Online Learning Methods for Pricing and Advertising

Ph.D. Thesis Defense

Marco Mussi

Milan — June 27th, 2024

Companies are continually seeking innovative strategies to:

- Enhance their **market presence**
- Capture **consumer attention**
- Optimize their **pricing models**

When we want to **sell a product online** we have to select:

- The **price** at which sell it
- How much to invest in **advertising**

Solve the problem of finding the **optimal price** and optimize the **advertising budget** in a **data-driven** way

- Design **theoretical frameworks** to **generalize** the ways in which we can **learn online** in these scenarios
- Machine Learning tools: **Multi-Armed Bandits** (MABs, Lattimore and Szepesvári, 2020)
  - We add **structure** to MABs to handle complex scenarios
  - We study the **statistical complexity** of learning in these settings

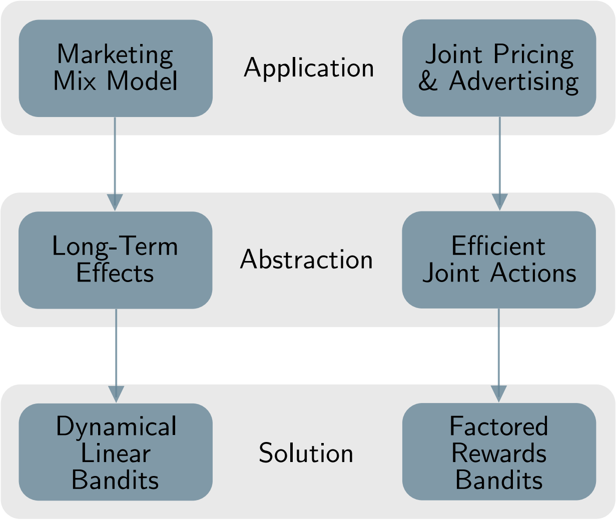
- The **base cases** in which we want to **independently optimize**:
  - Advertising budget (Nuara et al., 2022)
  - Selling price (Rothschild, 1974)

  can be both solved using **standard MAB techniques**

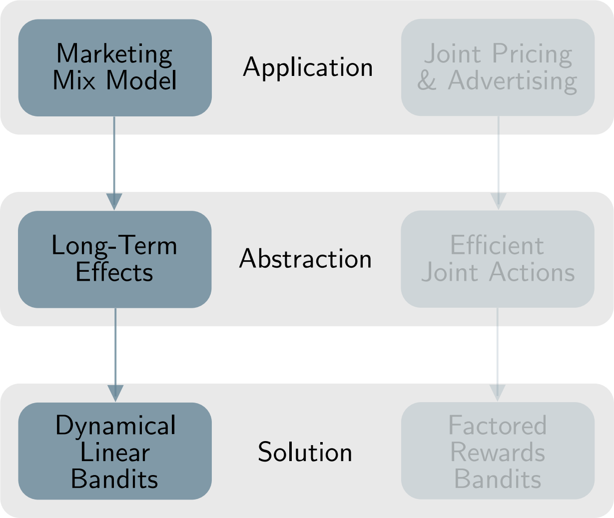
- **No work jointly optimize** the learning of a coherent pricing and advertising strategy

In this thesis, we focus on:

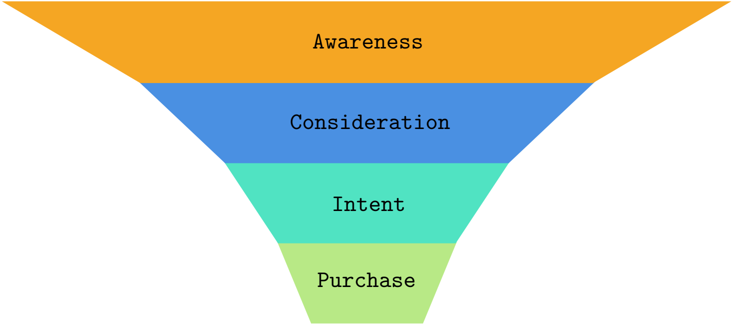
- **Challenges** of pricing and advertising:
  - Pricing: **long-tail** products (Mussi et al., 2022)
  - Pricing: temporal **autoregressive** dependencies (Bacchiocchi et al., 2024)
  - Advertising: **Marketing Mix Models (MMMs)** optimization (Mussi et al., 2023)
- **Joint optimization** of pricing and advertising strategies (Mussi et al., 2024)


Marketing Mix Model
Application
Joint Pricing & Advertising
Long-Term Effects
Abstraction
Efficient Joint Actions
Dynamical Linear Bandits
Solution
Factored Rewards Bandits

Advertising: Marketing Mix Model



Advertising: the Marketing Funnel



Why **not** Reinforcement Learning?

- We do not want the **high complexity** of RL
- In RL, this will be a **Partially-Observable MDP** (Åström, 1965) with **infinite state** and **action spaces**

Why **not** standard Multi-Armed Bandits?

- The effect of the actions lasts for **one time-step** only
- There is **no state** to model **action-dependent** phenomenas over time

We consider a problem in which:

- The **effect** of an action **persists over time**
- The effect of previous actions is modeled thanks to an **hidden state** evolving as an effect (assumed **linear**) of the **actions**

We address this problem by introducing the **Dynamical Linear Bandits** setting

- The state $\mathbf{x}_t \in \mathbb{R}^n$ is **not observable**
- The action $\mathbf{a}_t$ can be chosen in action space $\mathcal{A} \subseteq \mathbb{R}^d$
- At every time step we see a noisy realization of the reward y_t:

$$\underbrace{y_t}_{\text{Reward at time } t} = \underbrace{\langle \boldsymbol{\omega}, \mathbf{x}_t \rangle}_{\text{Current State Contribution}} + \underbrace{\langle \boldsymbol{\theta}, \mathbf{a}_t \rangle}_{\text{Action Contribution}} + \underbrace{\eta_t}_{\text{Subgaussian Random Noise}}$$

$$\underbrace{\mathbf{x}_{t+1}}_{\text{New State Vector}} = \underbrace{\mathbf{A}\mathbf{x}_t}_{\text{Previous State Contribution}} + \underbrace{\mathbf{B}\mathbf{a}_t}_{\text{Action Contribution}} + \underbrace{\boldsymbol{\epsilon}_t}_{\text{Subgaussian Random Noise}}$$

- $\boldsymbol{\omega}$, $\boldsymbol{\theta}$, $\mathbf{A}$, and $\mathbf{B}$ are **unknown**

- The goal is to minimize the **expected cumulative policy regret**:

$$\mathbb{E}[R_T(\underline{\boldsymbol{\pi}}, \underline{\boldsymbol{\nu}})] = \mathbb{E}\left[\sum_{t=1}^{T} J^* - y_t\right]$$

where J^* is the value of J corresponding to the optimal policy ($J^* = \sup_{\underline{\boldsymbol{\pi}}} J(\underline{\boldsymbol{\pi}})$), and:

$$J(\underline{\boldsymbol{\pi}}) := \liminf_{H \to +\infty} \mathbb{E}\left[\frac{1}{H}\sum_{t=1}^{H} y_t\right]$$

- (**Stability**) Spectral radius: $\rho(\mathbf{A}) < 1$
- (**Boundedness**) $\|\cdot\|_2$ of $\boldsymbol{\theta}$, $\boldsymbol{\omega}$, $\mathbf{B}$, $\mathbf{a}$, $\mathbf{x}$ bounded

Under **Stability** and **Boundedness** assumptions:

- There is an **optimal steady state**
- There is a constant optimal action $\mathbf{a}^*$:

$$\mathbf{a}^* \in \arg\max_{\mathbf{a} \in \mathcal{A}} J(\mathbf{a}) = \langle \mathbf{h}, \mathbf{a} \rangle$$

where $\mathbf{h}$ is a **Markov parameter** describing the system at the steady state:

$$\mathbf{h} = \boldsymbol{\theta} + \mathbf{B}^{\mathsf{T}}(\mathbf{I} - \mathbf{A})^{-\mathsf{T}}\boldsymbol{\omega}$$

Theorem (Lower Bound)

For any algorithm $\mathfrak{A}$, there exists an instance $\underline{\boldsymbol{\nu}}$ of DLB fulfilling **Stability** *and* **Boundedness** *assumptions, such that the expected regret is lower bounded by:*

$$\mathbb{E}[R_T(\mathfrak{A}, \underline{\boldsymbol{\nu}})] \geq \Omega\left(\frac{d\sqrt{T}}{(1-\rho(\mathbf{A}))^{\frac{1}{2}}}\right).$$

- The **knowledge** of at least an **upper bound on the maximum eigenvalue** $\rho(\mathrm{A})$ is needed
- Reduces to the **Linear Bandits** one for $\rho(\mathbf{A}) = 0$

DynLin-UCB is an **optimistic** regret minimization algorithm that operates in **epochs**:

- Epochs are in the order of $\mathcal{O}\left(\frac{\log T}{1-\overline{\rho}}\right)$
  - where $\overline{\rho} < 1$ is an upper bound on the **spectral radius** $\rho(\mathbf{A})$
- In each epoch, we **persist** the optimistic action until we reach an approximation of the **steady state**
- We estimate using a **Ridge-regularized** regression the **Markov parameters** $\widehat{\mathbf{h}}_t$ **using only the last sample**
  - when the hidden state is approximately **steady**

Theorem (Policy Regret Upper Bound)

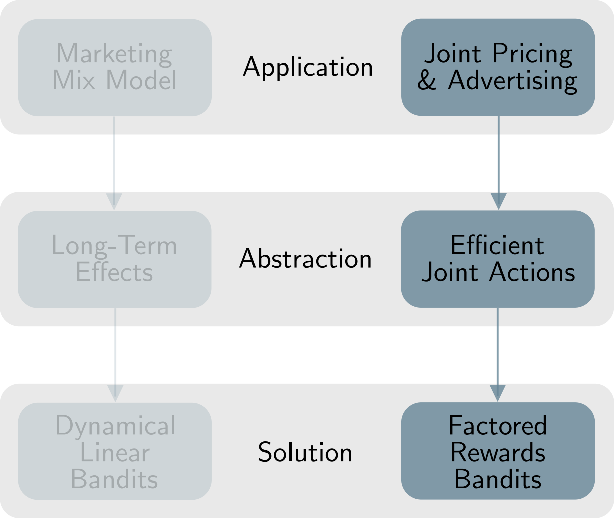
Under **Stability** *and* **Boundedness** *assumptions, properly selecting β_t,* `DynLin-UCB` *suffers an expected policy regret bounded as:*

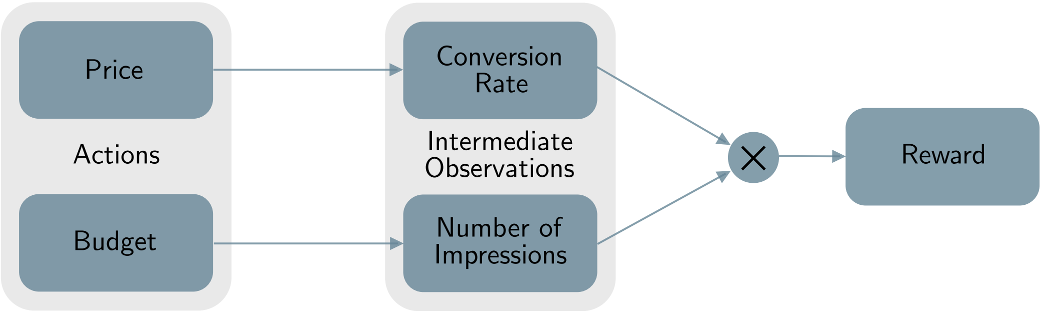
$$\mathbb{E}[R_T(\texttt{DynLin-UCB}, \underline{\boldsymbol{\nu}})] \leq \mathcal{O}\left(\frac{d\sigma\sqrt{T}(\log T)^2}{(1-\overline{\rho})^{3/2}} + \frac{1}{(1-\rho(\mathbf{A}))^2}\right).$$

- The bound reduces to the one of **Linear Bandits** for $\overline{\rho} = 0$ (up to logarithmic factors)

Future works on DLBs:

- Fill the **gap** between upper and lower bounds for $\rho(\mathbf{A})$
- Consider **non-linear dynamics** for the state evolution


Marketing Mix Model
Application
Joint Pricing & Advertising
Long-Term Effects
Abstraction
Efficient Joint Actions
Dynamical Linear Bandits
Solution
Factored Rewards Bandits


Price
Actions
Budget
Conversion Rate
Intermediate Observations
Number of Impressions
Reward

- At every round $t \in [\![T]\!]$, we choose an **action vector**:

$$\mathbf{a}(t) = (a_1(t), \ldots, a_d(t)) \in \mathcal{A} \coloneqq [\![k_1]\!] \times \cdots \times [\![k_d]\!]$$

  - $\forall i \in [\![d]\!]$ we have k_i options
  - d is the action vector dimension

- We observe a vector of d **non-correlated intermediate observations** $\mathbf{x}(t) = (x_1(t), \dots, x_d(t))$ and receive as reward the **product of the observations** $r(t) = \prod_{i \in [\![d]\!]} x_i(t)$
- The i^{th} component $x_i(t)$ of the intermediate observation vector $\mathbf{x}(t)$ is the effect of the i^{th} action component $a_i(t)$ in the action vector: $x_i(t) = \mu_{i,a_i(t)} + \epsilon_i(t)$
  - $\mu_{i,a_i(t)} \in [0, 1]$ is the **expected observation** of the i^{th} component $a_i(t)$
  - $\epsilon_i(t)$ is σ^2-subgaussian noise

- We can solve this problem using **standard Multi-Armed Bandit** techniques considering all the **price-budget couples** as actions
- However, if we look just at the reward and disregard this **factored** structure, the learning problem will:
  - Present an **unnecessarily large action space**, including all the $\prod_{i \in [\![d]\!]} k_i$ possible combinations of action components
  - Suffer an **amplified heavy-tailed noise effect** $\prod_{i \in [\![d]\!]} \epsilon_i(t)$ in the reward due to the product of the noisy intermediate observations

- An **optimal action vector** is:

$$\mathbf{a}^* = (a_1^*, \ldots, a_d^*) \in \underset{\mathbf{a}=(a_1,\ldots,a_d)\in\mathcal{A}}{\arg\max} \prod_{i\in[\![d]\!]} \mu_{i,a_i}$$

  and we abbreviate $\mu_i^* = \mu_{i,a_i^*}, \forall i \in [\![d]\!]$

- We define the **suboptimality gaps** related to:
  - the i^{th} **action component** $\Delta_{i,a_i} := \mu_i^* - \mu_{i,a_i}$ for $a_i \in [\![k_i]\!]$
  - the **action vector** $\mathbf{a} = (a_1, \ldots, a_d) \in \mathcal{A}$ as $\Delta_{\mathbf{a}} := \prod_{i\in[\![d]\!]} \mu_i^* - \prod_{i\in[\![d]\!]} \mu_{i,a_i}$

- The **goal** of an algorithm $\mathfrak{A}$ is to minimize the **expected cumulative regret**:

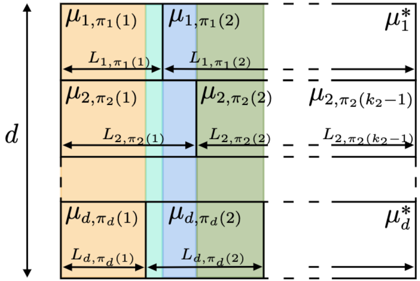
$$\mathbb{E}[R_T(\mathfrak{A}, \underline{\boldsymbol{\nu}})] := \mathbb{E}\left[T \prod_{i \in [\![d]\!]} \mu_i^* - \sum_{t \in [\![T]\!]} \prod_{i \in [\![d]\!]} \mu_{i,a_i(t)}\right] = \mathbb{E}\left[\sum_{t \in [\![T]\!]} \Delta_{\mathbf{a}(t)}\right]$$

- Every consistent algorithm $\mathfrak{A}$ has to **pull at least**:

$$\liminf_{T\to\infty} \frac{\mathbb{E}[N_{i,j}]}{\log T} \geq \frac{2\sigma^2}{\Delta_{i,j}^2}$$

  times every **suboptimal action component**.
- We have to find the **best way** in which we can combine the pulls
- Solution: formulate a **Linear Programming** problem

- We can avoid to solve the optimization problem: we have just to search for the **best way** to arrange the pulls
- We can make use of **rearrangement inequality** for integrals to find the best solution (Luttinger and Friedberg, 1976)



- A first solution we propose to learn in this setting Factored Upper Confidence Bound (F-UCB)
- F-UCB performs a **UCB-like** exploration (Auer et al., 2002) **independently** for every dimension $i \in [\![d]\!]$:

$$\mathbf{a}(t) \in \underset{(a_1,\dots,a_d)^T \in \mathcal{A}}{\arg\max} \prod_{i \in [\![d]\!]} \mathrm{UCB}_{i,a_i}(t)$$

where:

$$\mathrm{UCB}_{i,a_i}(t) = \widehat{\mu}_{i,a_i}(t-1) + \sigma \sqrt{\frac{\alpha \log t}{N_{i,a_i}(t-1)}}$$

- F-UCB enjoys **worst-case optimal guarantees**:

$$\mathbb{E}\left[R_T(\texttt{F-UCB}, \boldsymbol{\nu})\right] \leq \widetilde{\mathcal{O}}(\sigma \sum_{i \in [\![d]\!]} \sqrt{k_i T})$$

- We will pull **at most**:

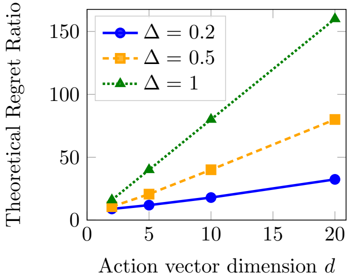
$$\mathbb{E}[N_{i,j}] \leq \frac{4\alpha\sigma^2 \log T}{\Delta_{i,j}^2}$$

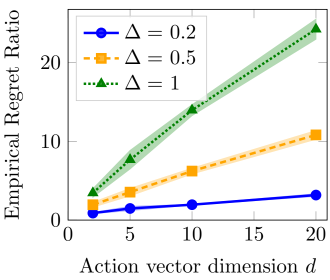
  times **every suboptimal arm**

- To get an instance-dependent upper bound on the expected regret, we search for the **worst combination** of the pulls

- We can find the solution as a **Linear Programming** problem

For $T \to +\infty$, we observe that $\dfrac{\texttt{F-UCB}\text{ Upper Bound}}{\text{Lower Bound}} \leq \dfrac{2\alpha d \Delta}{1-(1-\Delta)^d} \overset{\Delta \to 1}{=} \mathcal{O}(d)$





- **F-UCB** does not enjoy instance-depedent optimality due to the **lack of syncronization** over the components of the action vector
- To overcome this problem, we have to plan an **optimal sequence of actions**
- We propose `F-Track`, an algorithm which **tracks the lower bound** (Lattimore and Szepesvari, 2017)

`F-Track` **coordinates among the d dimensions** in three phases:

1. **Warm-up**: Play action vectors in **round robin** until every action component has been pulled **at least a minimum amount of times**
2. **LB Matching**: Use warm-up data to compute estimates of $\hat{\mu}_{i,j}$ and $\hat{\Delta}_{i,j}$. Solve the lower bound (efficient) LP to **find an optimal pull schedule**
3. **Recovery**: If, during phase 2, the **estimation error** of any $\hat{\mu}_{i,j}$ goes above a threshold, the scheduling is invalidated and the algorithm **falls back to `F-UCB`** until T

This algorithm is **asymptotically instance-dependent optimal**

Future works on FRB should:

- Consider **alternative functions** w.r.t. the product
- Design an optimal algorithm with **both instance-dependent and worst-case optimal guarantees**

Future Works on Pricing and Advertising

- **Pricing**: consider **positive** and **negative interactions** among products
- **Advertising**: consider a **non-linear tractable structure** for modeling MMMs
- **Joint Pricing & Advertising**: integrate complex dynamics

Thank you!

Åström, K. J. (1965). Optimal control of markov processes with incomplete state information. *Journal of Mathematical Analysis and Applications*, 10(1):174–205.

Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2):235–256.

Bacchiocchi, F., Genalti, G., Maran, D., Mussi, M., Restelli, M., Gatti, N., and Metelli, A. M. (2024). Autoregressive bandits. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 238 of *Proceedings of Machine Learning Research*, pages 937–945. PMLR.

Lattimore, T. and Szepesvari, C. (2017). The end of optimism? an asymptotic analysis of finite-armed linear bandits. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 728–737. PMLR.

Lattimore, T. and Szepesvári, C. (2020). *Bandit algorithms*. Cambridge University Press.

Luttinger, J. and Friedberg, R. (1976). A new rearrangement inequality for multiple integrals. *Archive for Rational Mechanics and Analysis*, 61:45–64.

Mussi, M., Drago, S., Restelli, M., and Metelli, A. M. (2024). Factored-reward bandits with intermediate observations. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*. PMLR.

Mussi, M., Genalti, G., Trovò, F., Nuara, A., Gatti, N., and Restelli, M. (2022). Pricing the long tail by explainable product aggregation and monotonic bandits. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3623–3633.

Mussi, M., Metelli, A. M., and Restelli, M. (2023). Dynamical linear bandits. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pages 25563–25587. PMLR.

Nuara, A., Trovò, F., Gatti, N., and Restelli, M. (2022). Online joint bid/daily budget optimization of internet advertising campaigns. *Artificial Intelligence*, 305:103663.

Rothschild, M. (1974). A two-armed bandit theory of market pricing. *Journal of Economic Theory*, 9(2):185–202.