Non-Stationary Markov Decision Processes
a Worst-Case Approach using Model-Based Reinforcement Learning

Erwan Lecarpentier
Université de Toulouse
ONERA - The French Aerospace Lab
erwan.lecarpentier@isae-supaero.fr

Emmanuel Rachelson
Université de Toulouse
ISAE-SUPAERO
emmanuel.rachelson@isae-supaero.fr

Abstract

We study MDPs evolving over time (Definition 1) and consider planning in this setting. We make two hypotheses:

1. Continuous, bounded, evolution (Definition 2);
2. Snapshot model (Definition 3) known at each decision epoch.

Our contribution can be presented in three points:

1. Proposal of a planning method robust to the environment's evolution;
2. Introduction of a zero-shot model-based algorithm, Risk-Averse Tree-Search (RATS), computing the best worst-case action;
3. Illustration of the benefits of the approach in experiments.

Non-Stationary Markov Decision Processes

Definition (1)

An **NSMDP** is an MDP whose transition and reward functions depend on the decision epoch. It is defined by a 5-tuple $\{\mathcal{S}, \mathcal{T}, \mathcal{A}, (p_t)_{t\in\mathcal{T}}, (r_t)_{t\in\mathcal{T}}\}$ where $\mathcal{S}$ is a state space; $\mathcal{T} \equiv \{1, 2, \dots, N\}$ is the set of decision epochs, $N \leq +\infty$; $\mathcal{A}$ is an action space; $p_t(s' \mid s, a)$ is the probability of reaching state s' with action a at decision epoch t in state s; $r_t(s, a, s')$ is the reward associated to the transition from s to s' with action a at decision epoch t.

Definition (2)

An (L_p, L_r)-**LC-NSMDP** is an NSMDP whose transition and reward functions are respectively L_p-LC and L_r-LC w.r.t. time, i.e.,

$$\forall (t, \hat{t}, s, s', a) \in \mathcal{T}^2 \times \mathcal{S}^2 \times \mathcal{A}, \begin{cases} W_1(p_t(\cdot \mid s,a), p_{\hat{t}}(\cdot \mid s,a)) & \leq L_p|t-\hat{t}| \\ |r_t(s,a,s') - r_{\hat{t}}(s,a,s')| & \leq L_r|t-\hat{t}|. \end{cases}$$

Risk-Averse Tree-Search algorithm

Definition (3)

The snapshot of an NSMDP $\{\mathcal{S}, \mathcal{T}, \mathcal{A}, (p_t)_{t\in\mathcal{T}}, (r_t)_{t\in\mathcal{T}}\}$ at decision epoch t_0, denoted by MDP_{t_0}, is the stationary MDP defined by the 4-tuple $\{\mathcal{S}, \mathcal{A}, p_{t_0}, r_{t_0}\}$ where $p_{t_0}(s' \mid s, a)$ and $r_{t_0}(s, a, s')$ are the transition and reward functions of the NSMDP at t_0.

Proposition (1)

Set of admissible snapshot models. *Consider an (L_p, L_r)-LC-NSMDP, $s, t, a \in \mathcal{S} \times \mathcal{T} \times \mathcal{A}$. The transition and expected reward functions (p_t, R_t) of the snapshot MDP_t respect*

$$(p_t, R_t) \in \Delta_t := \mathcal{B}_{W_1}(p_{t-1}(\cdot \mid s,a), L_p) \times \mathcal{B}_{|\cdot|}(R_{t-1}(s,a), L_p + L_r)$$

where $\mathcal{B}_d(c, r)$ is the ball of centre c, defined with metric d and radius r.

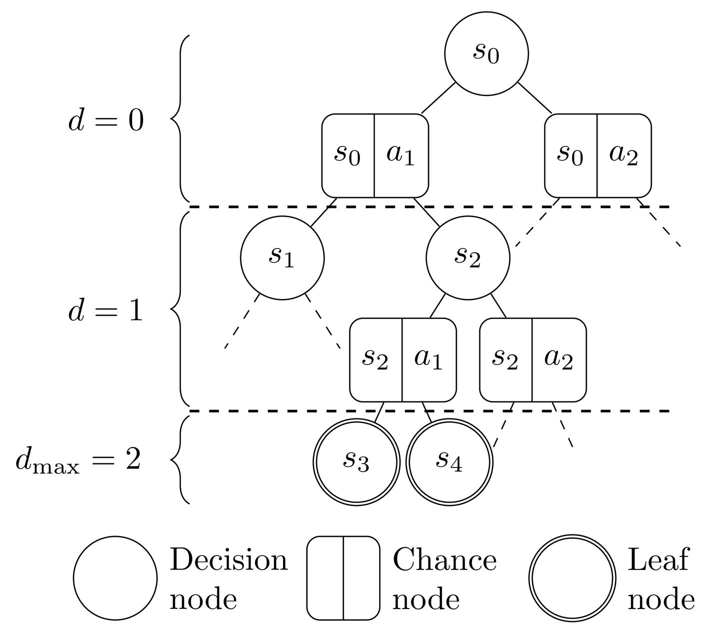

Figure: Tree structure: alternation between **decision nodes** labelled by a unique state and **chance nodes** labelled by a state-action pair. Maximum depth $d_{\max} = 2$, action space $\mathcal{A} = \{a_1, a_2\}$. The tree is entirely developed until $d_{\max}$ which makes the per-time-step complexity of the RATS algorithm $\mathcal{O}(|\mathcal{S}|^{3.5}|\mathcal{A}|^2)^{d_{\max}}$.

Algorithm 1: RATS algorithm

RATS (s_0, t_0, maxDepth)
ν_0 = rootNode(s_0, t_0)
Minimax(ν_0)
$\nu^* = \arg\max_{\nu' \text{ in } \nu.\text{children}} \nu'.\text{value}$
return ν^*.action

Minimax (ν, maxDepth)
if *ν is DecisionNode* **then**
  if *ν.state is terminal* **or** *ν.depth = maxDepth* **then**
    return ν.value = heuristicValue(ν.state)
  else
    return ν.value = $\max_{\nu' \in \nu.\text{children}}$Minimax($\nu'$, maxDepth)
else
  return $\nu.\text{value} = \min_{(p,R) \in \Delta_{t_0}^t} R(\nu) + \gamma \sum_{\nu' \in \nu.\text{children}} p(\nu' \mid \nu)\text{Minimax}(\nu', \text{maxDepth})$

Experiments

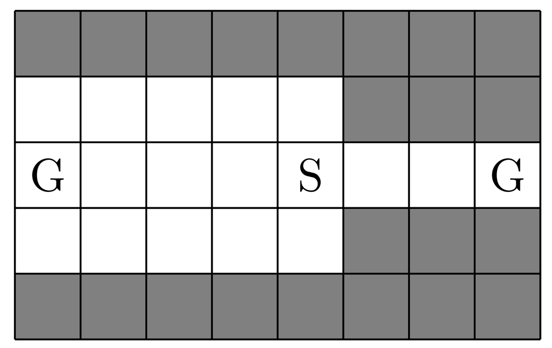

Figure: Non-Stationary bridge MDP.

The value of $\epsilon \in [0, 1]$ defines different possible evolutions:

- $\epsilon = 0$ left cells are slippery;
- $\epsilon = 1$ right cells are slippery;
- $\epsilon \in (0, 1)$ linear balance between extreme cases.

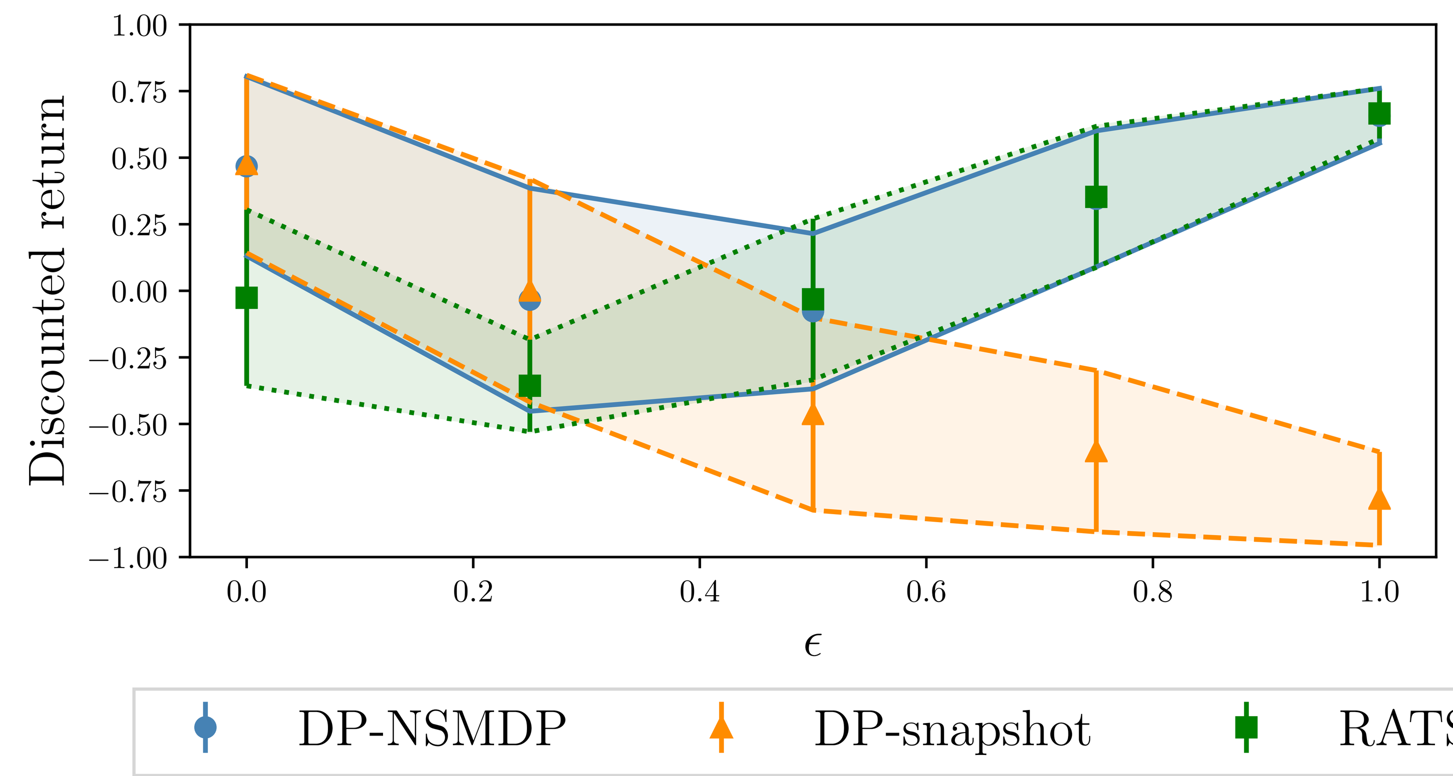

Figure: Discounted return vs ϵ, 50% of standard deviation.

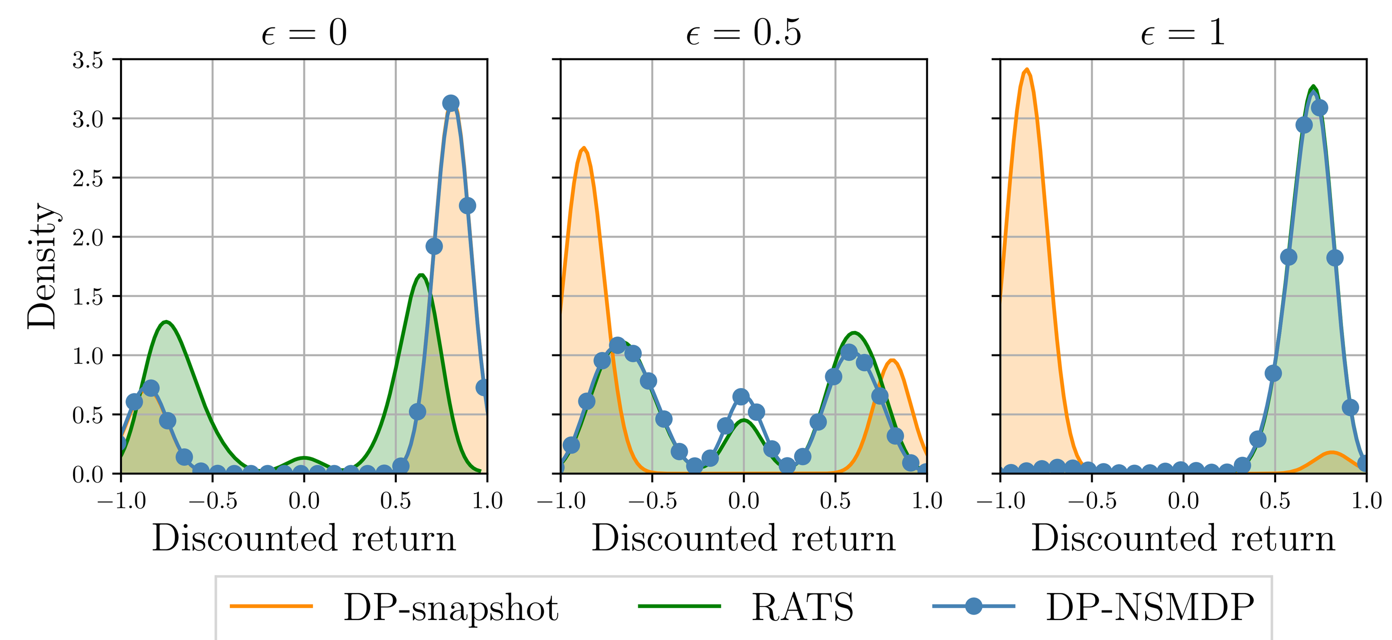

Figure: Discounted return distributions $\epsilon \in \{0, 0.5, 1\}$.