Deep Reinforcement Learning for Bayes-Adaptive Impulse Control of PDMPs

Orlane Rossini [1], Alice Cleynen [1,2], Benoîte de Saporta [1],
Régis Sabbadin [3] and Meritxell Vinyals [3]

[1]IMAG, Univ Montpellier, CNRS, Montpellier, France
[2]John Curtin School of Medical Research, The Australian National University, Canberra, ACT, Australia
[3]Univ Toulouse, INRAE-MIAT, Toulouse, France

November 2025

Medical context

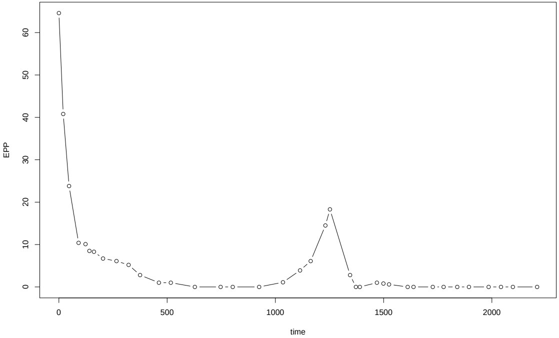


FIGURE: Example of patient data[a]

- Patients who have had cancer benefit from regular follow-up;
- The concentration of clonal immunoglobulin is measured over time;
- The doctor has to make new decisions at each visit.

[a] IUCT Oncopole and CRCT, Toulouse, France

Medical context

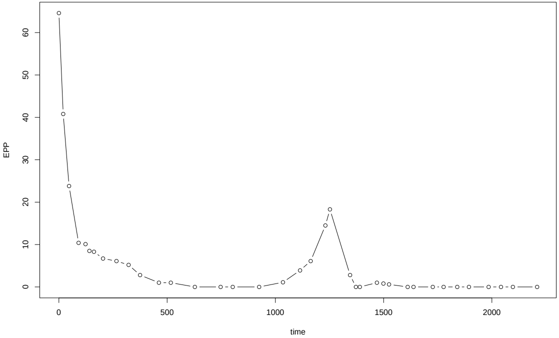


FIGURE: Example of patient data[a]

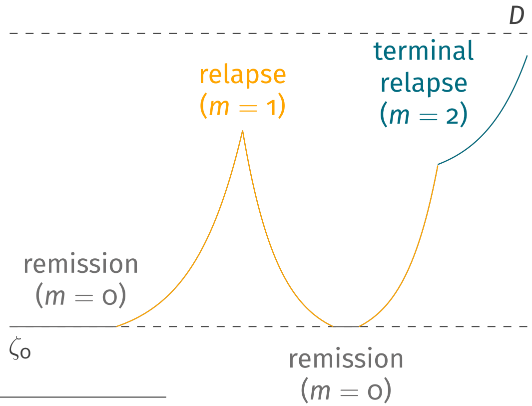
- Patients who have had cancer benefit from regular follow-up;
- The concentration of clonal immunoglobulin is measured over time;
- The doctor has to make new decisions at each visit.

$\Longrightarrow$ **Optimising decision-making to ensure the patient's quality of life**

[a] IUCT Oncopole and CRCT, Toulouse, France

Controlled PDMP[1]

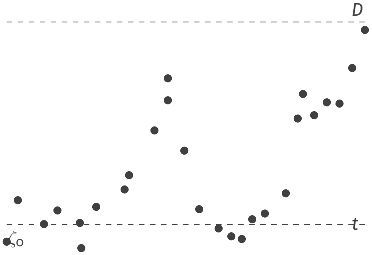
We switch randomly from one deterministic regime to another.



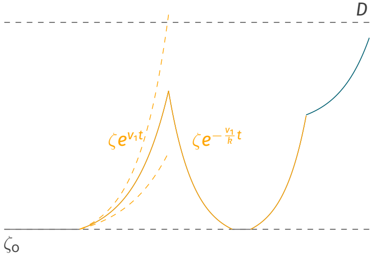
[1] Piecewise Deterministic Markov Processes

Difficulties

Partial observation



Partially known dynamics



Hypothesis: $v_1 \sim$ Log-Normal (μ, σ^{-2}), with μ and σ unknown.

Methods

Simplified real-life problem

continuous time
continuous state space
partially observed
partially known dynamics
non simulable

Controlled PDMP[2]

Modelling

Deep Reinforcement Learning

Data consuming

[2] Piecewise Deterministic Markov Processes

Methods

Simplified real-life problem

Controlled PDMP

BAPOMDP[3]

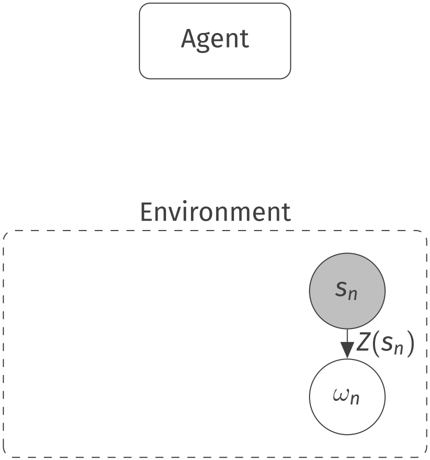
Modelling

discrete observation dates
continuous state space
partially observed
known dynamics
simulable

Deep Reinforcement Learning

[3] Bayes-Adaptive Partially Observed Markov Decision Process

Characteristics of a POMDP[4]



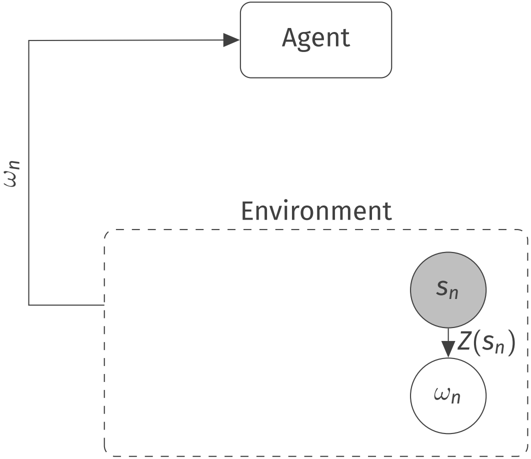
POMDP Definition

A POMDP is defined by a tuple $(\mathbb{S}, \mathbb{A}, P, \Omega, Z, c)$.

- Patient condition $s = (m, k, \zeta, u) \in \mathbb{S}$;
- Actions $a = (\ell, r) \in \mathbb{A}$;
- Transition function $P(s'|s, a)$;
- **Observation** $\omega = (k, F(\zeta, \epsilon), \mathbb{1}_{m=3}) \in \Omega$;
- **Observation function** $Z(\omega|s)$;
- Cost function $c : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \to \mathbb{R}$.

[4] Partially Observed Markov Decision Process

Characteristics of a POMDP[4]



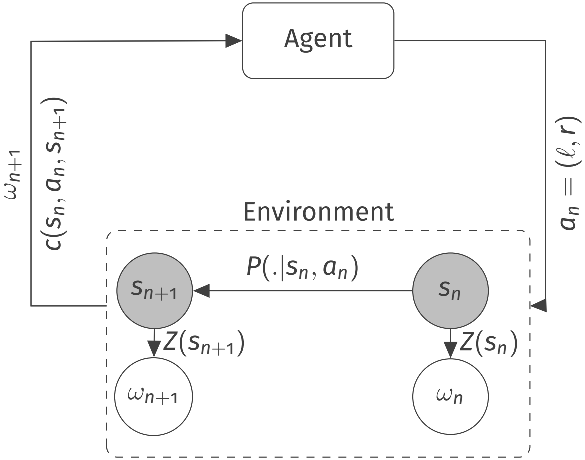
POMDP Definition

A POMDP is defined by a tuple $(\mathbb{S}, \mathbb{A}, P, \Omega, Z, c)$.

- Patient condition $s = (m, k, \zeta, u) \in \mathbb{S}$;
- Actions $a = (\ell, r) \in \mathbb{A}$;
- Transition function $P(s'|s, a)$;
- **Observation** $\omega = (k, F(\zeta, \epsilon), \mathbb{1}_{m=3}) \in \Omega$;
- **Observation function** $Z(\omega|s)$;
- Cost function $c : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \to \mathbb{R}$.

[4] Partially Observed Markov Decision Process

Characteristics of a POMDP[4]



POMDP Definition

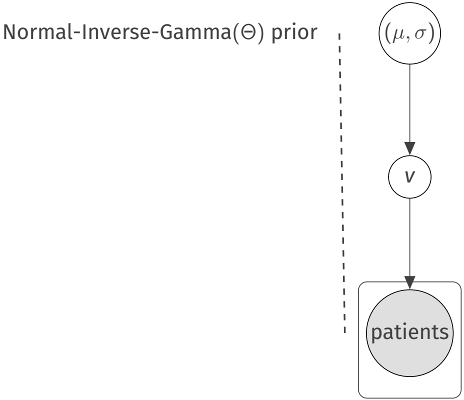
A POMDP is defined by a tuple $(\mathbb{S}, \mathbb{A}, P, \Omega, Z, c)$.

- Patient condition $s = (m, k, \zeta, u) \in \mathbb{S}$;
- Actions $a = (\ell, r) \in \mathbb{A}$;
- Transition function $P(s'|s, a)$;
- **Observation** $\omega = (k, F(\zeta, \epsilon), \mathbb{1}_{m=3}) \in \Omega$**;**
- **Observation function** $Z(\omega|s)$**;**
- Cost function $c : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \to \mathbb{R}$.

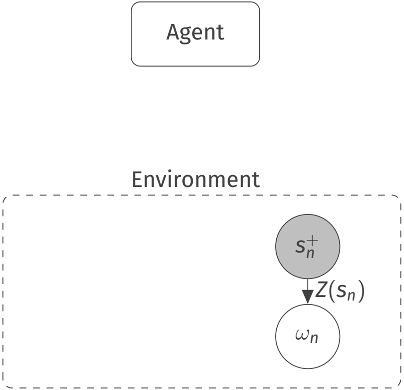
The transition function $P(s'|s, a)$ is a combination of PDMP local characteristics.

[4]Partially Observed Markov Decision Process

Handle uncertainty with Bayesian framework



Characteristics of a BAPOMDP[5]



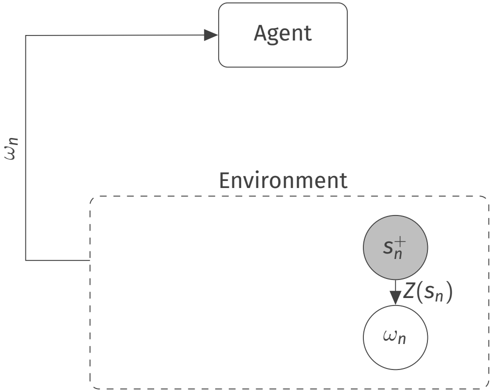
BAPOMDP Definition

Un BAPOMDP se définit par un tuple $(\mathbb{S}^+, \mathbb{A}, P^+, \Omega, Z, c)$.

- **Space of hyperstate** $\mathbb{S}^+ = \mathbb{S} \times \Theta$**;**
- Actions $a = (\ell, r) \in \mathbb{A}$;
- **Transition function** $P^+(s', \theta' | s, a, \theta)$**;**
- Observation $\omega = (k, F(\zeta, \epsilon), \mathbb{1}_{m=3}) \in \Omega$;
- Observation function $Z(\omega|s)$;
- Cost function $c : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \to \mathbb{R}$.

[5]Bayes Adaptive Partially observed Markov decision process

Characteristics of a BAPOMDP[5]



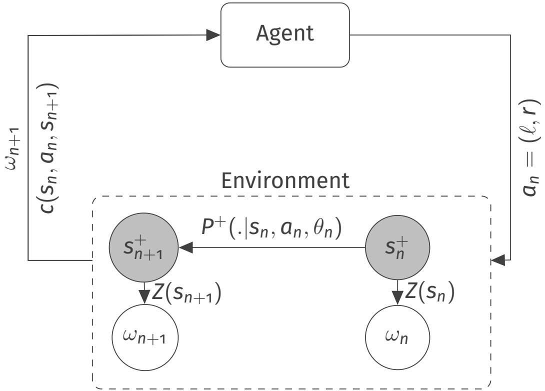
BAPOMDP Definition

Un BAPOMDP se définit par un tuple $(\mathbb{S}^+, \mathbb{A}, P^+, \Omega, Z, c)$.

- **Space of hyperstate** $\mathbb{S}^+ = \mathbb{S} \times \Theta$**;**
- Actions $a = (\ell, r) \in \mathbb{A}$;
- **Transition function** $P^+(s', \theta' | s, a, \theta)$**;**
- Observation $\omega = (k, F(\zeta, \epsilon), \mathbb{1}_{m=3}) \in \Omega$;
- Observation function $Z(\omega | s)$;
- Cost function $c : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \to \mathbb{R}$.

[5] Bayes Adaptive Partially observed Markov decision process

Characteristics of a BAPOMDP[5]



BAPOMDP Definition

Un BAPOMDP se définit par un tuple $(\mathbb{S}^+, \mathbb{A}, P^+, \Omega, Z, c)$.

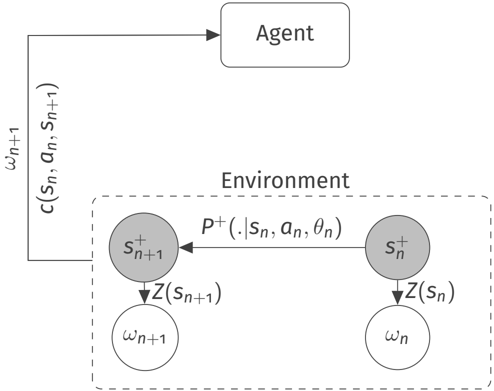
- **Space of hyperstate** $\mathbb{S}^+ = \mathbb{S} \times \Theta$**;**
- Actions $a = (\ell, r) \in \mathbb{A}$;
- **Transition function** $P^+(s', \theta' | s, a, \theta)$**;**
- Observation $\omega = (k, F(\zeta, \epsilon), \mathbb{1}_{m=3}) \in \Omega$;
- Observation function $Z(\omega|s)$;
- Cost function $c : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \to \mathbb{R}$.

$$P^+\big((s', \theta') \in B_E \times B_\Theta \mid (s, \theta), a\big)$$

$$= \int_{B_E} 1_{B_\Theta} \mathcal{U}(\theta, s, a, s') \times P(ds' \mid s, a, \theta).$$

[5] Bayes Adaptive Partially observed Markov decision process

Characteristics of a BAPOMDP[5]



BAPOMDP Definition

Un BAPOMDP se définit par un tuple $(\mathbb{S}^+, \mathbb{A}, P^+, \Omega, Z, c)$.

- **Space of hyperstate** $\mathbb{S}^+ = \mathbb{S} \times \Theta$**;**
- Actions $a = (\ell, r) \in \mathbb{A}$;
- **Transition function** $P^+(s', \theta' | s, a, \theta)$**;**
- Observation $\omega = (k, F(\zeta, \epsilon), \mathbb{1}_{m=3}) \in \Omega$;
- Observation function $Z(\omega | s)$;
- Cost function $c : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \to \mathbb{R}$.

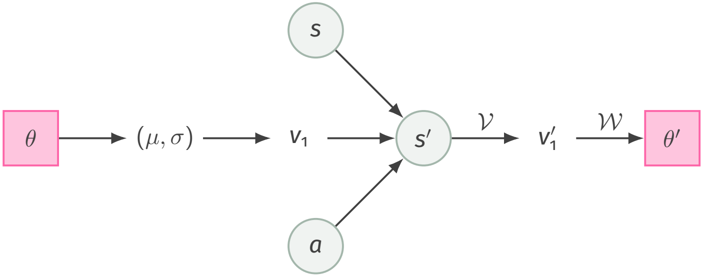
$$P^+\big((s', \theta') \in B_E \times B_\Theta \mid (s, \theta), a\big)$$

$$= \int_{B_E} 1_{B_\Theta} \mathcal{U}(\theta, s, a, s') \times P(ds' \mid s, a, \theta).$$

[5]Bayes Adaptive Partially observed Markov decision process

Generate transition from prior

$$\mathcal{U}(\theta, s, a, s') = \mathcal{W}(\theta, \mathcal{V}(s, a, s')),$$



Solving a BAPOMDP[6]

Identify an optimal policy $\pi^\star$

$$\underbrace{c(s,a,s')}_{\text{Cost function}} = \underbrace{C_V}_{\text{visit cost}} + \underbrace{C_D(H-t') \times \mathbb{1}_{m'=3}}_{\text{death cost}} + \underbrace{\kappa_C \times r \times \mathbb{1}_{\ell=a}}_{\text{treatment cost}}$$

[6] Bayes Adaptative Partially Observable Markov Decision Process

Solving a BAPOMDP[6]

Identify an optimal policy $\pi^\star$

$$\underbrace{V(\pi, s)}_{\text{Optimization criterion}} = \underbrace{\mathbb{E}_s^\pi[\sum_{n=0}^{H-1} c(S_{n-1}, A_n, S_n)]}_{\text{Expected total cost as a result of the policy } \pi}$$

[6] Bayes Adaptative Partially Observable Markov Decision Process

Solving a BAPOMDP[6]

Identify an optimal policy $\pi^\star$

$$\underbrace{V(\pi, s)}_{\text{Optimization criterion}} = \underbrace{\mathbb{E}_s^\pi[\sum_{n=0}^{H-1} c(S_{n-1}, A_n, S_n)]}_{\text{Expected total cost as a result of the policy } \pi}$$

$$\underbrace{V^\star(s)}_{\text{Value function}} = \underbrace{\min_{\pi \in \Pi} V(\pi, s)}_{\text{Minimisation across policy space}}$$

[6] Bayes Adaptative Partially Observable Markov Decision Process

Solving a BAPOMDP[6]

Identify an optimal policy $\pi^\star$

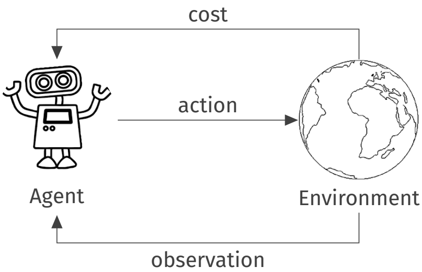
In reality, we do not observe state space!

Let $h_n = (\omega_0, a_0, \omega_1, a_1, \ldots, \omega_n)$ be the history

$$\underbrace{V^\star(h)}_{\text{Value function}} = \underbrace{\min_{\pi \in \Pi} V(\pi, h)}_{\text{Minimisation across policy space.}}$$

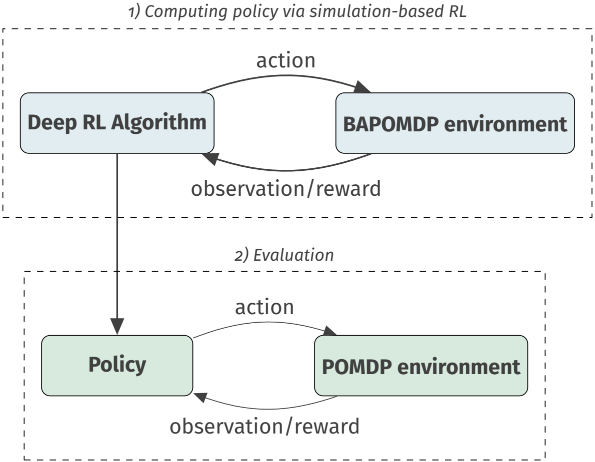
[6]Bayes Adaptative Partially Observable Markov Decision Process

Reinforcement Learning



The optimal policy is obtained from the experiments $< \omega, a, \omega', c >$, generated from P^+ transition function

Evaluate BAPOMDP framework



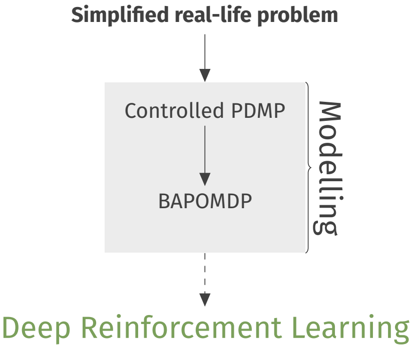
Performance BAPOMDP

Policies	Mean Cost (log)	Survival rate	Treatment number	Visit number
OH	13.45 ± 0.01	99.60%	0.87	120.57
Random	12.79 ± 0.01	92.08%	4.36	67.21
Inactive	8.21 ± 0.07	63.42%	0.00	35.39
Threshold	7.03 ± 0.04	99.98%	5.07	64.67
DQN[7]	5.08 ± 0.00	99.70%	19.99	38.99
PPO[8]	5.94 ± 0.01	99.80%	19.99	58.99

TABLE: Policy evaluation performance on 5000 Monte-Carlo simulations

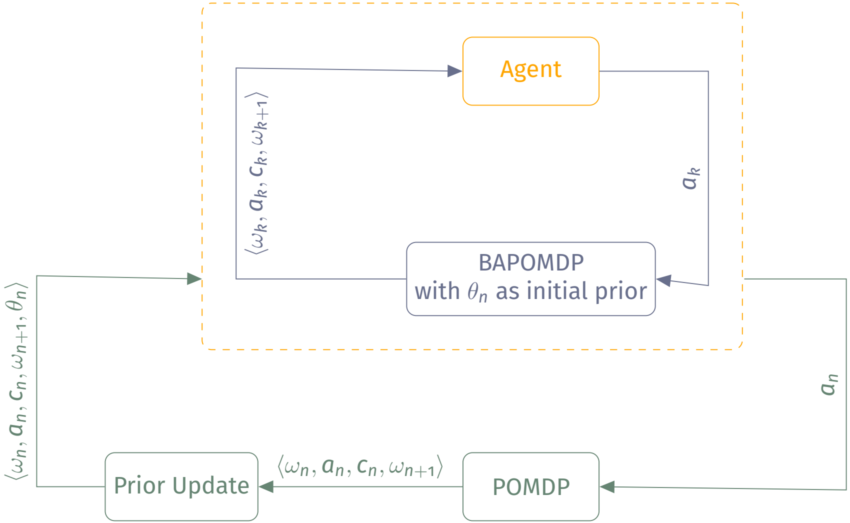
[7] Deep Q-Network
[8] Proximal Policy Optimization

Conclusion



- Bayes-adaptive method to address the PDMP control problem
- No explicit policies
- No estimates of unknown parameters

Future work



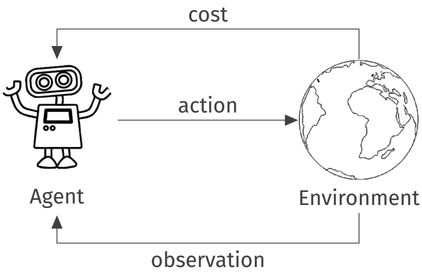
Agent
$\langle \omega_k, a_k, c_k, \omega_{k+1} \rangle$
a_k
BAPOMDP
with θ_n as initial prior
$\langle \omega_n, a_n, c_n, \omega_{n+1}, \theta_n \rangle$
a_n
Prior Update
$\langle \omega_n, a_n, c_n, \omega_{n+1} \rangle$
POMDP

Policy behavior indicators

TABLE: **Summary of policy behavior indicators based on 5 000 Monte-Carlo simulations.**

Indicator	PPO with AM	DQN with AM
Survival rates	$99.80\% \pm 0.00$	$99.70\% \pm 0.00$
Average number of treatment	19.99 ± 0.00	19.99 ± 0.01
Average time spend under treatment	1199.63 ± 00.04	1199.56 ± 0.05
Average number of visit	58.99 ± 0.01	38.99 ± 0.01
Average delay between two visits	40.00 ± 0.00	60.00 ± 0.00
Rate of visits occurring within 15 days	0.01 ± 0.00	0.00 ± 0.00
Rate of visits occurring within 30 days	66.66 ± 0.17	0.00 ± 0.00
Rate of visits occurring within 60 days	33.33 ± 0.17	100 ± 0.00

Reinforcement Learning



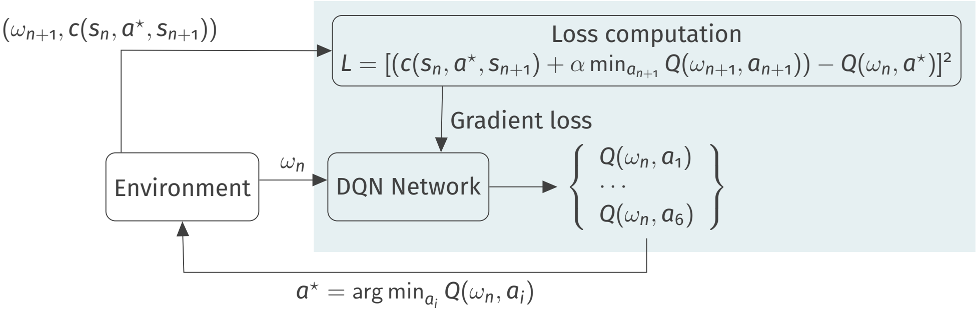
$$\underbrace{Q^{\pi}(s,a)}_{\text{Q value}} = \underbrace{\mathbb{E}^{\pi}[\sum_{n=0}^{H-1} c(S_{n-1}, A_n, S_n) | s, a = (\ell, r)]}_{\text{Value of an action in a state according to the policy } \pi}$$

$$\underbrace{Q^{\star}(s,a)}_{\text{Q function}} = \min_{\pi \in \Pi} Q^{\pi}(s,a)$$

The optimal policy is obtained from the experiments $< \omega, a, \omega', c >$, generated from P^{+} transition function

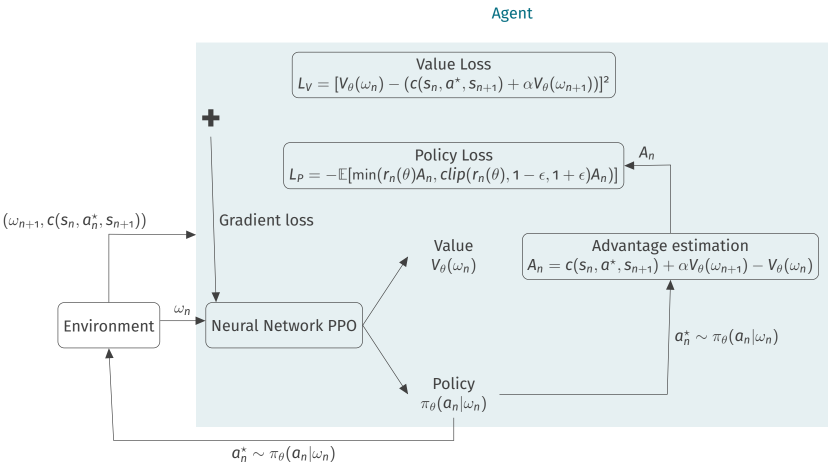
$$\underbrace{A(s,a)}_{\text{Advantage function}} = \underbrace{Q(s,a) - V(s)}_{\text{Extra cost obtained by the agent by taking the action}}$$

Algorithm example: DQN[9]



[9] Deep Q-Network

Algorithm example: PPO[10]



[10] Proximal policy optimization