POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Learning in Constrained Tree-Form Sequential Decision problems

Laurea Magistrale in Computer Science and Engineering - Ingegneria Informatica

Author: Emanuele Pocelli

Advisor: Prof. Matteo Castiglioni

Co-advisor: Francesco Emanuele Stradi, Prof. Alberto Marchesi

Academic year: 2024-2025

1. Introduction

This research explores online learning in constrained environments. In particular, it focuses on settings characterized by a sequential structure and imperfect information, formally modeled as Constrained Tree-Form Sequential Decision Making (CTFSDM) problems. In this context, a novel algorithm is developed to improve computational efficiency over existing approaches, without requiring additional assumptions about the environment.

Current methods typically rely on linear programming formulations, which limit scalability and efficiency. In contrast, the proposed approach adopts a primal–dual framework that leverages the state-of-the-art algorithm for unconstrained adversarial settings, namely imperfect-information zero-sum games. This allows to overcome the aforementioned limitations, while maintaining optimal performance guarantees.

2. Preliminaries

The formal setting considered in this work is a Constrained Tree-Form Sequential Decision-Making problem (CTFSDM) with stochastic rewards and constraints, and unknown transition dynamics. In this framework, a single agent iteratively interacts with the environment by taking sequential decisions and observing feedback. The objective is to maximize cumulative reward, while ensuring that the constraints remain below predefined thresholds. Reward and constraint functions are assumed to be bounded in $[0, 1]$. Additionally, the agent does not have direct knowledge of the underlying states; instead, it observes signals emitted by the states.

Formally, a CTFSDM problem is defined by a tuple $< S, X, A, \nu, m, r, g, \alpha >$, where:

- S set of states organized in a tree structure;
- X information set, partition of S. For a state $s \in S$, $x(s) \in X$ denotes an information set such that $s \in x(s)$;
- A_x set of actions available in $x \in X$;
- $\nu : S \to [0, 1]^{|X|}$ probability distribution, with $\nu_s(x)$ denoting the probability of the state s emitting signal x;
- $m \in \mathbb{N}_{>0}$ number of constraints;
- $r : S \times A \to \mathbb{R}$ is the reward function. $r(s, a)$ represents the reward taken playing action a in state s. $r \in [0, 1]^{|S \times A|}$ represents the corresponding reward vector;
- $g \in [0, 1]^{m|S \times A|}$ is the cost function. $g(s, a)$ represents the costs taken playing action a in state s. $G \in [0, 1]^{m|S \times A|}$ represents cost matrix, with $g_i(s, a)$ representing the cost incurred playing action a in state s for each

constraint $i \in [m]$[1];
- $\alpha \in [0, H]^m$ threshold vector, the agent must keep the total costs for the i-th constraint below α_i, $i \in [m]$.

Within this setting, policies can be parameterized using the *sequence-form* representation, which enables expressing the reward or cost achieved by a policy in an episode as a scalar product. Formally, a sequence policy $\pi \in \Pi$ is a vector that assigns to each information set/action pair (x, a) its corresponding realization probability. Then, for any $x \in X$, let $\theta(x)$ be the probability of observing x due to environment transitions only. The expected utility that can be obtained by playing (x, a) can be written as:

$$r^*(x, a) := \theta(x)\mathbb{E}[r(x, a)],$$

and, similarly, the expected cost of the i-th constraint:

$$g_i^*(x, a) := \theta(x)\mathbb{E}[g_i(x, a)].$$

These probabilities are collected into a vector $\theta \in [0, 1]^{|X|}$ where each element θ_x corresponds to $\theta(x)$. Finally, let r^* and g^* be the vectors where each element $r^*_{x,a}$, $g^*_{i,x,a}$ corresponds to $r^*(x, a)$ and $g_i^*(x, a)$ respectively. The expected reward received following a sequence policy π in a round is given as the scalar product $\pi^\top r^*$. Similarly for the costs: $\pi^\top g_i^*$.
In this formulation, *strong* regret is defined as:

$$R_T := \sum_{t=1}^{T} \left[\pi^{*\top} r^* - \pi_t^\top r^*\right]^+,$$

where $[\cdot]^+ := \max\{0, \cdot\}$. Similarly, the *strong* violation is defined as:

$$V_T := \max_{i \in [m]} \sum_{t=1}^{T} \left[\pi^\top g_i^* - \alpha_i\right]^+.$$

Another concept relevant for this work is the *Lagrangian function*, which combines both rewards and constraints:

Definition 1. *Given a CTFSDM problem characterised by a reward vector $r \in [0, 1]^{|X \times A|}$ and cost matrix $G \in [0, 1]^{i \times |X \times A|}$, the Lagrangian function $\mathcal{L}_{r,G}(\pi, \lambda) : \Pi \times \mathbb{R}^m_{\geq 0} \to \mathbb{R}$ is defined for every sequence $\pi \in \Pi$ and lagrangian vector $\lambda \in \mathbb{R}^m_{\geq 0}$*

$$\mathcal{L}_{r,G}(\pi, \lambda) := \pi^\top r^* - \sum_{i \in [m]} \lambda_i (\pi^\top g_i^* - \alpha_i).$$

[1] $[m] := \{0, 1, ..., m - 1\}$

Figure 1 illustrates an example of a CTFSDM.

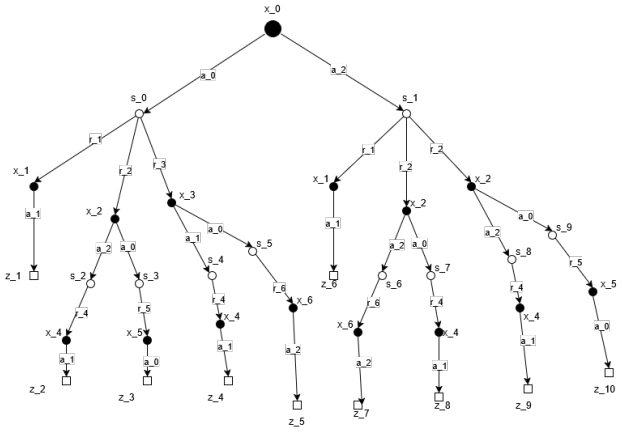

Figure 1: Example of CTFSDM: observation nodes (denoted by s) follow the action nodes (denoted by x): in each action node, the agent takes an action, the environment transitions to the next state that emits a signal, until a terminal node (denoted by z) is reached. It is worth noting that different states may emit the same signal: these states are indisinguishable to the agent and the available actions are identical.

The overall agent-environment interaction is detailed below:

Algorithm 1 Agent-Environment interaction

Input: Policy $\pi : X \times A \to [0, 1]$
Environment initializes state $s_0 \in S$
for $h = 0, ..., H - 1$ **do**
  environment draws signal $x_h \sim \nu_{s_h}$
  agent observes x_h
  agent plays action $a_h \sim \pi(\cdot \mid x_h)$
  agent receives reward $r(s_h, a_h)$
  agent incurs in cost $g_i(s_h, a_h)$ for each $i \in [m]$
  environment evolves to state s_{h+1}
end for

3. Related Works

Tree-Form Sequential Decision-Making (TFSDM) problems have been originally introduced by Farina et al. [2], who focus on the adversarial setting. In contrast, the environment considered in Section 2 is *constrained* and *stochastic*.
The setting proposed in this work is closely related to the soft-threshold formulation of Bernasconi et al. [1], with one important

distinction: in their framework, the agent receives feedback only at terminal nodes, whereas in this case, feedback is observed after every action. Moreover, while their proposed algorithm achieves optimal guarantees, it relies on a linear programming formulation, which can be computationally demanding. The present work addresses this limitation, by introducing a novel algorithm that is computationally efficient, scalable, and does not require additional assumptions on the environment, while still maintaining strong theoretical guarantees.
It is worth noting that, despite the difference in feedback structure, the latter formulation can be adapted to the first by accumulating feedback across action steps and revealing it only at the terminal node.
Furthermore, Fiegel et al. [3] recently proposed the Adaptive-FTRL algorithm, which represents the state of the art for regret minimization in unconstrained adversarial tree-form environments, specifically imperfect-information zero-sum games. The structural connection between such environments and CTFSDMs motivates its use as the main component in the present approach.

4. CPD-FTRL

The proposed algorithm builds on a primal–dual scheme similar to that of Stradi et al. [4]. Unlike standard approaches in the literature that perform mirror-descent updates on the dual variables, the Lagrange multipliers are selected as a binary decision at each round between two values, 0 and $^{H+1}/_{\rho}$, for each $i \in [m]$, in order to maximize the constraint penalization term $(\pi_t^\top \widetilde{g}_{i,t} - \alpha_i), \forall i \in m$. On one hand, the value 0 is selected when the optimistic estimation of the i-th constraint is not violated by the current policy. On the other hand, if the optimistic estimation is not satisfied, then the value assigned is $^{H+1}/_{\rho}$. This quantity is chosen to be large enough to guarantee that any policy cannot gain more utility than the optimal policy that satisfies all constraints.

4.1. Estimators

Estimators are used to construct artificial feedback, which is then fed to the primal algorithm and used to update the dual variables.
More precisely, let $N_t(x,a)$ be the number of episodes in which each pair $(x,a) \in X \times A$ is visited until time $t-1 \in [T]$ and let $\mathbb{1}_\tau(x,a)$ be the indicator function equal to 1 when, at round τ the pair $(x,a) \in X \times A$. Then, the following estimators can be defined:

- $\hat{r}_t(x,a) := \frac{\sum_{\tau=1}^{t-1} r_\tau(x,a)\mathbb{1}_\tau(x,a)}{max\{1,N_t(x,a)\}}$,
- $\hat{g}_{i,t}(x,a) := \frac{\sum_{\tau=1}^{t-1} g_{i,\tau}(x,a)\mathbb{1}_\tau(x,a)}{max\{1,N_t(x,a)\}}$.

These estimators are unbiased, as they represent the empirical mean of the observed rewards and constraints for each state-action pair (x,a).
A UCB-style approach is then used to build an optimistic model of reward and cost functions so that, with high probability, the estimation error does not exceed prescribed bounds. More specifically, the reward confidence bound is defined as

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

while the confidence bound for the constraints is

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

An additional estimator is required for the dual update, whose purpose is to evaluate the current policy and update the Lagrange multipliers accordingly.
Let $\mathbb{1}_\tau^{play}(x,a)$ be the indicator function equal to 1 when, at round τ, a sequence π_τ that leads to the action pair (x,a) is played, and $\mathbb{1}_\tau^{env}(x,a)$ the indicator function equal to 1 when, at round τ, following the agent policy, the environment actually leads to the pair (x,a). Finally, given $N_t^{play} := \sum_{\tau=0}^{t} \mathbb{1}_\tau^{play}(x,a)$, the following estimator is defined:

$$\widetilde{\hat{g}}_{i,t}(x,a) = \frac{\sum_{\tau=1}^{t-1} g_{i,\tau}(x,a)\mathbb{1}_\tau^{play}(x,a)\mathbb{1}_\tau^{env}(x,a)}{N_t^{play}(x,a)}.$$

The quantity $\widetilde{\hat{g}}_i$ estimates g_i^* and implicitly contains the transitions of the environment. The corresponding confidence bound is defined as:

$$\widetilde{\xi}_t(x,a) := \min\left\{1, \frac{\sqrt{\ln(T|X||A|m)/\delta}}{max(1,N_t^{play}(x,a)}\right\}.$$

4.2. Algorithm

The prospect of the algorithm built in this work is the following:

Algorithm 2 Constrained Primal-Dual Follow the Regularized Leader (CPD-FTRL)

Input: Number of rounds $T \in \mathbb{N}_{>0}$, problem-specific parameter $\rho \in [0, H]$, confidence $\delta \in (0, 1)$
Initialise $\pi_0 \leftarrow$ uniform policy
Initialise all estimators, bounds and counters
for $t = 1, ..., T$ **do**
  Interact as in Algorithm 1 with $\pi = \pi_t$
  Observe $(x_t, a_t), r_t(x_t, a_t), g_{t,i}(x_t, a_t)$ for every $i \in [m]$ and $h \in [H]$ as feedback
  Update estimators, bounds and counters $\hat{r}, \hat{g}, \widetilde{\hat{g}}, \phi, \xi, \widetilde{\xi}$ as prescribed in Section 4.1
  $\lambda_{t,i} \leftarrow \arg\max_{\lambda \in \{0, \frac{H+1}{\rho}\}} \lambda(\pi_t^\top \widetilde{g}_{i,t} - \alpha_i) \quad \forall i \in [m]$
  Build artificial reward for every pair (x_h, a_h) received as feedback:

$$\tilde{r}_t(x_k, a_k) \leftarrow \frac{\frac{(H+1)}{\rho} + \left[\bar{r}_t(x_k, a_k) - \sum_{i \in [m]} \lambda_{t,i}(\underline{g}_{t,i}(x_k, a_k) - \frac{\alpha_i}{H})\right]}{\frac{2(H+1)}{\rho} + 1}$$

  Update policy π_{t+1} employing **Balanced FTRL** with artificial rewards $\tilde{r}_t(x_h, a_h), \ \forall h \in \{0, ..., H-1\}$
end for

4.3. Theoretical Analysis

Algorithm 2 achieves optimal guarantees. In particular, the following theorems provide the regret and violation bounds:

Theorem 4.1. *Given any $\delta \in (0, 1)$, Algorithm 2 attains:*

$$R_t \leq \widetilde{\mathcal{O}}\left(\sqrt{\frac{H^2 m}{\rho}|A||X|T\log(1/(4\delta))}\right),$$

with probability at least $1 - \mathcal{O}(\delta)$.

Similarly,

Theorem 4.2. *Given any $\delta \in (0, 1)$, Algorithm 2 attains:*

$$V_t \leq \widetilde{\mathcal{O}}\left(\sqrt{\frac{H^2 m}{\rho}|A||X|T\log(1/(4\delta))}\right),$$

with probability at least $1 - \mathcal{O}(\delta)$.

The approach obtaines guarantees that are comparable to those of Bernasconi et al. [1], while also achieving improved computational efficiency. In detail, each policy update can be performed in $\mathcal{O}(HA)$ time [3]. The estimators and the dual variables, on the other hand, can be updated in place within the same $\mathcal{O}(HA)$ complexity. This results in an overall time complexity of $\mathcal{O}(THA)$ over all rounds.

5. Conclusions

The method presented in this work builds upon state-of-the-art algorithms for learning in adversarial, unconstrained environments and extends them through a primal–dual framework. As a result, optimal regret and constraint-violation guarantees are achieved, while improving computational efficiency compared to existing approaches, without requiring prior assumptions on the tree structure.

Future research directions can include an experimental implementation of the algorithm. In particular, benchmarking the algorithm in structured environments, such as strategic card games, would provide deeper insights into its empirical behavior. Additionally, the algorithm is suitable for real-world applications, where performance optimization must be balanced against operational constraints. Potential domains include robotics (where safety and resource limitations are critical), safe navigation, and online resource allocation and management problems, such as planning under uncertainty.

References

[1] Martino Bernasconi, Federico Cacciamani, Matteo Castiglioni, Alberto Marchesi, Nicola Gatti, and Francesco Trovò. Safe learning in tree-form sequential decision making: Handling hard and soft constraints. In *International Conference on Machine Learning*, pages 1854–1873. PMLR, 2022.

[2] Gabriele Farina, Robin Schmucker, and Tuomas Sandholm. Bandit linear optimization for sequential decision making and extensive-form games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 5372–5380, 2021.

[3] Côme Fiegel, Pierre Ménard, Tadashi Kozuno, Rémi Munos, Vianney Perchet, and Michal Valko. Adapting to game trees in zero-sum imperfect information games. In *International Conference on Machine Learning*, pages 10093–10135. PMLR, 2023.

[4] Francesco Emanuele Stradi, Matteo Castiglioni, Alberto Marchesi, and Nicola Gatti. Optimal strong regret and violation in constrained mdps via policy optimization. In *The Thirteenth International Conference on Learning Representations*, 2025.