POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Dealing with non-stationarity in Constrained Markov Decision Processes

Laurea Magistrale in Mathematical Engineering - Ingegneria Matematica

Author: Anna Lunghi

Advisor: Prof. Nicola Gatti

Co-advisors: Francesco Emanuele Stradi, Prof. Matteo Castiglioni, Prof. Alberto Marchesi

Academic year: 2023-2024

1. Introduction

1.1. Goal

This thesis develops algorithms for safe sequential decision making in a non-stationary environment, modeled as constrained Markov decision processes (CMDP). The thesis starts from the negative result of [2], which states that under adversarial rewards and costs constraints it is not possible to achieve both sub-linear cumulative regret and sub-linear cumulative violation constraints. To ease this result we propose an algorithm that, given a measure of non-stationarity on the environment C (it can be seen as a measure of adversariality too), achieves both cumulative regret and cumulative violation constraints of order $\widetilde{\mathcal{O}}(\sqrt{T} + C)$. Algorithm `NS-SOPS` achieves this result starting from previous knowledge on C, while `Lag-FTRL` extend the result to the case where there is no previous knowledge on C. Finally this thesis present also algorithm `BVOPS-NSC`, useful in case of asymmetric non-stationarity, with high non-stationarity on the rewards (C_r) and low (ideally sublinear) non-stationarity on the costs (C_G). The algorithm achieves cumulative regret of order $\widetilde{\mathcal{O}}(\sqrt{T})$ and cumulative positive constraints violation of order $\widetilde{\mathcal{O}}(\sqrt{T} + C_G)$, though it requires knowledge of C_G.

1.2. Preliminaries

We study *episodic constrained* MDPs (CMDPs), in which a learner interacts with an unknown environment over T episodes, with goal of maximizing long-term rewards subject to some constraints. The CMDPs are defined as tuples $M = \left(X, A, P, \{\mathcal{R}_t\}_{t=1}^{T}, \{\mathcal{G}_t\}_{t=1}^{T}, \alpha\right)$.

- X is a finite set of states of the environment. Without loss of generality this thesis considers *loop-free* CMDPs, *i.e.* X is partitioned into L layers $X_0, \dots, X_L$ such that the first and the last layers are singletons ($X_0 = \{x_0\}$ and $X_L = \{x_L\}$).
- A is a finite set of actions available to the learner in each state
- $P : X \times A \times X \to [0, 1]$ is the transition function, with $P(x'|x, a)$ denoting the probability of going from state $x \in X$ to $x' \in X$ by taking action $a \in A$
- $\mathcal{R}_t$ probability distribution from which reward vector $r_t \in [0, 1]^{|X \times A|}$ is sampled , with $r_t(x, a)$ being the reward of taking ac-

tion $a \in A$ in state $x \in X$ at episode t.We consider a setting in which the sequences of probability distributions $\{\mathcal{R}_t\}_{t=1}^T$ are selected *adversarially*. In particular, we define the adversarial corruption C_r for the rewards as follows:

$$C_r := \min_{r \in [0,1]^{|X \times A|}} \sum_{t \in [T]} \|\mathbb{E}[r_t] - r\|_1 .$$

- $\mathcal{G}_t$ is the probability distribution from which the constraint cost matrix $G_t \in [0,1]^{|X \times A| \times m}$ is sampled, with $g_{t,i}(x,a)$ being the cost of constraint $i \in [m]$ [1] when taking action $a \in A$ in state $x \in X$ at episode t. We consider a setting in which the sequences of probability distributions $\{\mathcal{G}_t\}_{t=1}^T$ are selected *adversarially*. We introduce the adversarial corruption C_G for constraint costs, which is defined as follows:

$$C_G := \min_{g \in [0,1]^{|X \times A|}} \sum_{t \in [T]} \max_{i \in [m]} \|\mathbb{E}[g_{t,i}] - g\|_1.$$

- $\alpha \in [0, L]^m$ is the vector encoding the thresholds of all the m constraints , with α_i denoting the threshold of the i-th constraint.

Finally, we introduce the total adversarial corruption C, which is defined as $C := \max\{C_G, C_r\}$.In this thesis we derive most results in terms of the total adversarial corruption C.

Algorithm 1 Learner-Environment Interaction

1: $\mathcal{R}_t$ and $\mathcal{G}_t$ are chosen *adversarially*
2: Choose a policy $\pi_t : X \times A \to [0,1]$
3: Observe initial state x_0
4: **for** $k = 0, \dots, L-1$ **do**
5: Play $a_k \sim \pi_t(\cdot|x_k)$
6: Observe $r_t(x_k, a_k)$ and $g_{t,i}(x_k, a_k)$ for $i \in [m]$
7: Observe new state $x_{k+1} \sim P(\cdot|x_k, a_k)$
8: **end for**

1.3. Occupancy measures

We introduce here *occupancy measures*. Given a transition function P and a policy π, the occupancy measure $q^{P,\pi} \in [0,1]^{|X \times A \times X|}$ induced by P and π is such that, for every $x \in X_k$, $a \in A$, and $x' \in X_{k+1}$ with $k \in [0 \dots L-1]$:

$$q^{P,\pi}(x,a,x') := \mathbb{P}[x_k = x, a_k = a, x_{k+1} = x' | P, \pi], \tag{1}$$

which represents the probability that, under P and π, the learner reaches state x, plays action a, and gets to the next state x'. Moreover, we also define the following quantities:

$$q^{P,\pi}(x,a) := \sum_{x' \in X_{k+1}} q^{P,\pi}(x,a,x') \quad \text{and}$$

$$q^{P,\pi}(x) := \sum_{a \in A} q^{P,\pi}(x,a).$$

The following proposition characterizes when a vector $q \in [0,1]^{|X \times A \times X|}$ is a *valid* occupancy measure.

Proposition 1.1 ([3]). *A vector $q \in [0,1]^{|X \times A \times X|}$ is a valid occupancy measure of an episodic loop-free CMDP if and only if it satisfies the following conditions:*
for all $k \in [0 \dots L-1]$ it holds $\sum_{x \in X_k} \sum_{a \in A} \sum_{x' \in X_{k+1}} q(x,a,x') = 1$,
for all $k \in [1 \dots L-1]$ it holds $\sum_{a \in A} \sum_{x' \in X_{k+1}} q(x,a,x') = \sum_{x' \in X_{k-1}} \sum_{a \in A} q(x',a,x)$,
and finally $P^q = P$, where P is the transition function of the CMDP and P^q is the one induced by q.

Notice that any valid occupancy measure q induces a transition function P^q and a policy π^q as:

$$P^q(x'|x,a) = \frac{q(x,a,x')}{q(x,a)} \quad \text{and}$$

$$\pi^q(a|x) = \frac{q(x,a)}{q(x)}. \tag{2}$$

1.4. Performance metrics to evaluate learning algorithms

In order to define the performance metrics used to evaluate our *online* learning algorithms, we need to introduce an *offline* optimization problem. Given a CMDP with transition function P, we define the following parametric *linear program* (Program (3)), which is parametrized by a reward vector $r \in [0,1]^{|X \times A|}$, a constraint cost matrix $G \in [0,1]^{|X \times A| \times m}$ and a threshold vector $\alpha \in [0, L]^m$.

$$\mathrm{OPT}_{r,G,\alpha} := \begin{cases} \max_{q \in \Delta(P)} & r^\top q \quad \text{s.t.} \\ & G^\top q \le \alpha, \end{cases} \tag{3}$$

[1] In this thesis, $[a \dots b]$ denotes the set of all the natural numbers from $a \in \mathbb{N}$ to $b \in \mathbb{N}$ (both included), while $[b] := [1 \dots b]$ is the set of the first $b \in \mathbb{N}$ natural numbers.

where $q \in [0,1]^{|X\times A|}$ is a vector encoding an occupancy measure and $\Delta(P)$ is the set of all valid occupancy measures given the transition function P.

We also introduce a problem-specific *feasibility parameter* related to Program (3), denoted by $\rho \in [0, L]$ and formally defined as $\rho := \sup_{q\in\Delta(P)} \min_{i\in[m]} \left[\alpha - G^\top q\right]_i$.[2] Intuitively, ρ represents by how much feasible solutions to Program (3) strictly satisfy the constraints.

We say that an instance of Program (3) satisfies *Slater's condition* if there exists an occupancy measure $q^\circ \in \Delta(P)$ such that $G^\top q^\circ < \alpha$.

Notice that Slater Condition is equivalent to say that ρ, the *feasibility parameter*, is strictly greater than 0.

As performance metrics used to evaluate our learning algorithm, let's introduce the notion of *(cumulative) regret* and *positive (cumulative) constraints violation*. In particular the cumulative regret measures the cumulative difference in terms of rewards from the optimal static solution, while positive cumulative constraints violation measures the the violation on the costs constraints. We define the cumulative regret over T episodes as follows:

$$R_T := T \cdot \mathrm{OPT}_{\overline{r},\overline{G},\alpha} - \sum_{t\in[T]} \mathbb{E}[r_t]^\top q^{P,\pi_t},$$

where $\overline{r} := \frac{1}{T}\sum_{t\in[T]} \mathbb{E}[r_t]$ and $\overline{G} := \frac{1}{T}\sum_{t\in[T]} \mathbb{E}[G_t]$. Furthermore, we define the positive cumulative constraint violation over T episodes as follows:

$$V_T := \max_{i\in[m]} \sum_{t\in[T]} \left[\mathbb{E}[G_t]^\top q^{P,\pi_t} - \alpha\right]_i^+,$$

where $[\cdot]^+ := \max\{0, \cdot\}$. Notice that, in our definition of V_T, constraint violations are *not* allowed to cancel out across episodes.

2. Learning with Known C

The algorithm keeps track of suitably-defined confidence bounds for transition probabilities, rewards, and constraint costs. At each episode $t \in [T]$, the algorithm builds a confidence set $\mathcal{P}_t$ for the transition function P by following the same approach as [1]. Instead, for rewards and constraint costs, the algorithm adopts novel *enlarged* confidence bounds, which are suitably designed to tackle non-stationarity.

Algorithm 2 Non-stationary safe optimistic policy search (`NS-SOPS`)

Require: C, $\delta \in (0,1)$
1: $\pi_1 \leftarrow$ select any policy
2: **for** $t \in [T]$ **do**
3: Choose policy π_t in Algorithm 1 and observe feedback from interaction
4: Compute $\mathcal{P}_t$, $\overline{r}_t$, and $\underline{G}_t$
5: $q \leftarrow$ solution to $\texttt{OPT-CB}_{\Delta(\mathcal{P}_t),\overline{r}_t,\underline{G}_t,\alpha}$
6: **if** problem is *feasible* **then**
7: $\widehat{q}_{t+1} \leftarrow q$
8: **else**
9: $\widehat{q}_{t+1} \leftarrow$ take any $q \in \Delta(\mathcal{P}_t)$
10: **end if**
11: $\pi_{t+1} \leftarrow \pi^{\widehat{q}_{t+1}}$
12: **end for**

Given $\delta \in (0,1)$, by letting $N_t(x,a)$ be the total number of visits to the state-action pair $(x,a) \in X \times A$ up to episode t (excluded), the confidence bound for the reward $r_t(x,a)$ is:

$$\phi_t(x,a) := \min\Big\{1, \sqrt{\frac{\ln\left(2T|X||A|/\delta\right)}{2\max\{N_t(x,a),1\}}} + \\ + \frac{C}{\max\{N_t(x,a),1\}} + \frac{C}{T}\Big\},$$

while the confidence bound for the constraint costs $g_{t,i}(x,a)$ is defined as:

$$\xi_t(x,a) := \min\Big\{1, \sqrt{\frac{\ln\left(2mT|X||A|/\delta\right)}{2\max\{N_t(x,a),1\}}} + \\ + \frac{C}{\max\{N_t(x,a),1\}} + \frac{C}{T}\Big\}.$$

Intuitively, the first term in the expressions above is derived from Azuma-Hoeffding inequality, the second term allows to deal with the non-stationarity of rewards and constraint costs, while the third term is needed to bound how much the average reward vector $\overline{r}$ and the average constraint costs $[\overline{G}]_i$ differ from their "fictitious" non-corrupted counterparts . Algorithm 2 also computes empirical rewards and constraint costs. At each episode $t \in [T]$, for any state-action pair $(x,a) \in X \times A$ and constraint $i \in [m]$,

[2] Given a vector y, we denote by $[y]_i$ its i-th component.

these are defined as follows:

$$\widehat{r}_t(x,a) \coloneqq \frac{\sum_{\tau\in[t]} \mathbb{I}_\tau(x,a) r_\tau(x,a)}{\max\{N_t(x,a),1\}} \quad \text{and}$$
$$\widehat{g}_{t,i}(x,a) \coloneqq \frac{\sum_{\tau\in[t]} \mathbb{I}_\tau(x,a) g_{\tau,i}(x,a)}{\max\{N_t(x,a),1\}},$$

where $\mathbb{I}_\tau(x,a) = 1$ if and only if (x,a) is visited during episode τ, while $\mathbb{I}_\tau(x,a) = 0$ otherwise. For ease of notation, we let $\widehat{G}_t \in [0,1]^{|X\times A|\times m}$ be the matrix with components $\widehat{g}_{t,i}(x,a)$. Algorithm 2 selects policies with an UCB-like approach encompassing *optimism* in both rewards and constraints satisfaction. Specifically, at each episode $t \in [T]$ and for any state-action pair $(x,a) \in X \times A$, the algorithm employs an *upper* confidence bound for the reward $r_t(x,a)$, defined as $\overline{r}_t(x,a) \coloneqq \widehat{r}_t(x,a) + \phi_t(x,a)$, while it uses *lower* confidence bounds for the constraint costs $g_{t,i}(x,a)$, defined as $\underline{g}_{t,i}(x,a) \coloneqq \widehat{g}_{t,i}(x,a) - \xi_t(x,a)$ for every constraint $i \in [m]$. Then, by letting $\overline{r}_t \in [0,1]^{|X\times A|}$ be the vector with components $\overline{r}_t(x,a)$ and $\underline{G}_t$ be the matrix with entries $\underline{g}_{t,i}(x,a)$, Algorithm 2 chooses the policy to be employed in the next episode $t+1$ by solving the following linear program:

$$\texttt{OPT-CB}_{\Delta(\mathcal{P}_t),\overline{r}_t,\underline{G}_t,\alpha} \coloneqq \begin{cases} \arg\max_{q\in\Delta(\mathcal{P}_t)} & \overline{r}_t^\top q \quad \text{s.t.} \\ & \underline{G}_t^\top q \le \alpha, \end{cases} \tag{4}$$

where $\Delta(\mathcal{P}_t)$ is the set of all the possible valid occupancy measures given the confidence set $\mathcal{P}_t$.

2.1. Theoretical results

Theorem 2.1. *Given any $\delta \in (0,1)$, with probability at least $1-9\delta$, Algorithm 2 attains:*

$$R_T = \tilde{\mathcal{O}}(\sqrt{T} + C)$$

Theorem 2.2. *Given any $\delta \in (0,1)$, with probability at least $1-8\delta$, Algorithm 2 attains:*

$$V_T = \tilde{\mathcal{O}}(\sqrt{T} + C).$$

Intuitively, Theorem 2.2 is proved by showing that every constraint-satisfying occupancy measure is also feasible for Program (4) with high probability. This holds since Program (4) employs lower confidence bounds for constraint costs. Thus, in order to bound V_T, it is sufficient to analyze at which rate the feasible region of Program (4) concentrates to the *true* one (*i.e.*, the one defined by $\overline{G}$ in Program (3)). Since by definition of $\xi_t(x,a)$ the feasibility region of Program (4) concentrates as $1/\sqrt{t} + C/t$, the resulting bound for the positive constraint violation V_T is of the order of $\tilde{\mathcal{O}}(\sqrt{T} + C)$. Theorem 2.1 is proved similarly to Theorem 2.2. Indeed, since every constraint-satisfying occupancy measure is feasible for Program (4) with high probability, this also holds for q^*, as it satisfies constraints by definition. Thus, since by definition of $\phi_t(x,a)$ the upper confidence bound for the rewards maximized by Program (4) concentrates as $1/\sqrt{t} + C/t$, the regret bound follows.

3. Learning with Unknown C

We develop a *meta-procedure* (called `Lag-FTRL`, see Algorithm 3) that instantiates multiple instances of an algorithm working for the case in which C is known, with each instance taking care of a different "guess" on the value of C. The algorithm plays the role of a *master* by choosing which instance $\texttt{Alg}^j$ to use at each episode. The selection is done by employing an FTRL approach with a suitable log-barrier regularization. In particular, at each episode $t \in [T]$, by letting $\texttt{Alg}^{j_t}$ be the selected instance, the `Lag-FTRL` algorithm employs the policy $\pi_t^{j_i}$ prescribed by $\texttt{Alg}^{j_t}$ and provides the observed feedback to instance $\texttt{Alg}^{j_t}$ only.

At episode $t \in [T]$ and for every $j \in [M]$, the optimistic loss estimator is defined as:

$$\ell_{t,j} \coloneqq \frac{\mathbb{I}(j_t = j)}{w_{t,j} + \gamma} \Bigg(L - \sum_{k\in[0\dots L-1]} r_t(x_k^t, a_k^t) + \Lambda \sum_{i\in[m]} \left[(\widehat{G}_t^j)^\top \widehat{q}_t^j - \alpha \right]_i^+ \Bigg), \tag{5}$$

where γ is a suitably-defined implicit exploration factor, (x_k^t, a_k^t) is the state-action pair visited at layer k during episode t, Λ is a suitably-defined upper bound on the optimal values of Lagrangian multipliers. The choice of the multiplicative factor in the Lagrangian loss function is based on the following result.

Theorem 3.1. *Given a CMDP with a transition function P, for every reward vector $r \in [0,1]^{|X\times A|}$, constraint cost matrix $G \in$*

$[0,1]^{|X\times A|\times m}$, *and threshold vector* $\alpha \in [0,L]^m$, *if Program* (3) *satisfies Slater's condition , then the following holds:*

$$\begin{aligned}\max_{q\in\Delta(P)} \mathcal{L}(L/\rho, q) \\ &= \max_{q\in\Delta(P)} r^\top q - \frac{L}{\rho}\sum_{i\in[m]}\left[G^\top q-\alpha\right]_i^+ \\ &= \mathrm{OPT}_{r,G,\alpha},\end{aligned}$$

where ρ *is the feasibility parameter of Program* (3).

Theorem 3.1 intuitively shows that a L/ρ multiplicative factor on the positive constraint violation is enough to compensate the large rewards that non-feasible policies would attain when employed by the learner. This result is crucial since, without properly defining the Lagrangian function optimized by Algorithm 3, the FTRL optimization procedure would choose instances with both large rewards and large constraint violation, thus preventing the violation bound from being sublinear.

Algorithm 3 Lagrangified follow-the-regularized-leader (`Lag-FTRL`)

Require: $\delta \in (0,1)$
$\Lambda \leftarrow \frac{Lm+1}{\rho}$, $M \leftarrow \lceil \log_2 T \rceil$
$\gamma \leftarrow \sqrt{\ln(M/\delta)/TM}$, $\eta \leftarrow \frac{1}{2Lm\Lambda}\sqrt{\ln(T)/T}$
for $j \in [M]$ **do**
  $\texttt{Alg}^j \leftarrow$ Algorithm 2 with $C = 2^j$
end for
$w_{1,j} \leftarrow 1/M$ for all $j \in [M]$
for $t \in [T]$ **do**
  Sample index $j_t \sim w_t$
  $\pi_t^{j_t} \leftarrow$ policy that $\texttt{Alg}^{j_t}$ would choose
  Choose policy $\pi_t^{j_t}$ in Algorithm 1 and observe feedback from interaction
  Let $\texttt{Alg}^{j_t}$ observe received feedback
  for $j \in [M]$ **do**
    Build $\ell_{t,j}$ as in Equation (5)
  end for
  $w_{t+1} \leftarrow \arg\min_{w\in\Delta_M} w^\top \sum_{\tau\in[t]}\ell_t + \frac{1}{\eta}\sum_{j\in[M]}\frac{1}{w_j}$
end for

3.1. Theoretical Results

Theorem 3.2. *Given any* $\delta \in (0,1)$*, with probability at least* $1-26\delta$*, Algorithm 3 attains:*

$$R_T = \tilde{\mathcal{O}}\left(\sqrt{T}+C\right).$$

The regret bound is strongly related to the optimal value of the problem associated with the positive Lagrangian, which, by definition of $[\cdot]^+$ cannot perform worse than the optimum of Program (3), in terms of rewards gained. Thus, by letting j^* be the index of the instance associated with true corruption value C, proving Theorem 3.2 reduces to bounding the regret and the constraint violation of instance $\texttt{Alg}^{j^*}$, with the additional challenge of bounding the estimation error of the optimistic loss estimator. Finally, by means of the results for the *known* C case , we are able to show that the regret is at most $\tilde{\mathcal{O}}(\sqrt{T}+C)$, which is the desired bound.

Theorem 3.3. *Given any* $\delta \in (0,1)$*, with probability at least* $1-30\delta$*, Algorithm 3 attains:*

$$V_T = \tilde{\mathcal{O}}\left(\sqrt{T}+C\right).$$

Intuitively, to prove Theorem 3.3, it is necessary to bound the negative regret attained by the algorithm, *i.e.*, how better Algorithm 3 can perform in terms of rewards with respect to an optimal occupancy in hindsight q^*. Notice that this is equivalent to showing that the FTRL procedure cannot gain more than $\mathrm{OPT}_{\overline{r},\overline{G},\alpha}$ by playing policies that are *not* feasible, or, equivalently, by choosing instances $\texttt{Alg}^j$ with a large corruption guess, which, by definition of the confidence sets employed by Algorithm 2, may play non-feasible policies attaining large rewards. This is done by employing Theorem 3.1, which shows that the positive Lagrangian does *not* allow the algorithm to achieve too large rewards with respect to q^*. Thus, the violations are still upper bounded by $\tilde{\mathcal{O}}(\sqrt{T}+C)$.

4. Adversarial rewards with C_G known

In the field of non-stationary CMDPs, it may be useful to consider an environment where only one of the two adversarial corruptions is potentially linear, while the other remains sublinear. Specifically, we consider the case where the adversarial reward corruption is potentially linear and the adversarial corruption on the cost constraints is known (or at least an overestimation of it is known) and sublinear. In this scenario, we use an approach based on Online Mirror Descent (OMD), inspired by the unconstrained algorithm proposed by [1], which deals with adver-

sarial losses, unknown transition probabilities, and bandit feedback.

Algorithm 4 Bounded Violation Optimistic Policy Search for Non-stationary Constraints with known C_G (BVOPS-NSC)

Require: $C_G, \delta \in (0,1)$

1: $\eta, \gamma \leftarrow \sqrt{\frac{L \ln(\frac{L|X||A|}{\delta})}{T|X||A|}}$

2: Initialize , $\widehat{q}_1(x,a,x') \leftarrow \frac{1}{|X_k||A||X_{k+1}|}$, $\forall (x,a,x') \in X_k \times A \times X_{k+1},\ k \in [0, L-1]$

3: $\pi_1 \leftarrow \pi^{\widehat{q}_1}$

4: **for** $t \in [T]$ **do**

5: Choose policy π_t in Algorithm 1 and observe feedback from interaction

6: Compute $\mathcal{P}_t$, f_t, and $\underline{G}_t$

7: $\widehat{q}_{t+1} \longleftarrow \underset{q \in \Delta(\mathcal{P}_t): \underline{G}_t^\top q \leq \alpha}{\arg\min} \sum_{x,a} f_t(x,a) q(x,a) + D(q||\widehat{q}_t)$

8: Update policy $\pi_{t+1} = \pi^{\widehat{q}_{t+1}}$

9: **end for**

We define the OMD loss estimator function $f_t(x,a)$ as :

$$f_t(x,a) := \frac{\mathbb{I}_t(x,a)}{u_t(x,a) + \gamma}(1 - r_t(x,a)), \qquad (6)$$

where

$$u_t(x,a) := \sup_{P' \in \mathcal{P}_t} q^{\pi_t, P'}(x,a). \qquad (7)$$

4.1. Theoretical Results

Theorem 4.1. *For any $\delta \in (0,1)$, Algorithm 4 attains, with probability at least $1 - 8\delta$:*

$$V_T = \widetilde{\mathcal{O}}(\sqrt{T} + C_G).$$

Theorem 4.2. *For any $\delta \in (0,1)$, with probability at least $1 - 9\delta$, Algorithm 4 attains:*

$$R_T = \widetilde{\mathcal{O}}(\sqrt{T}).$$

Notice that the regret can be bounded using a similar resoning to [1], while the positive constraints violation is proven following the same exact reasoning as in the prevoious sections.

5. Conclusion

In conclusion, we presented `NS-SOPS`,an innovative algorithm that operates in non-stationary CMDPs and achieves regret and positive constraint violation of order $\tilde{\mathcal{O}}(\sqrt{T} + C)$ when C is known. Additionally, we introduced `Lag-FTRL`, a meta-procedure that attains the same guarantees when C is not known *a priori*. Finally, we proposed Algorithm `BVOPS-NSC`, a hybrid algorithm combining the `NS-SOPS` designed in this thesis and `UOB-REPS` from [1], which is effective in scenarios with high corruption on the rewards, extending beyond the intended use cases of the previous algorithms.

References

[1] Chi Jin, Tiancheng Jin, Haipeng Luo, Suvrit Sra, and Tiancheng Yu. Learning adversarial Markov decision processes with bandit feedback and unknown transition. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4860–4869. PMLR, 13–18 Jul 2020.

[2] Shie Mannor, John N. Tsitsiklis, and Jia Yuan Yu. Online learning with sample path constraints. *Journal of Machine Learning Research*, 10(20):569–590, 2009.

[3] Aviv Rosenberg and Yishay Mansour. Online convex optimization in adversarial Markov decision processes. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5478–5486. PMLR, 09–15 Jun 2019.