consequently, the incorporation of a replay buffer enhances the stability and efficiency of the learning process.

3.2 Equivalent MDP on the history

The deep Q-network (DQN) algorithm operates within the framework of a Markov decision process (MDP). Our problem is a partially observable Markov decision process (POMDP). To bridge this gap, a necessary transformation is required to convert our POMDP into an MDP defined in the space of histories. Employing the MDP framework on the histories will empower us to base our decisions on the entire trajectory, as opposed to only relying on the last observation. This modification is anticipated to enhance the performance of the DQN algorithm by providing a more comprehensive context for decision-making within our sequential control problem.

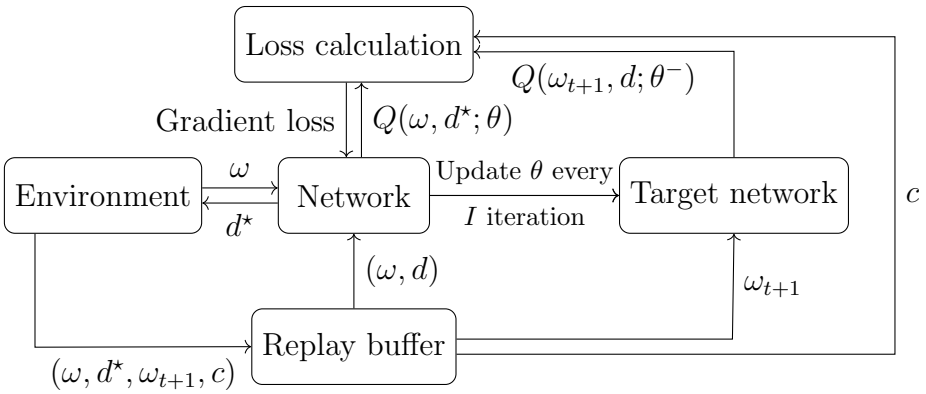

Figure 1: Conceptual diagram of a double deep Q-network. *The primary network estimates the Q-value for a given observation ω. The target network provides an estimate of the Q-value for observation ω_{t+1}, based on the last copy of the weights in the main network. For every I learning iteration, the weights of the main network are updated in the target network.*

Consider the POMDP described in section 2.2 and defined by the tuple $(\mathbb{S}, \mathbb{D}, \mathbb{K}, \mathcal{T}, C)$. Let $\mathcal{B}(s_k, h) = \mathbb{P}(s_k \in S | h_k = h)$ be the belief state, i.e. the probability distribution over states given history $h \in H_k$. It is updated as the agent takes actions and receives observations, allowing it to make decisions based on records of past observations.

Consider the derived MDP with histories as states, defined by the tuple $(H, \mathbb{D}, \mathbb{K}, \tilde{\mathcal{T}}, \tilde{C})$, where $\tilde{\mathcal{T}}(h, d)(h') = \int_{s \in \mathbb{S}} \int_{s' \in \mathbb{S}} g(s') \mathcal{B}(ds, h) \mathcal{T}(s, d)(ds')$ and $\tilde{C}(h, d) = \int_{s \in \mathbb{S}} \mathcal{B}(ds, h) C(s, d)$. The value function $\tilde{V}^\pi(h)$ of the MDP on history is equal to the value function $V^\pi(h)$ of the POMDP, for every $\pi \in \Pi$. The detailed proof is available in [SV10].

3.3 Numerical experiments

In our experimental setup, we aim to investigate and compare the performance of two distinct strategies for solving our sequential decision-making problem. The first scenario involves applying the (double) deep Q-network (DQN) algorithm within the partially observable Markov decision process (POMDP) framework, where the algorithm relies only on the current observation for decision-making, as illustrated in Figure 2. In contrast, the second scenario entails leveraging the DQN algorithm within the Markov decision process (MDP) framework. The algorithm takes into account the entire history of observations when making decisions, as depicted in Figure 3. This comparative analysis will provide insights into the impact of historical information on the algorithm's decision-making process and overall performance.

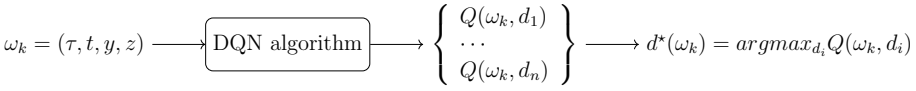

Figure 2: DQN Applied to POMDP

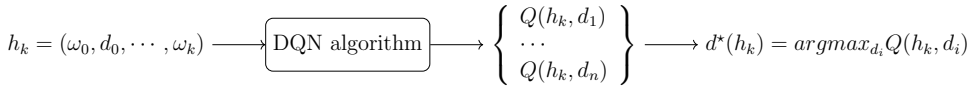

Figure 3: DQN Applied to MDP on history

The efficacy of decision-making policies is evaluated based on their cost implications, with superior policies invariably associated with lower costs. We expect that DQN within the MDP framework will outperform. This hypothesis is grounded in the idea that a richer context, encapsulating the entire history of interactions, can lead to more informed decision-making. The results of our comparative analysis will be presented and discussed at the upcoming conference.

4 Conclusion

In conclusion, the monitoring of cancer treatment in patients can be modelled by a hidden controlled piecewise deterministic semi-Markov process (PDsMP). The formalism of this process is complex and does not allow for its direct resolution. For this reason, its transformation into an equivalent partially observable Markov decision process (POMDP) is essential. Typically, POMDPs are solved over the space of histories, yet deep learning methods in RLlib often focus only on observations. By translating the POMDP into a Markov decision process (MDP) over histories we can use deep Q-networks (DQN) to account for the entire historical context. Our underlying hypothesis posits that this approach will yield a more effective policy. The outcomes of this exploration will be presented on the day of the conference.

The exploration of alternative modelling avenues remains a compelling direction for future research. Instead of exclusively adopting the MDP on history, an intriguing avenue could involve transitioning towards an MDP formulated on belief states. This shift could offer a more nuanced representation of uncertainty and enhance decision-making capabilities. Additionally, while our present study focuses on translating a POMDP into an MDP on historical states for integration with DQN, it is crucial to acknowledge existing methods utilizing Recurrent Neural Networks (RNNs) directly on historical sequences [HS15; Kap+18]. Finally, we hypothesize that a more informative framework leads to more efficient decision-making. Consequently, exploring model-based approaches, particularly Bayesian model-based methods for learning the model, presents a promising avenue for future investigations.

References

[CD89] O. L. V. Costa and M. H. A. Davis. "Impulse control of piecewise-deterministic processes". en. In: *Mathematics of Control, Signals, and Systems* 2.3 (1989), pp. 187–206. DOI: 10.1007/BF02551384.

[Cle+24] Alice Cleynen et al. "Medical follow-up optimization: A Monte-Carlo planning strategy". In: *arXiv* (2024). DOI: 10.48550/arXiv.2401.03972.

[CS18] Alice Cleynen and Benoıte de Saporta. "Change-point detection for piecewise deterministic Markov processes". In: *Automatica* 97 (Nov. 2018), pp. 234–247. DOI: 10.1016/j.automatica.2018.08.011.

[CS23] Alice Cleynen and Benoıte de Saporta. "Numerical method to solve impulse control problems for partially observed piecewise deterministic Markov processes". In: *arXiv* (2023). DOI: 10.48550/arXiv.2112.09408.

[HGS16] Hado van Hasselt, Arthur Guez, and David Silver. "Deep reinforcement learning with double Q-Learning". In: *AAAI'16: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. 2016, pp. 2094–2100. DOI: 10.5555/3016100.3016191.

[HS15] Matthew Hausknecht and Peter Stone. "Deep Recurrent Q-Learning for Partially Observable MDPs". In: *Papers from the 2015 AAAI Fall Symposium* (July 2015). DOI: 10.48550/arXiv.1507.06527. eprint: 1507.06527.

[Kap+18] Steven Kapturowski et al. "Recurrent Experience Replay in Distributed Reinforcement Learning". In: *International conference on learning representations* (Sept. 2018).

[Mni+13] Volodymyr Mnih et al. "Playing Atari with Deep Reinforcement Learning". In: (2013). DOI: 10.48550/arXiv.1312.5602.

[SV10] David Silver and Joel Veness. "Monte-Carlo Planning in Large POMDPs". In: *Advances in Neural Information Processing Systems*. Vol. 23. Curran Associates, Inc., 2010.

[Wu+23] XiaoDan Wu et al. "A value-based deep reinforcement learning model with human expertise in optimal treatment of sepsis". en. In: *npj Digital Medicine* 6.1 (2023), pp. 1–12. DOI: 10.1038/s41746-023-00755-5.