Online Combinatorial Optimization under Bandit Feedback

M. Sadegh Talebi *

*Department of Automatic Control
KTH The Royal Institute of Technology

February 2016

Combinatorial Optimization

- Decision space $\mathcal{M} \subset \{0,1\}^d$
  - Each decision $M \in \mathcal{M}$ is a binary d-dimensional vector.
  - Combinatorial structure, e.g., matchings, spanning trees, fixed-size subsets, graph cuts, paths
- Weights $\theta \in \mathbb{R}^d$
- Generic combinatorial (linear) optimization

$$\text{maximize} \quad M^\top \theta = \sum_{i=1}^{d} M_i \theta_i$$
$$\text{over} \quad M \in \mathcal{M}$$

- Sequential decision making over T rounds

Combinatorial Optimization

- Decision space $\mathcal{M} \subset \{0,1\}^d$
  - Each decision $M \in \mathcal{M}$ is a binary d-dimensional vector.
  - Combinatorial structure, e.g., matchings, spanning trees, fixed-size subsets, graph cuts, paths
- Weights $\theta \in \mathbb{R}^d$
- Generic combinatorial (linear) optimization

$$\text{maximize} \quad M^\top \theta = \sum_{i=1}^{d} M_i \theta_i$$
$$\text{over} \quad M \in \mathcal{M}$$

- Sequential decision making over T rounds

Combinatorial Optimization under Uncertainty

Sequential decision making over T rounds

- Known $\theta \implies$ always select $M^\star := \mathrm{argmax}_{M \in \mathcal{M}} M^\top \theta$.
- Weights θ could be initially unknown or unpredictably varying.
- At time n, environment chooses a reward vector $X(n) \in \mathbb{R}^d$
  - Stochastic: $X(n)$ i.i.d., $\mathbb{E}[X(n)] = \theta$.
  - Adversarial: $X(n)$ chosen beforehand by an adversary.
- Selecting M gives reward $M^\top X(n) = \sum_{i=1}^d M_i X_i(n)$.

Sequential Learning: at each step n, select $M(n) \in \mathcal{M}$ based on the previous decisions and observed rewards

Combinatorial Optimization under Uncertainty

Sequential decision making over T rounds

- Known $\theta \implies$ always select $M^\star := \mathrm{argmax}_{M \in \mathcal{M}} M^\top \theta$.
- Weights θ could be initially unknown or unpredictably varying.
- At time n, environment chooses a reward vector $X(n) \in \mathbb{R}^d$
  - Stochastic: $X(n)$ i.i.d., $\mathbb{E}[X(n)] = \theta$.
  - Adversarial: $X(n)$ chosen beforehand by an adversary.
- Selecting M gives reward $M^\top X(n) = \sum_{i=1}^d M_i X_i(n)$.

Sequential Learning: at each step n, select $M(n) \in \mathcal{M}$ based on the previous decisions and observed rewards

Combinatorial Optimization under Uncertainty

Sequential decision making over T rounds

- Known $\theta \implies$ always select $M^\star := \operatorname{argmax}_{M \in \mathcal{M}} M^\top \theta$.
- Weights θ could be initially unknown or unpredictably varying.
- At time n, environment chooses a reward vector $X(n) \in \mathbb{R}^d$
  - Stochastic: $X(n)$ i.i.d., $\mathbb{E}[X(n)] = \theta$.
  - Adversarial: $X(n)$ chosen beforehand by an adversary.
- Selecting M gives reward $M^\top X(n) = \sum_{i=1}^d M_i X_i(n)$.

Sequential Learning: at each step n, select $M(n) \in \mathcal{M}$ based on the previous decisions and observed rewards

Combinatorial Optimization under Uncertainty

Sequential decision making over T rounds

- Known $\theta \implies$ always select $M^\star := \mathrm{argmax}_{M \in \mathcal{M}} M^\top \theta$.
- Weights θ could be initially unknown or unpredictably varying.
- At time n, environment chooses a reward vector $X(n) \in \mathbb{R}^d$
  - Stochastic: $X(n)$ i.i.d., $\mathbb{E}[X(n)] = \theta$.
  - Adversarial: $X(n)$ chosen beforehand by an adversary.
- Selecting M gives reward $M^\top X(n) = \sum_{i=1}^d M_i X_i(n)$.

Sequential Learning: at each step n, select $M(n) \in \mathcal{M}$ based on the previous decisions and observed rewards

Combinatorial Optimization under Uncertainty

Sequential decision making over T rounds

- Known $\theta \implies$ always select $M^\star := \operatorname{argmax}_{M \in \mathcal{M}} M^\top \theta$.
- Weights θ could be initially unknown or unpredictably varying.
- At time n, environment chooses a reward vector $X(n) \in \mathbb{R}^d$
  - Stochastic: $X(n)$ i.i.d., $\mathbb{E}[X(n)] = \theta$.
  - Adversarial: $X(n)$ chosen beforehand by an adversary.
- Selecting M gives reward $M^\top X(n) = \sum_{i=1}^d M_i X_i(n)$.

Sequential Learning: at each step n, select $M(n) \in \mathcal{M}$ based on the previous decisions and observed rewards

Regret

- **Goal:** Maximize collected rewards in expectation

$$\mathbb{E}\left[\sum_{n=1}^{T} M(n)^\top X(n)\right].$$

- Or equivalently, minimize **regret** over T rounds:

$$R(T) = \underbrace{\max_{M \in \mathcal{M}} \mathbb{E}\left[\sum_{n=1}^{T} M^\top X(n)\right]}_{\text{oracle}} - \underbrace{\mathbb{E}\left[\sum_{n=1}^{T} M(n)^\top X(n)\right]}_{\text{your algorithm}}.$$

- Quantifies cumulative loss of not choosing the best decision (in hindsight).
- Algorithm is learning iff $R(T) = o(T)$.

Regret

- **Goal:** Maximize collected rewards in expectation

$$\mathbb{E}\left[\sum_{n=1}^{T} M(n)^\top X(n)\right].$$

- Or equivalently, minimize **regret** over T rounds:

$$R(T) = \underbrace{\max_{M \in \mathcal{M}} \mathbb{E}\left[\sum_{n=1}^{T} M^\top X(n)\right]}_{\text{oracle}} - \underbrace{\mathbb{E}\left[\sum_{n=1}^{T} M(n)^\top X(n)\right]}_{\text{your algorithm}}.$$

- Quantifies cumulative loss of not choosing the best decision (in hindsight).
- Algorithm is learning iff $R(T) = o(T)$.

Regret

- **Goal:** Maximize collected rewards in expectation

$$\mathbb{E}\left[\sum_{n=1}^{T} M(n)^\top X(n)\right].$$

- Or equivalently, minimize **regret** over T rounds:

$$R(T) = \underbrace{\max_{M\in\mathcal{M}} \mathbb{E}\left[\sum_{n=1}^{T} M^\top X(n)\right]}_{\text{oracle}} - \underbrace{\mathbb{E}\left[\sum_{n=1}^{T} M(n)^\top X(n)\right]}_{\text{your algorithm}}.$$

- Quantifies cumulative loss of not choosing the best decision (in hindsight).
- Algorithm is learning iff $R(T) = o(T)$.

Feedback

Choose $M(n)$ based on previous decisions and observed feedback

- Full information: $X(n)$ is revealed.
- Semi-bandit feedback: $X_i(n)$ is revealed iff $M_i(n) = 1$.
- Bandit feedback: only the reward $M(n)^\top X(n)$ is revealed.

Sequential learning is modeled as a Multi-Armed Bandit (MAB) problem.

Combinatorial MAB:

$$\text{Decision } M \in \mathcal{M} \iff \text{Arm}$$
$$\text{Element } \{1, \dots, d\} \iff \text{Basic action}$$

Each arm is composed of several basic actions.

Feedback

Choose $M(n)$ based on previous decisions and observed feedback

- Full information: $X(n)$ is revealed.
- Semi-bandit feedback: $X_i(n)$ is revealed iff $M_i(n) = 1$.
- Bandit feedback: only the reward $M(n)^\top X(n)$ is revealed.

Sequential learning is modeled as a Multi-Armed Bandit (MAB) problem.

Combinatorial MAB:

$$\text{Decision } M \in \mathcal{M} \iff \text{Arm}$$
$$\text{Element } \{1, \dots, d\} \iff \text{Basic action}$$

Each arm is composed of several basic actions.

Feedback

Choose $M(n)$ based on previous decisions and observed feedback

- Full information: $X(n)$ is revealed.
- Semi-bandit feedback: $X_i(n)$ is revealed iff $M_i(n) = 1$.
- Bandit feedback: only the reward $M(n)^\top X(n)$ is revealed.

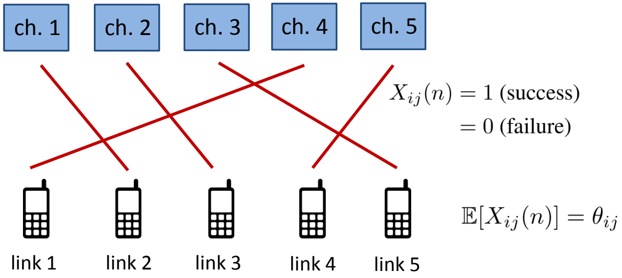
Sequential learning is modeled as a Multi-Armed Bandit (MAB) problem.

Combinatorial MAB:

$$\text{Decision } M \in \mathcal{M} \iff \text{Arm}$$
$$\text{Element } \{1, \ldots, d\} \iff \text{Basic action}$$

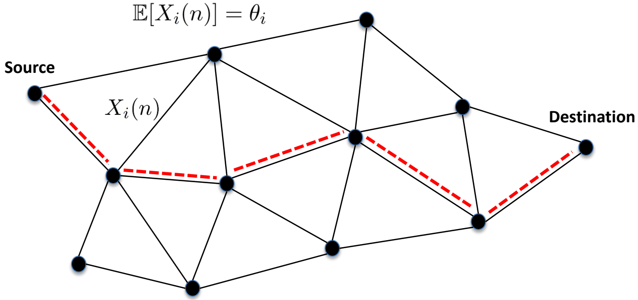
Each arm is composed of several basic actions.

Application 1: Spectrum Sharing



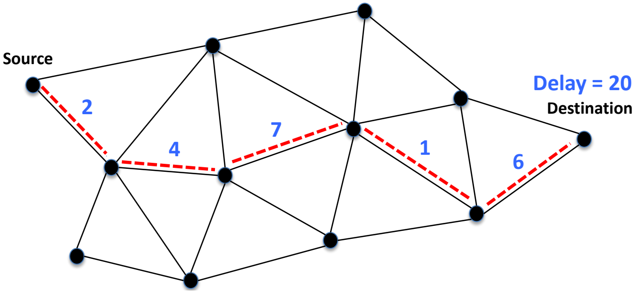
- K channels, L links
- $\mathcal{M} \equiv$ the set of matchings from $[L]$ to $[K]$
- $\theta_{ij} \equiv$ data rate on the connection (link-i, channel-j)
- $X_{ij}(n) \equiv$ success/failure indicator for transmission of link i on channel j

Application 2: Shortest-path Routing



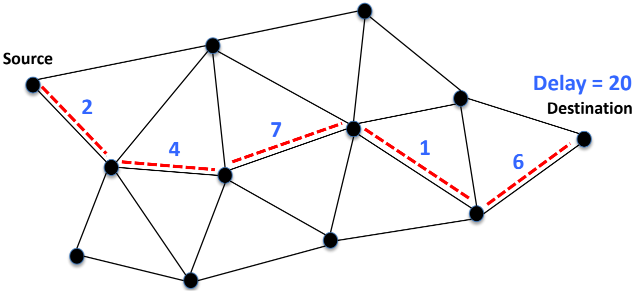
- $\mathcal{M} \equiv$ the set of paths
- $\theta_i \equiv$ average transmission delay on link i
- $X_i(n) \equiv$ transmission delay of link i for n-th packet

Application 2: Shortest-path Routing



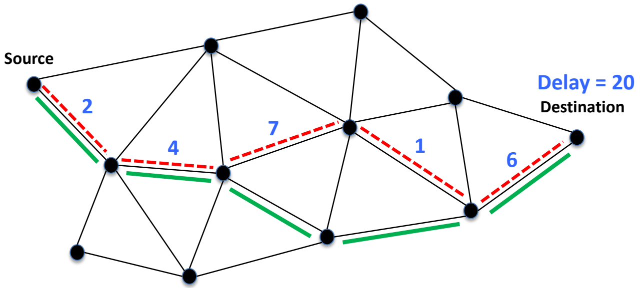
- Semi-bandit feedback: $(2, 4, 7, 1, 6)$ are revealed for chosen links (red).
- Bandit feedback: 20 is revealed for the chosen *path*.

Application 2: Shortest-path Routing



- Semi-bandit feedback: $\color{red}{(2, 4, 7, 1, 6)}$ are revealed for chosen links (red).
- Bandit feedback: 20 is revealed for the chosen *path*.

Application 2: Shortest-path Routing



- Semi-bandit feedback: $(2, 4, 7, 1, 6)$ are revealed for chosen links (red).
- Bandit feedback: 20 is revealed for the chosen *path*.

Exploiting Combinatorial Structure

- Classical MAB ($\mathcal{M}$ set of singletons; $|\mathcal{M}| = d$):
  - Stochastic $R(T) \sim |\mathcal{M}| \log(T)$
  - Adversarial $R(T) \sim \sqrt{|\mathcal{M}|T}$
- Generic combinatorial $\mathcal{M}$
  - $|\mathcal{M}|$ could grow exponentially in $d \Longrightarrow$ prohibitive regret
  - Arms are correlated; they share basic actions.

  $\Longrightarrow$ exploit combinatorial structure in $\mathcal{M}$ to get $R(T) \sim C \log(T)$ or $R(T) \sim \sqrt{CT}$ where $C \ll |\mathcal{M}|$

How much can we reduce the regret by exploiting the combinatorial structure of $\mathcal{M}$?
How to optimally do so?

Exploiting Combinatorial Structure

- Classical MAB ($\mathcal{M}$ set of singletons; $|\mathcal{M}| = d$):
  - Stochastic $R(T) \sim |\mathcal{M}| \log(T)$
  - Adversarial $R(T) \sim \sqrt{|\mathcal{M}|T}$
- Generic combinatorial $\mathcal{M}$
  - $|\mathcal{M}|$ could grow exponentially in $d \implies$ prohibitive regret
  - Arms are correlated; they share basic actions.

  $\implies$ exploit combinatorial structure in $\mathcal{M}$ to get $R(T) \sim C \log(T)$ or $R(T) \sim \sqrt{CT}$ where $C \ll |\mathcal{M}|$

How much can we reduce the regret by exploiting the combinatorial structure of $\mathcal{M}$?
How to optimally do so?

Exploiting Combinatorial Structure

- Classical MAB ($\mathcal{M}$ set of singletons; $|\mathcal{M}| = d$):
  - Stochastic $R(T) \sim |\mathcal{M}| \log(T)$
  - Adversarial $R(T) \sim \sqrt{|\mathcal{M}|T}$
- Generic combinatorial $\mathcal{M}$
  - $|\mathcal{M}|$ could grow exponentially in $d \implies$ prohibitive regret
  - Arms are correlated; they share basic actions.

  $\implies$ exploit combinatorial structure in $\mathcal{M}$ to get $R(T) \sim C \log(T)$ or $R(T) \sim \sqrt{CT}$ where $C \ll |\mathcal{M}|$

How much can we reduce the regret by exploiting the combinatorial structure of $\mathcal{M}$?
How to optimally do so?

Exploiting Combinatorial Structure

- Classical MAB ($\mathcal{M}$ set of singletons; $|\mathcal{M}| = d$):
  - Stochastic $R(T) \sim |\mathcal{M}| \log(T)$
  - Adversarial $R(T) \sim \sqrt{|\mathcal{M}|T}$
- Generic combinatorial $\mathcal{M}$
  - $|\mathcal{M}|$ could grow exponentially in $d \implies$ prohibitive regret
  - Arms are correlated; they share basic actions.

  $\implies$ exploit combinatorial structure in $\mathcal{M}$ to get $R(T) \sim C \log(T)$ or $R(T) \sim \sqrt{CT}$ where $C \ll |\mathcal{M}|$

How much can we reduce the regret by exploiting the combinatorial structure of $\mathcal{M}$?
How to optimally do so?

Map of Thesis

How much can we reduce the regret by exploiting the combinatorial structure of $\mathcal{M}$?
How to optimally do so?

Chapter	**Combinatorial Structure** $\mathcal{M}$	**Reward** X
Ch. 3	Generic	Bernoulli
Ch. 4	Matroid	Bernoulli
Ch. 5	Generic	Geometric
Ch. 6	Generic (with fixed cardinality)	Adversarial

Outline

Outline

Stochastic CMABs

Rewards:

- $X(n)$ i.i.d. , Bernoulli distributed with $\mathbb{E}[X(n)] = \theta \in [0,1]^d$
- $X_i(n),\ i \in [d]$ are independent across i
- $\mu_M := M^\top \theta$ average reward of arm M
- Average reward gap $\Delta_M = \mu^\star - \mu_M$
- Optimality gap $\Delta_{\min} = \min_{M \neq M^\star} \Delta_M$

Algorithm	Regret
LLR (Gai et al., 2012)	$\mathcal{O}\left(\frac{m^4 d}{\Delta_{\min}^2} \log(T)\right)$
CUCB (Chen et al., 2013)	$\mathcal{O}\left(\frac{m^2 d}{\Delta_{\min}} \log(T)\right)$
CUCB (Kveton et al., 2015)	$\mathcal{O}\left(\frac{md}{\Delta_{\min}} \log(T)\right)$
ESCB	$\mathcal{O}\left(\frac{\sqrt{m} d}{\Delta_{\min}} \log(T)\right)$

m = maximal cardinality of arms

Stochastic CMABs

Rewards:

- $X(n)$ i.i.d. , Bernoulli distributed with $\mathbb{E}[X(n)] = \theta \in [0,1]^d$
- $X_i(n),\ i \in [d]$ are independent across i
- $\mu_M := M^\top \theta$ average reward of arm M
- Average reward gap $\Delta_M = \mu^\star - \mu_M$
- Optimality gap $\Delta_{\min} = \min_{M \neq M^\star} \Delta_M$

Algorithm	Regret
LLR (Gai et al., 2012)	$\mathcal{O}\left(\frac{m^4 d}{\Delta_{\min}^2} \log(T)\right)$
CUCB (Chen et al., 2013)	$\mathcal{O}\left(\frac{m^2 d}{\Delta_{\min}} \log(T)\right)$
CUCB (Kveton et al., 2015)	$\mathcal{O}\left(\frac{md}{\Delta_{\min}} \log(T)\right)$
ESCB	$\mathcal{O}\left(\frac{\sqrt{m}d}{\Delta_{\min}} \log(T)\right)$

m = maximal cardinality of arms

Stochastic CMABs

Rewards:

- $X(n)$ i.i.d. , Bernoulli distributed with $\mathbb{E}[X(n)] = \theta \in [0,1]^d$
- $X_i(n),\ i \in [d]$ are independent across i
- $\mu_M := M^\top \theta$ average reward of arm M
- Average reward gap $\Delta_M = \mu^\star - \mu_M$
- Optimality gap $\Delta_{\min} = \min_{M \neq M^\star} \Delta_M$

Algorithm	Regret
LLR (Gai et al., 2012)	$\mathcal{O}\left(\frac{m^4 d}{\Delta_{\min}^2} \log(T)\right)$
CUCB (Chen et al., 2013)	$\mathcal{O}\left(\frac{m^2 d}{\Delta_{\min}} \log(T)\right)$
CUCB (Kveton et al., 2015)	$\mathcal{O}\left(\frac{m d}{\Delta_{\min}} \log(T)\right)$
ESCB	$\mathcal{O}\left(\frac{\sqrt{m} d}{\Delta_{\min}} \log(T)\right)$

m = maximal cardinality of arms

Stochastic CMABs

Rewards:

- $X(n)$ i.i.d. , Bernoulli distributed with $\mathbb{E}[X(n)] = \theta \in [0,1]^d$
- $X_i(n),\ i \in [d]$ are independent across i
- $\mu_M := M^\top \theta$ average reward of arm M
- Average reward gap $\Delta_M = \mu^\star - \mu_M$
- Optimality gap $\Delta_{\min} = \min_{M \neq M^\star} \Delta_M$

Algorithm	Regret
LLR (Gai et al., 2012)	$\mathcal{O}\left(\frac{m^4 d}{\Delta_{\min}^2} \log(T)\right)$
CUCB (Chen et al., 2013)	$\mathcal{O}\left(\frac{m^2 d}{\Delta_{\min}} \log(T)\right)$
CUCB (Kveton et al., 2015)	$\mathcal{O}\left(\frac{md}{\Delta_{\min}} \log(T)\right)$
ESCB	$\mathcal{O}\left(\frac{\sqrt{m}d}{\Delta_{\min}} \log(T)\right)$

m = maximal cardinality of arms

Optimism in the face of uncertainty

- Construct a confidence bound $[b^-, b^+]$ for (unknown) μ s.t.

$$\mu \in [b^-, b^+] \quad \text{with high probability}$$

- Maximization problem $\Longrightarrow$ we replace (unknown) μ by b^+, its Upper Confidence Bound (UCB) index.

"Optimism in the face of uncertainty" principle:
Choose arm M with the highest UCB index

Algorithm based on optimistic principle:

- For arm M and time n, find confidence interval for μ_M:

$$\mathbb{P}\left[\mu_M \in [b_M^-(n),\ b_M^+(n)]\right] \geq 1 - \mathcal{O}\left(\frac{1}{n\log(n)}\right)$$

- Choose $M(n) \in \operatorname{argmax}_{M \in \mathcal{M}} b_M^+(n)$

Optimism in the face of uncertainty

- Construct a confidence bound $[b^-, b^+]$ for (unknown) μ s.t.

$$\mu \in [b^-, b^+] \quad \text{with high probability}$$

- Maximization problem $\Longrightarrow$ we replace (unknown) μ by b^+, its Upper Confidence Bound (UCB) index.

"Optimism in the face of uncertainty" principle:
Choose arm M with the highest UCB index

Algorithm based on optimistic principle:

- For arm M and time n, find confidence interval for μ_M:

$$\mathbb{P}\left[\mu_M \in [b_M^-(n),\ b_M^+(n)]\right] \geq 1 - \mathcal{O}\left(\frac{1}{n\log(n)}\right)$$

- Choose $M(n) \in \mathrm{argmax}_{M \in \mathcal{M}}\, b_M^+(n)$

Optimism in the face of uncertainty

- Construct a confidence bound $[b^-, b^+]$ for (unknown) μ s.t.

$$\mu \in [b^-, b^+] \quad \text{with high probability}$$

- Maximization problem $\Longrightarrow$ we replace (unknown) μ by b^+, its Upper Confidence Bound (UCB) index.

> "Optimism in the face of uncertainty" principle:
> Choose arm M with the highest UCB index

Algorithm based on optimistic principle:

- For arm M and time n, find confidence interval for μ_M:

$$\mathbb{P}\left[\mu_M \in [b_M^-(n),\ b_M^+(n)]\right] \geq 1 - \mathcal{O}\left(\frac{1}{n\log(n)}\right)$$

- Choose $M(n) \in \mathrm{argmax}_{M \in \mathcal{M}}\, b_M^+(n)$

Optimism in the face of uncertainty

- Construct a confidence bound $[b^-, b^+]$ for (unknown) μ s.t.

$$\mu \in [b^-, b^+] \quad \text{with high probability}$$

- Maximization problem $\Longrightarrow$ we replace (unknown) μ by b^+, its Upper Confidence Bound (UCB) index.

> "Optimism in the face of uncertainty" principle:
> Choose arm M with the highest UCB index

Algorithm based on optimistic principle:

- For arm M and time n, find confidence interval for μ_M:

$$\mathbb{P}\left[\mu_M \in [b_M^-(n),\ b_M^+(n)]\right] \geq 1 - \mathcal{O}\left(\frac{1}{n\log(n)}\right)$$

- Choose $M(n) \in \operatorname{argmax}_{M\in\mathcal{M}} b_M^+(n)$

Optimism in the face of uncertainty

- Construct a confidence bound $[b^-, b^+]$ for (unknown) μ s.t.

$$\mu \in [b^-, b^+] \quad \text{with high probability}$$

- Maximization problem $\Longrightarrow$ we replace (unknown) μ by b^+, its Upper Confidence Bound (UCB) index.

> "Optimism in the face of uncertainty" principle:
> Choose arm M with the highest UCB index

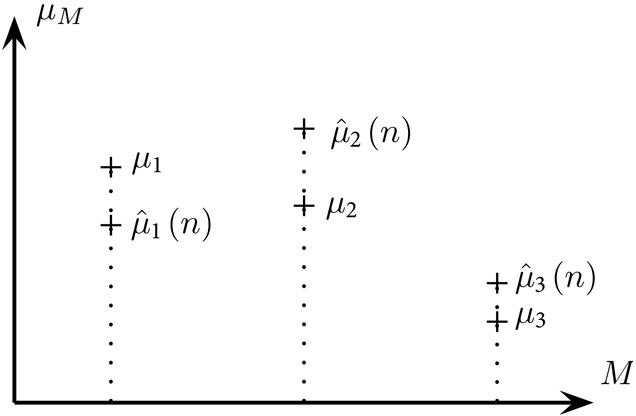
Algorithm based on optimistic principle:

- For arm M and time n, find confidence interval for μ_M:

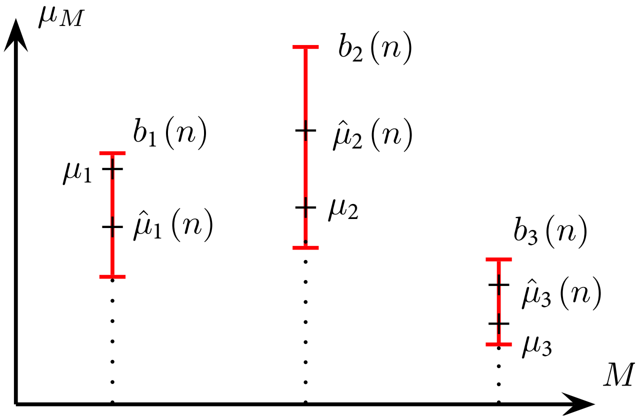
$$\mathbb{P}\left[\mu_M \in [b_M^-(n),\ b_M^+(n)]\right] \geq 1 - \mathcal{O}\left(\frac{1}{n\log(n)}\right)$$

- Choose $M(n) \in \mathrm{argmax}_{M \in \mathcal{M}}\, b_M^+(n)$

Optimistic Principle

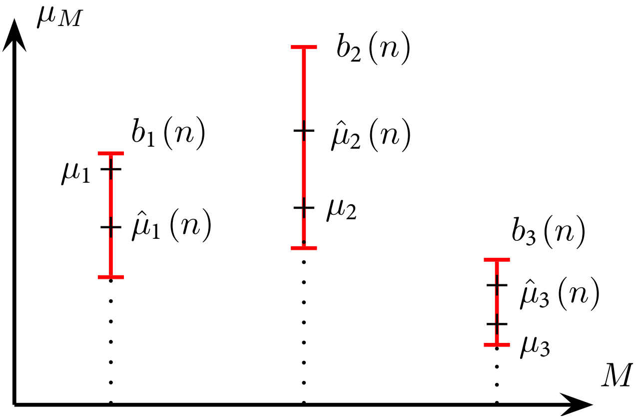


Optimistic Principle



How to construct the index for arms?

Optimistic Principle



How to construct the index for arms?

Index Construction

- **Naive approach:** construct index for basic actions
  $\Longrightarrow$ index of arm M = sum of indexes of basic action in arm M
- Empirical mean $\hat{\theta}_i(n)$, number of observations: $t_i(n)$.
- Hoeffding's inequality:

$$\mathbb{P}\left[\theta_i \in \left(\hat{\theta}_i(n) - \sqrt{\frac{\log(1/\delta)}{2t_i(n)}},\ \hat{\theta}_i(n) + \sqrt{\frac{\log(1/\delta)}{2t_i(n)}}\right)\right] \geq 1 - 2\delta$$

- Choose $\delta = \frac{1}{n^3}$

$$\text{Index:} \quad b_M(n) = \underbrace{\sum_{i=1}^{d} M_i \hat{\theta}_i(n)}_{\hat{\mu}_M(n)} + \underbrace{\sum_{i=1}^{d} M_i \sqrt{\frac{3\log(n)}{2t_i(n)}}}_{\text{confidence radius}}.$$

$$\mu_M \in \left[\hat{\mu}_M(n) - \sum_{i=1}^{d} M_i \sqrt{\frac{3\log(n)}{2t_i(n)}},\ \hat{\mu}_M(n) + \sum_{i=1}^{d} M_i \sqrt{\frac{3\log(n)}{2t_i(n)}}\right] \quad \text{w.p.} \quad \geq 1 - \frac{1}{n^3}$$

Index Construction

- **Naive approach:** construct index for basic actions
  $\Longrightarrow$ index of arm M = sum of indexes of basic action in arm M
- Empirical mean $\hat{\theta}_i(n)$, number of observations: $t_i(n)$.
- Hoeffding's inequality:

$$\mathbb{P}\left[\theta_i \in \left(\hat{\theta}_i(n) - \sqrt{\frac{\log(1/\delta)}{2t_i(n)}},\ \hat{\theta}_i(n) + \sqrt{\frac{\log(1/\delta)}{2t_i(n)}}\right)\right] \geq 1 - 2\delta$$

- Choose $\delta = \frac{1}{n^3}$

Index: $$b_M(n) = \underbrace{\sum_{i=1}^{d} M_i \hat{\theta}_i(n)}_{\hat{\mu}_M(n)} + \underbrace{\sum_{i=1}^{d} M_i \sqrt{\frac{3\log(n)}{2t_i(n)}}}_{\text{confidence radius}}.$$

$$\mu_M \in \left[\hat{\mu}_M(n) - \sum_{i=1}^{d} M_i \sqrt{\frac{3\log(n)}{2t_i(n)}},\ \hat{\mu}_M(n) + \sum_{i=1}^{d} M_i \sqrt{\frac{3\log(n)}{2t_i(n)}}\right] \quad \text{w.p.} \quad \geq 1 - \frac{1}{n^3}$$

Index Construction

- **Naive approach:** construct index for basic actions
  $\Longrightarrow$ index of arm M = sum of indexes of basic action in arm M
- Empirical mean $\hat{\theta}_i(n)$, number of observations: $t_i(n)$.
- Hoeffding's inequality:

$$\mathbb{P}\left[\theta_i \in \left(\hat{\theta}_i(n) - \sqrt{\frac{\log(1/\delta)}{2t_i(n)}},\ \hat{\theta}_i(n) + \sqrt{\frac{\log(1/\delta)}{2t_i(n)}}\right)\right] \geq 1 - 2\delta$$

- Choose $\delta = \frac{1}{n^3}$

$$\text{Index:} \quad b_M(n) = \underbrace{\sum_{i=1}^{d} M_i \hat{\theta}_i(n)}_{\hat{\mu}_M(n)} + \underbrace{\sum_{i=1}^{d} M_i \sqrt{\frac{3\log(n)}{2t_i(n)}}}_{\text{confidence radius}}.$$

$$\mu_M \in \left[\hat{\mu}_M(n) - \sum_{i=1}^{d} M_i \sqrt{\frac{3\log(n)}{2t_i(n)}},\ \hat{\mu}_M(n) + \sum_{i=1}^{d} M_i \sqrt{\frac{3\log(n)}{2t_i(n)}}\right] \quad \text{w.p.} \quad \geq 1 - \frac{1}{n^3}$$

Index Construction

- **Naive approach:** construct index for basic actions
  $\Longrightarrow$ index of arm M = sum of indexes of basic action in arm M
- Empirical mean $\hat{\theta}_i(n)$, number of observations: $t_i(n)$.
- Hoeffding's inequality:

$$\mathbb{P}\left[\theta_i \in \left(\hat{\theta}_i(n) - \sqrt{\frac{\log(1/\delta)}{2t_i(n)}},\ \hat{\theta}_i(n) + \sqrt{\frac{\log(1/\delta)}{2t_i(n)}}\right)\right] \geq 1 - 2\delta$$

- Choose $\delta = \frac{1}{n^3}$

$$\text{Index:}\quad b_M(n) = \underbrace{\sum_{i=1}^{d} M_i \hat{\theta}_i(n)}_{\hat{\mu}_M(n)} + \underbrace{\sum_{i=1}^{d} M_i \sqrt{\frac{3\log(n)}{2t_i(n)}}}_{\text{confidence radius}}.$$

$$\mu_M \in \left[\hat{\mu}_M(n) - \sum_{i=1}^{d} M_i \sqrt{\frac{3\log(n)}{2t_i(n)}},\ \hat{\mu}_M(n) + \sum_{i=1}^{d} M_i \sqrt{\frac{3\log(n)}{2t_i(n)}}\right] \quad \text{w.p.} \quad \geq 1 - \frac{1}{n^3}$$

Index Construction

- **Our approach:** constructing confidence interval directly for each arm M
- Motivated by concentration for sum of empirical KL-divergences.
- For a given δ, consider a set

$$B = \left\{ \lambda \in [0,1]^d : \sum_{i=1}^{d} t_i(n)\, \mathrm{kl}(\hat{\theta}_i(n), \lambda_i) \le \log(1/\delta) \right\}$$

with

$$\mathrm{kl}(u,v) := u \log \frac{u}{v} + (1-u) \log \frac{1-u}{1-v}.$$

Find an upper confidence bound for μ_M such that

$$\mu_M \in \left[\times, \ \ M^\top \lambda \right] \text{ w.p. at least } 1-\delta, \ \ \forall \lambda \in B.$$

Equivalently,

$$\mu_M \le \max_{\lambda \in B} M^\top \lambda \text{ w.p. at least } 1-\delta.$$

Index Construction

- **Our approach:** constructing confidence interval directly for each arm M
- Motivated by concentration for sum of empirical KL-divergences.
- For a given δ, consider a set

$$B = \left\{ \lambda \in [0,1]^d : \sum_{i=1}^{d} t_i(n)\, \mathrm{kl}(\hat{\theta}_i(n), \lambda_i) \leq \log(1/\delta) \right\}$$

with

$$\mathrm{kl}(u,v) := u \log \frac{u}{v} + (1-u) \log \frac{1-u}{1-v}.$$

Find an upper confidence bound for μ_M such that

$$\mu_M \in \left[\times, \;\; M^\top \lambda \right] \text{ w.p. at least } 1-\delta, \;\; \forall \lambda \in B.$$

Equivalently,

$$\mu_M \leq \max_{\lambda \in B} M^\top \lambda \text{ w.p. at least } 1-\delta.$$

Index Construction

- **Our approach:** constructing confidence interval directly for each arm M
- Motivated by concentration for sum of empirical KL-divergences.
- For a given δ, consider a set

$$B = \left\{ \lambda \in [0,1]^d : \sum_{i=1}^{d} t_i(n)\, \mathrm{kl}(\hat{\theta}_i(n), \lambda_i) \leq \log(1/\delta) \right\}$$

with

$$\mathrm{kl}(u,v) := u \log \frac{u}{v} + (1-u) \log \frac{1-u}{1-v}.$$

Find an upper confidence bound for μ_M such that

$$\mu_M \in \left[\times,\ \ M^\top \lambda\right] \text{ w.p. at least } 1-\delta,\ \ \forall \lambda \in B.$$

Equivalently,

$$\mu_M \leq \max_{\lambda \in B} M^\top \lambda \text{ w.p. at least } 1-\delta.$$

Index Construction

- **Our approach:** constructing confidence interval directly for each arm M
- Motivated by concentration for sum of empirical KL-divergences.
- For a given δ, consider a set

$$B = \left\{ \lambda \in [0,1]^d : \sum_{i=1}^{d} t_i(n)\, \mathrm{kl}(\hat{\theta}_i(n), \lambda_i) \leq \log(1/\delta) \right\}$$

with

$$\mathrm{kl}(u,v) := u \log \frac{u}{v} + (1-u) \log \frac{1-u}{1-v}.$$

Find an upper confidence bound for μ_M such that

$$\mu_M \in \left[\times, \;\; M^\top \lambda \right] \text{ w.p. at least } 1-\delta, \;\; \forall \lambda \in B.$$

Equivalently,

$$\mu_M \leq \max_{\lambda \in B} M^\top \lambda \text{ w.p. at least } 1-\delta.$$

Proposed Indexes

Two new indexes:

- (1) Index b_M as the optimal value of the following problem:

$$\begin{aligned} b_M(n) &= \max_{\lambda \in [0,1]^d} \sum_{i=1}^{d} M_i \lambda_i \\ & \text{subject to :} \sum_{i=1}^{d} M_i t_i(n)\, \mathrm{kl}(\hat{\theta}_i(n), \lambda_i) \le \underbrace{f(n)}_{\log(1/\delta)}, \end{aligned}$$

with $f(n) = \log(n) + 4m \log(\log(n))$.

- b_M is computed by a line search (derived based on KKT conditions)
- Generalizes the KL-UCB index (Garivier & Cappé, 2011) to the case of combinatorial MABs
- (2) Index c_M:

$$c_M(n) = \hat{\mu}_M(n) + \sqrt{\frac{f(n)}{2} \sum_{i=1}^{d} \frac{M_i}{t_i(n)}}.$$

Proposed Indexes

Two new indexes:

- (1) Index b_M as the optimal value of the following problem:

$$\begin{aligned} b_M(n) &= \max_{\lambda \in [0,1]^d} \sum_{i=1}^{d} M_i \lambda_i \\ \text{subject to :} & \sum_{i=1}^{d} M_i t_i(n)\, \mathrm{kl}(\hat{\theta}_i(n), \lambda_i) \le \underbrace{f(n)}_{\log(1/\delta)}, \end{aligned}$$

  with $f(n) = \log(n) + 4m \log(\log(n))$.
- b_M is computed by a line search (derived based on KKT conditions)
- Generalizes the KL-UCB index (Garivier & Cappé, 2011) to the case of combinatorial MABs
- (2) Index c_M:

$$c_M(n) = \hat{\mu}_M(n) + \sqrt{\frac{f(n)}{2} \sum_{i=1}^{d} \frac{M_i}{t_i(n)}}.$$

Two new indexes:

- (1) Index b_M as the optimal value of the following problem:

$$\begin{aligned} b_M(n) \quad &= \quad \max_{\lambda \in [0,1]^d} \sum_{i=1}^{d} M_i \lambda_i \\ \text{subject to :} \quad &\sum_{i=1}^{d} M_i t_i(n)\, \mathrm{kl}(\hat{\theta}_i(n), \lambda_i) \le \underbrace{f(n)}_{\log(1/\delta)}, \end{aligned}$$

  with $f(n) = \log(n) + 4m \log(\log(n))$.

- b_M is computed by a line search (derived based on KKT conditions)
- Generalizes the KL-UCB index (Garivier & Cappé, 2011) to the case of combinatorial MABs
- (2) Index c_M:

$$c_M(n) = \hat{\mu}_M(n) + \sqrt{\frac{f(n)}{2} \sum_{i=1}^{d} \frac{M_i}{t_i(n)}}.$$

Proposed Indexes

Two new indexes:

- (1) Index b_M as the optimal value of the following problem:

$$\begin{aligned} b_M(n) &= \max_{\lambda \in [0,1]^d} \sum_{i=1}^{d} M_i \lambda_i \\ &\text{subject to : } \sum_{i=1}^{d} M_i t_i(n)\, \mathrm{kl}(\hat{\theta}_i(n), \lambda_i) \le \underbrace{f(n)}_{\log(1/\delta)}, \end{aligned}$$

  with $f(n) = \log(n) + 4m \log(\log(n))$.

- b_M is computed by a line search (derived based on KKT conditions)
- Generalizes the KL-UCB index (Garivier & Cappé, 2011) to the case of combinatorial MABs
- (2) Index c_M:

$$c_M(n) = \hat{\mu}_M(n) + \sqrt{\frac{f(n)}{2} \sum_{i=1}^{d} \frac{M_i}{t_i(n)}}.$$

Proposed Indexes

$$
\begin{aligned}
b_M(n) &= \max_{\lambda \in [0,1]^d} \sum_{i=1}^{d} M_i \lambda_i \\
&\quad \text{subject to}: \sum_{i=1}^{d} M_i t_i(n)\, \mathrm{kl}(\hat{\theta}_i(n), \lambda_i) \le f(n),
\end{aligned}
$$

$$
c_M(n) = \hat{\mu}_M(n) + \sqrt{\frac{f(n)}{2} \sum_{i=1}^{d} \frac{M_i}{t_i(n)}}.
$$

Theorem

For all $M \in \mathcal{M}$ and $n \ge 1$: $c_M(n) \ge b_M(n)$.

- Proof idea: Pinsker's inequality + Cauchy-Schwarz inequality

Proposed Indexes

$$
\begin{aligned}
b_M(n) &= \max_{\lambda \in [0,1]^d} \sum_{i=1}^{d} M_i \lambda_i \\
&\quad \text{subject to : } \sum_{i=1}^{d} M_i t_i(n) \, \mathrm{kl}(\hat{\theta}_i(n), \lambda_i) \le f(n),
\end{aligned}
$$

$$
c_M(n) = \hat{\mu}_M(n) + \sqrt{\frac{f(n)}{2} \sum_{i=1}^{d} \frac{M_i}{t_i(n)}}.
$$

Theorem

For all $M \in \mathcal{M}$ and $n \ge 1$: $c_M(n) \ge b_M(n)$.

- Proof idea: Pinsker's inequality + Cauchy-Schwarz inequality

ESCB Algorithm

ESCB $\equiv$ **Efficient Sampling for Combinatorial Bandits**

Algorithm 1 ESCB

for $n \geq 1$ **do**

Select arm $M(n) \in \mathrm{argmax}_{M \in \mathcal{M}} \zeta_M(n)$.

Observe the rewards, and update $t_i(n)$ and $\hat{\theta}_i(n), \forall i \in M(n)$.

end for

ESCB-1 if $\zeta_M = b_M$, ESCB-2 if $\zeta_M = c_M$.

Regret Analysis

Theorem

The regret under ESCB satisfies

$$R(T) \leq \frac{16d\sqrt{m}}{\Delta_{\min}} \log(T) + \mathcal{O}(\log(\log(T))).$$

- Proof idea
  - $c_M(n) \geq b_M(n) \geq \mu_M$ with high probability
  - Crucial concentration inequality (Magureanu et al., COLT 2014):

$$\mathbb{P}\left[\max_{n \leq T} \sum_{i=1}^{d} M_i t_i(n) \mathrm{kl}(\hat{\theta}_i(n), \theta_i) \geq \delta\right] \leq C_m (\log(T)\delta)^m e^{-\delta}.$$

Regret Analysis

Theorem

The regret under ESCB satisfies

$$R(T) \leq \frac{16d\sqrt{m}}{\Delta_{\min}} \log(T) + \mathcal{O}(\log(\log(T))).$$

- Proof idea
  - $c_M(n) \geq b_M(n) \geq \mu_M$ with high probability
  - Crucial concentration inequality (Magureanu et al., COLT 2014):

$$\mathbb{P}\left[\max_{n \leq T} \sum_{i=1}^{d} M_i t_i(n) \mathrm{kl}(\hat{\theta}_i(n), \theta_i) \geq \delta\right] \leq C_m (\log(T)\delta)^m e^{-\delta}.$$

Regret Lower Bound

How far are we from the optimal algorithm?

- Uniformly good algorithm π: $R^\pi(T) = \mathcal{O}(\log(T))$ for all θ.
- Notion of bad parameter: λ is bad if:
  - (i) it is statistically indistinguishable from true parameter θ (in the sense of KL-divergence) $\equiv$ reward distribution of optimal arm $M^\star$ is the same under θ or λ,
  - (ii) $M^\star$ is not optimal under λ.
- Set of all bad parameters $B(\theta)$:

$$B(\theta) = \left\{ \lambda \in [0,1]^d : \underbrace{(\lambda_i = \theta_i,\ \forall i \in M^\star)}_{\text{condition (i)}} \text{ and } \underbrace{\max_{M \in \mathcal{M}} M^\top \lambda > \mu^\star}_{\text{condition (ii)}} \right\}.$$

Regret Lower Bound

How far are we from the optimal algorithm?

- Uniformly good algorithm π: $R^\pi(T) = \mathcal{O}(\log(T))$ for all θ.
- Notion of bad parameter: λ is bad if:
  - (i) it is statistically indistinguishable from true parameter θ (in the sense of KL-divergence) $\equiv$ reward distribution of optimal arm $M^\star$ is the same under θ or λ,
  - (ii) $M^\star$ is not optimal under λ.
- Set of all bad parameters $B(\theta)$:

$$B(\theta) = \left\{ \lambda \in [0,1]^d : \underbrace{(\lambda_i = \theta_i, \ \forall i \in M^\star)}_{\text{condition (i)}} \text{ and } \underbrace{\max_{M \in \mathcal{M}} M^\top \lambda > \mu^\star}_{\text{condition (ii)}} \right\}.$$

Regret Lower Bound

How far are we from the optimal algorithm?

- Uniformly good algorithm π: $R^\pi(T) = \mathcal{O}(\log(T))$ for all θ.
- Notion of bad parameter: λ is bad if:
  - (i) it is statistically indistinguishable from true parameter θ (in the sense of KL-divergence) $\equiv$ reward distribution of optimal arm $M^\star$ is the same under θ or λ,
  - (ii) $M^\star$ is not optimal under λ.
- Set of all bad parameters $B(\theta)$:

$$B(\theta) = \Big\{\lambda \in [0,1]^d : \underbrace{(\lambda_i = \theta_i,\ \forall i \in M^\star)}_{\text{condition (i)}} \text{ and } \underbrace{\max_{M \in \mathcal{M}} M^\top \lambda > \mu^\star}_{\text{condition (ii)}}\Big\}.$$

Regret Lower Bound

Theorem

For any uniformly good algorithm π, $\liminf_{T\to\infty} \frac{R^\pi(T)}{\log(T)} \geq c(\theta)$, *with*

$$
\begin{aligned}
c(\theta) \;&=\; \inf_{x\in\mathbb{R}_+^{|\mathcal{M}|}} \sum_{M\in\mathcal{M}} \Delta_M x_M \\
&\text{subject to}: \sum_{i=1}^{d} \mathrm{kl}(\theta_i, \lambda_i) \sum_{M\in\mathcal{M}} M_i x_M \geq 1, \;\; \forall \lambda \in B(\theta).
\end{aligned}
$$

- The first problem dependent tight LB
- Interpretation: each arm M must be sampled at least $x_M^\star \log(T)$ times.
- Proof idea: adaptive control of Markov chains with unknown transition probabilities (Graves & Lai, 1997)

Regret Lower Bound

Theorem

For any uniformly good algorithm π, $\liminf_{T\to\infty} \frac{R^\pi(T)}{\log(T)} \geq c(\theta)$, *with*

$$\begin{aligned} c(\theta) &= \inf_{x \in \mathbb{R}_+^{|\mathcal{M}|}} \sum_{M \in \mathcal{M}} \Delta_M x_M \\ &\text{subject to}: \sum_{i=1}^{d} \mathrm{kl}(\theta_i, \lambda_i) \sum_{M \in \mathcal{M}} M_i x_M \geq 1, \quad \forall \lambda \in B(\theta). \end{aligned}$$

- The first problem dependent tight LB
- Interpretation: each arm M must be sampled at least $x_M^\star \log(T)$ times.
- Proof idea: adaptive control of Markov chains with unknown transition probabilities (Graves & Lai, 1997)

Regret Lower Bound

Theorem

For any uniformly good algorithm π, $\liminf_{T\to\infty} \frac{R^\pi(T)}{\log(T)} \ge c(\theta)$, *with*

$$
\begin{aligned}
c(\theta) \;&=\; \inf_{x \in \mathbb{R}_+^{|\mathcal{M}|}} \sum_{M\in\mathcal{M}} \Delta_M x_M \\
&\text{subject to}: \sum_{i=1}^{d} \mathrm{kl}(\theta_i,\lambda_i) \sum_{M\in\mathcal{M}} M_i x_M \ge 1, \;\; \forall \lambda \in B(\theta).
\end{aligned}
$$

- The first problem dependent tight LB
- Interpretation: each arm M must be sampled at least $x_M^\star \log(T)$ times.
- Proof idea: adaptive control of Markov chains with unknown transition probabilities (Graves & Lai, 1997)

Towards An Explicit LB

How does $c(\theta)$ scale with d, m?

Proposition

For most problems $c(\theta) = \Omega(d - m)$.

- Intuitive since $d - m$ basic actions are not sampled when playing $M^\star$.
- Proof idea
  - Construct a covering set $\mathcal{H}$ for suboptimal basic actions
  - Keeping constraints for $M \in \mathcal{H}$

Definition

$\mathcal{H}$ is a covering set for basic actions if it is a (inclusion-wise) maximal subset of $\mathcal{M} \setminus M^\star$ such that for all distinct $M, M' \in \mathcal{H}$, we have

$$(M \setminus M^\star) \cap (M' \setminus M^\star) = \emptyset.$$

Towards An Explicit LB

How does $c(\theta)$ scale with d, m?

Proposition

For most problems $c(\theta) = \Omega(d - m)$.

- Intuitive since $d - m$ basic actions are not sampled when playing $M^\star$.
- Proof idea
  - Construct a covering set $\mathcal{H}$ for suboptimal basic actions
  - Keeping constraints for $M \in \mathcal{H}$

Definition

$\mathcal{H}$ is a covering set for basic actions if it is a (inclusion-wise) maximal subset of $\mathcal{M} \setminus M^\star$ such that for all distinct $M, M' \in \mathcal{H}$, we have

$$(M \setminus M^\star) \cap (M' \setminus M^\star) = \emptyset.$$

Towards An Explicit LB

How does $c(\theta)$ scale with d, m?

Proposition

For most problems $c(\theta) = \Omega(d - m)$.

- Intuitive since $d - m$ basic actions are not sampled when playing $M^\star$.
- Proof idea
  - Construct a covering set $\mathcal{H}$ for suboptimal basic actions
  - Keeping constraints for $M \in \mathcal{H}$

Definition

$\mathcal{H}$ *is a covering set for basic actions if it is a (inclusion-wise) maximal subset of* $\mathcal{M} \setminus M^\star$ *such that for all distinct* $M, M' \in \mathcal{H}$, *we have*

$$(M \setminus M^\star) \cap (M' \setminus M^\star) = \emptyset.$$

Towards An Explicit LB

How does $c(\theta)$ scale with d, m?

Proposition

For most problems $c(\theta) = \Omega(d - m)$.

- Intuitive since $d - m$ basic actions are not sampled when playing $M^\star$.
- Proof idea
  - Construct a covering set $\mathcal{H}$ for suboptimal basic actions
  - Keeping constraints for $M \in \mathcal{H}$

Definition

$\mathcal{H}$ *is a covering set for basic actions if it is a (inclusion-wise) maximal subset of* $\mathcal{M} \setminus M^\star$ *such that for all distinct* $M, M' \in \mathcal{H}$, *we have*

$$(M \setminus M^\star) \cap (M' \setminus M^\star) = \emptyset.$$

Numerical Experiments

Matchings in $\mathcal{K}_{m,m}$

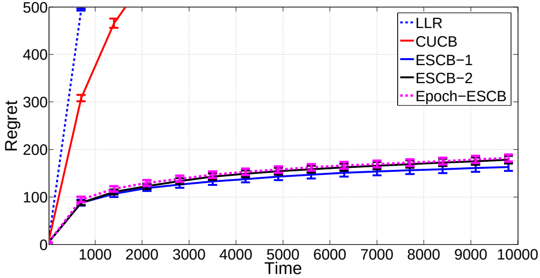
Parameter θ:

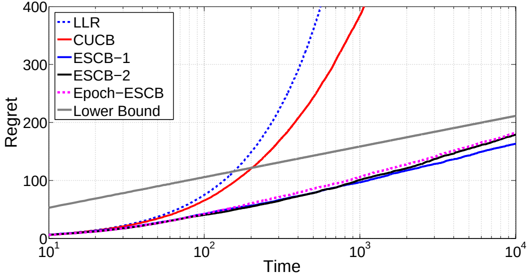
$$\theta_i = \begin{cases} a & i \in M^\star \\ b & \text{otherwise.} \end{cases}$$

$$c(\theta) = \frac{m(m-1)(a-b)}{2\mathrm{kl}(b,a)}$$

matchings $\mathcal{K}_{5,5}$

$a = 0.7,\ b = 0.5$





Outline

Matroid

Combinatorial optimization over a matroid

- Of particular interest in combinatorial optimization
- Power of greedy solution
- Matroid constraints arise in many applications
  - Cardinality constraints, partitioning constraints, coverage constraints

Definition

Given a finite set E and $\mathcal{I} \subset 2^E$, the pair $(E, \mathcal{I})$ is called a matroid if:

(i) *If $X \in \mathcal{I}$ and $Y \subseteq X$, then $Y \in \mathcal{I}$ (closed under subset).*

(ii) *If $X, Y \in \mathcal{I}$ with $|X| > |Y|$, then there is some element $\ell \in X \setminus Y$ such that $Y \cup \{\ell\} \in \mathcal{I}$ (augmentation property).*

Matroid

Combinatorial optimization over a matroid

- Of particular interest in combinatorial optimization
- Power of greedy solution
- Matroid constraints arise in many applications
  - Cardinality constraints, partitioning constraints, coverage constraints

Definition

Given a finite set E and $\mathcal{I} \subset 2^E$, the pair $(E, \mathcal{I})$ is called a matroid if:

(i) *If $X \in \mathcal{I}$ and $Y \subseteq X$, then $Y \in \mathcal{I}$ (closed under subset).*

(ii) *If $X, Y \in \mathcal{I}$ with $|X| > |Y|$, then there is some element $\ell \in X \setminus Y$ such that $Y \cup \{\ell\} \in \mathcal{I}$ (augmentation property).*

Matroid

- E is *ground set*, $\mathcal{I}$ is set of *independent sets*.
- Basis: any inclusion-wise maximal element of $\mathcal{I}$
- Rank: common cardinality of bases

Example: Graphic Matroid (for graph $G = (V, H)$):

$$(H, \mathcal{I}) \quad \text{with} \quad \mathcal{I} = \{F \subseteq H : (V, F) \text{ is a forest}\}.$$

A basis is an spanning forest of the G

Matroid

- E is *ground set*, $\mathcal{I}$ is set of *independent sets*.
- Basis: any inclusion-wise maximal element of $\mathcal{I}$
- Rank: common cardinality of bases

Example: Graphic Matroid (for graph $G = (V, H)$):

$$(H, \mathcal{I}) \quad \text{with} \quad \mathcal{I} = \{F \subseteq H : (V, F) \text{ is a forest}\}.$$

A basis is an spanning forest of the G

Matroid Optimization

- Weighted matroid: is triple $(E, \mathcal{I}, w)$ where w is a positive weight vector (w_ℓ is the weight of $\ell \in E$).
- Maximum-weight basis:

$$\max_{X \in \mathcal{I}} \sum_{\ell \in X} w_\ell$$

- Can be solved *greedily*: At each step of the algorithm, add a new element of E with the largest weight so that the resulting set remains in $\mathcal{I}$.

Matroid Bandits

- Weighted matroid $G = (E, \mathcal{I}, \theta)$
- Set of basic actions $\equiv$ ground set of matroid E
- For each i, $(X_i(n))_{n\geq 1}$ is i.i.d. with Bernoulli of mean θ_i
- Each arm is a basis of G; $\mathcal{M} \equiv$ set of bases of G

Prior work:

- Uniform matroids (Anantharam et al. 1985): Regret LB
- Generic matroids (Kveton et al., 2014): OMM with regret $\mathcal{O}\left(\frac{d}{\Delta_{\min}} \log(T)\right)$

Matroid Bandits

- Weighted matroid $G = (E, \mathcal{I}, \theta)$
- Set of basic actions $\equiv$ ground set of matroid E
- For each i, $(X_i(n))_{n \geq 1}$ is i.i.d. with Bernoulli of mean θ_i
- Each arm is a basis of G; $\mathcal{M} \equiv$ set of bases of G

Prior work:

- Uniform matroids (Anantharam et al. 1985): Regret LB
- Generic matroids (Kveton et al., 2014): OMM with regret $\mathcal{O}\left(\frac{d}{\Delta_{\min}} \log(T)\right)$

Regret LB

Theorem

For all θ and every weighted matroid $G = (E, \mathcal{I}, \theta)$, the regret of uniformly good algorithm π satisfies

$$\liminf_{T\to\infty} \frac{R^\pi(T)}{\log(T)} \geq c(\theta) = \sum_{i \notin M^\star} \frac{\theta_{\sigma(i)} - \theta_i}{\mathrm{kl}(\theta_i, \theta_{\sigma(i)})},$$

where for any i

$$\sigma(i) = \arg \min_{\ell:(M^\star \setminus \ell)\cup\{i\}\in\mathcal{I}} \theta_\ell.$$

- Tight LB, first explicit regret LB for matroid bandits
- Generalizes LB of (Anantharam et al., 1985) to matroids.
- Proof idea
  - Specialization of Graves-Lai result
  - Choosing $d - m$ box constraints in view of σ
  - Lower bounding $\Delta_M, M \in \mathcal{M}$ in terms of σ

Regret LB

Theorem

For all θ and every weighted matroid $G = (E, \mathcal{I}, \theta)$, the regret of uniformly good algorithm π satisfies

$$\liminf_{T\to\infty} \frac{R^\pi(T)}{\log(T)} \geq c(\theta) = \sum_{i \notin M^\star} \frac{\theta_{\sigma(i)} - \theta_i}{\mathrm{kl}(\theta_i, \theta_{\sigma(i)})},$$

where for any i

$$\sigma(i) = \arg \min_{\ell : (M^\star \setminus \ell) \cup \{i\} \in \mathcal{I}} \theta_\ell.$$

- Tight LB, first explicit regret LB for matroid bandits
- Generalizes LB of (Anantharam et al., 1985) to matroids.
- Proof idea
  - Specialization of Graves-Lai result
  - Choosing $d - m$ box constraints in view of σ
  - Lower bounding $\Delta_M, M \in \mathcal{M}$ in terms of σ

Regret LB

Theorem

For all θ and every weighted matroid $G = (E, \mathcal{I}, \theta)$, the regret of uniformly good algorithm π satisfies

$$\liminf_{T\to\infty} \frac{R^\pi(T)}{\log(T)} \geq c(\theta) = \sum_{i \notin M^\star} \frac{\theta_{\sigma(i)} - \theta_i}{\mathrm{kl}(\theta_i, \theta_{\sigma(i)})},$$

where for any i

$$\sigma(i) = \arg \min_{\ell:(M^\star \setminus \ell)\cup\{i\}\in\mathcal{I}} \theta_\ell.$$

- Tight LB, first explicit regret LB for matroid bandits
- Generalizes LB of (Anantharam et al., 1985) to matroids.
- Proof idea
  - Specialization of Graves-Lai result
  - Choosing $d - m$ box constraints in view of σ
  - Lower bounding $\Delta_M, M \in \mathcal{M}$ in terms of σ

KL-OSM Algorithm

KL-OSM (KL-based Optimal Sampling for Matroids)

- Uses KL-UCB index attached to each basic action $i \in E$:

$$\omega_i(n) = \max\Big\{q > \hat{\theta}_i(n) : t_i(n)\mathrm{kl}(\hat{\theta}_i(n), q) \le f(n)\Big\}$$

with $f(n) = \log(n) + 3\log(\log(n))$.

- Relies on $\textsc{Greedy}$

Algorithm 2 KL-OSM

for $n \ge 1$ **do**

Select

$$M(n) \in \arg\max_{M \in \mathcal{M}} \sum_{i \in M} \omega_i(n)$$

using the $\textsc{Greedy}$ algorithm.

Play $M(n)$, observe the rewards, and update $t_i(n)$ and $\hat{\theta}_i(n), \forall i \in M(n)$.

end for

KL-OSM Algorithm

KL-OSM (KL-based Optimal Sampling for Matroids)

- Uses KL-UCB index attached to each basic action $i \in E$:

$$\omega_i(n) = \max\left\{q > \hat{\theta}_i(n) : t_i(n)\mathrm{kl}(\hat{\theta}_i(n), q) \le f(n)\right\}$$

  with $f(n) = \log(n) + 3\log(\log(n))$.

- Relies on Greedy

Algorithm 3 KL-OSM

for $n \ge 1$ **do**

  Select

$$M(n) \in \arg\max_{M \in \mathcal{M}} \sum_{i \in M} \omega_i(n)$$

  using the Greedy algorithm.

  Play $M(n)$, observe the rewards, and update $t_i(n)$ and $\hat{\theta}_i(n), \forall i \in M(n)$.

end for

KL-OSM Regret

Theorem

For any $\varepsilon > 0$, the regret under KL-OSM satisfies

$$R(T) \leq (1+\varepsilon)c(\theta)\log(T) + \mathcal{O}(\log(\log(T)))$$

- KL-OSM is asymptotically optimal:

$$\limsup_{T\to\infty} \frac{R(T)}{\log(T)} \leq c(\theta)$$

- The first optimal algorithm for matroid bandits
- Runs in $\mathcal{O}(d\log(d)T)$ (in the *independence oracle model*)

KL-OSM Regret

Theorem

For any $\varepsilon > 0$, the regret under KL-OSM satisfies

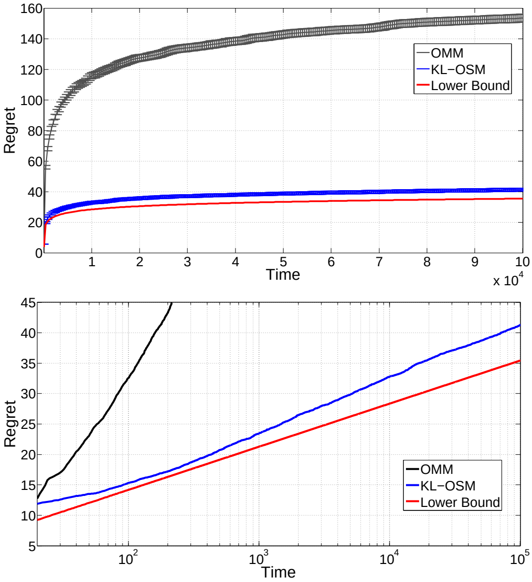
$$R(T) \leq (1+\varepsilon)c(\theta)\log(T) + \mathcal{O}(\log(\log(T)))$$

- KL-OSM is asymptotically optimal:

$$\limsup_{T\to\infty} \frac{R(T)}{\log(T)} \leq c(\theta)$$

- The first optimal algorithm for matroid bandits
- Runs in $\mathcal{O}(d\log(d)T)$ (in the *independence oracle model*)

Numerical Experiments: Spanning Trees



Outline

Adversarial Combinatorial MABs

- Arms have the same cardinality m (but otherwise arbitrary)
- Rewards $X(n) \in [0,1]^d$ are arbitrary (oblivious adversary)
- Bandit feedback: only $M(n)^\top X(n)$ is observed at round n.
- Regret

$$R(T) = \max_{M \in \mathcal{M}} \mathbb{E}\left[\sum_{n=1}^{T} M^\top X(n)\right] - \mathbb{E}\left[\sum_{n=1}^{T} M(n)^\top X(n)\right].$$

$\mathbb{E}[\cdot]$ is w.r.t. random seed of the algorithm.

CombEXP Algorithm

- Inspired by OSMD algorithm (Audibert et al., 2013)

$$\max_{M \in \mathcal{M}} M^\top X = \max_{\alpha \in \mathrm{conv}(\mathcal{M})} \alpha^\top X.$$

- Maintain a distribution $q = \alpha/m$ over basic actions $\{1, \ldots, d\}$.
- q induces a distribution p over arms $\mathcal{M}$.
- Sample M from p, play it, and receive bandit feedback.
- Update q (create $\tilde{q}$) based on feedback.
- Project $\tilde{\alpha} = m\tilde{q}$ onto $\mathrm{conv}(\mathcal{M})$.
- Introduce exploration

CombEXP Algorithm

- Inspired by OSMD algorithm (Audibert et al., 2013)

$$\max_{M \in \mathcal{M}} M^\top X = \max_{\alpha \in \mathrm{conv}(\mathcal{M})} \alpha^\top X.$$

- Maintain a distribution $q = \alpha / m$ over basic actions $\{1, \ldots, d\}$.
- q induces a distribution p over arms $\mathcal{M}$.
- Sample M from p, play it, and receive bandit feedback.
- Update q (create $\tilde{q}$) based on feedback.
- Project $\tilde{\alpha} = m\tilde{q}$ onto $\mathrm{conv}(\mathcal{M})$.
- Introduce exploration

CombEXP Algorithm

- Inspired by OSMD algorithm (Audibert et al., 2013)

$$\max_{M \in \mathcal{M}} M^\top X = \max_{\alpha \in \mathrm{conv}(\mathcal{M})} \alpha^\top X.$$

- Maintain a distribution $q = \alpha/m$ over basic actions $\{1, \dots, d\}$.
- q induces a distribution p over arms $\mathcal{M}$.
- Sample M from p, play it, and receive bandit feedback.
- Update q (create $\tilde{q}$) based on feedback.
- Project $\tilde{\alpha} = m\tilde{q}$ onto $\mathrm{conv}(\mathcal{M})$.
- Introduce exploration

CombEXP Algorithm

- Inspired by OSMD algorithm (Audibert et al., 2013)

$$\max_{M \in \mathcal{M}} M^\top X = \max_{\alpha \in \mathrm{conv}(\mathcal{M})} \alpha^\top X.$$

- Maintain a distribution $q = \alpha/m$ over basic actions $\{1, \ldots, d\}$.
- q induces a distribution p over arms $\mathcal{M}$.
- Sample M from p, play it, and receive bandit feedback.
- Update q (create $\tilde{q}$) based on feedback.
- Project $\tilde{\alpha} = m\tilde{q}$ onto $\mathrm{conv}(\mathcal{M})$.
- Introduce exploration

CombEXP Algorithm

- Inspired by OSMD algorithm (Audibert et al., 2013)

$$\max_{M \in \mathcal{M}} M^\top X = \max_{\alpha \in \mathrm{conv}(\mathcal{M})} \alpha^\top X.$$

- Maintain a distribution $q = \alpha/m$ over basic actions $\{1, \ldots, d\}$.
- q induces a distribution p over arms $\mathcal{M}$.
- Sample M from p, play it, and receive bandit feedback.
- Update q (create $\tilde{q}$) based on feedback.
- Project $\tilde{\alpha} = m\tilde{q}$ onto $\mathrm{conv}(\mathcal{M})$.
- Introduce exploration

CombEXP Algorithm

- Inspired by OSMD algorithm (Audibert et al., 2013)

$$\max_{M \in \mathcal{M}} M^\top X = \max_{\alpha \in \mathrm{conv}(\mathcal{M})} \alpha^\top X.$$

- Maintain a distribution $q = \alpha/m$ over basic actions $\{1, \ldots, d\}$.
- q induces a distribution p over arms $\mathcal{M}$.
- Sample M from p, play it, and receive bandit feedback.
- Update q (create $\tilde{q}$) based on feedback.
- Project $\tilde{\alpha} = m\tilde{q}$ onto $\mathrm{conv}(\mathcal{M})$.
- Introduce exploration

CombEXP Algorithm

- Inspired by OSMD algorithm (Audibert et al., 2013)

$$\max_{M \in \mathcal{M}} M^\top X = \max_{\alpha \in \mathrm{conv}(\mathcal{M})} \alpha^\top X.$$

- Maintain a distribution $q = \alpha/m$ over basic actions $\{1, \ldots, d\}$.
- q induces a distribution p over arms $\mathcal{M}$.
- Sample M from p, play it, and receive bandit feedback.
- Update q (create $\tilde{q}$) based on feedback.
- Project $\tilde{\alpha} = m\tilde{q}$ onto $\mathrm{conv}(\mathcal{M})$.
- Introduce exploration

CombEXP Algorithm

- Inspired by OSMD algorithm (Audibert et al., 2013)

$$\max_{M \in \mathcal{M}} M^\top X = \max_{\alpha \in \mathrm{conv}(\mathcal{M})} \alpha^\top X.$$

- Maintain a distribution $q = \alpha/m$ over basic actions $\{1, \dots, d\}$.
- q induces a distribution p over arms $\mathcal{M}$.
- Sample M from p, play it, and receive bandit feedback.
- Update q (create $\tilde{q}$) based on feedback.
- Project $\tilde{\alpha} = m\tilde{q}$ onto $\mathrm{conv}(\mathcal{M})$.
- Introduce exploration

CombEXP Algorithm

- Inspired by OSMD algorithm (Audibert et al., 2013)

$$\max_{M \in \mathcal{M}} M^\top X = \max_{\alpha \in \mathrm{conv}(\mathcal{M})} \alpha^\top X.$$

- Maintain a distribution $q = \alpha / m$ over basic actions $\{1, \ldots, d\}$.
- q induces a distribution p over arms $\mathcal{M}$.
- Sample M from p, play it, and receive bandit feedback.
- Update q (create $\tilde{q}$) based on feedback.
- Project $\tilde{\alpha} = m\tilde{q}$ onto $\mathrm{conv}(\mathcal{M})$.
- Introduce exploration

CombEXP Algorithm

Algorithm 4 CombEXP

Initialization: Set $q_0 = \mu^0$ (uniform distribution over $[d]$), $\gamma, \eta \propto \frac{1}{\sqrt{T}}$

for $n \geq 1$ **do**

`Mixing:` Let $q'_{n-1} = (1-\gamma)q_{n-1} + \gamma\mu^0$.

`Decomposition:` Select a distribution p_{n-1} over arms $\mathcal{M}$ such that

$$\sum_M p_{n-1}(M)M = mq'_{n-1}.$$

`Sampling:` Select $M(n) \sim p_{n-1}$ and receive reward $Y_n = M(n)^\top X(n)$.

`Estimation:` Let $\Sigma_{n-1} = \mathbb{E}_{M\sim p_{n-1}}\left[MM^\top\right]$. Set $\tilde{X}(n) = Y_n \Sigma_{n-1}^{+} M(n)$.

`Update:` Set $\tilde{q}_n(i) \propto q_{n-1}(i)e^{\eta \tilde{X}_i(n)}$, $\forall i \in [d]$.

`Projection:` Set

$$q_n = \arg\min_{p\in\mathrm{conv}(\mathcal{M})} \mathrm{KL}\left(\frac{1}{m}p, \tilde{q}_n\right).$$

end for

CombEXP: Regret

Theorem

$$R^{\textsc{CombEXP}}(T) \leq 2\sqrt{m^3 T \left(d + \frac{m^{1/2}}{\lambda_{\min}}\right) \log \mu_{\min}^{-1}} + \mathcal{O}(1),$$

where $\lambda_{\min}$ is the smallest nonzero eigenvalue of $\mathbb{E}[MM^\top]$ when M is uniformly distributed and

$$\mu_{\min} = \min_i \frac{1}{|\mathcal{M}|} \sum_{M \in \mathcal{M}} M_i.$$

For most problems $\lambda_{\min} = \Omega(\frac{m}{d})$ and $\mu_{\min}^{-1} = \mathcal{O}(\mathrm{poly}(d/m))$:

$$R(T) \sim \sqrt{m^3 d T \log \frac{d}{m}}.$$

COMBEXP: Regret

Theorem

$$R^{\text{COMBEXP}}(T) \leq 2\sqrt{m^3 T \left(d + \frac{m^{1/2}}{\lambda_{\min}}\right) \log \mu_{\min}^{-1}} + \mathcal{O}(1),$$

where $\lambda_{\min}$ *is the smallest nonzero eigenvalue of* $\mathbb{E}[MM^\top]$ *when* M *is uniformly distributed and*

$$\mu_{\min} = \min_i \frac{1}{|\mathcal{M}|} \sum_{M \in \mathcal{M}} M_i.$$

For most problems $\lambda_{\min} = \Omega(\frac{m}{d})$ and $\mu_{\min}^{-1} = \mathcal{O}(\mathrm{poly}(d/m))$:

$$R(T) \sim \sqrt{m^3 d T \log \frac{d}{m}}.$$

ComBEXP: Regret

Theorem

$$R^{\textsc{ComBEXP}}(T) \le 2\sqrt{m^3 T\left(d + \frac{m^{1/2}}{\lambda_{\min}}\right)\log \mu_{\min}^{-1}} + \mathcal{O}(1),$$

where $\lambda_{\min}$ *is the smallest nonzero eigenvalue of* $\mathbb{E}[MM^\top]$ *when* M *is uniformly distributed and*

$$\mu_{\min} = \min_i \frac{1}{|\mathcal{M}|}\sum_{M\in\mathcal{M}} M_i.$$

For most problems $\lambda_{\min} = \Omega(\frac{m}{d})$ and $\mu_{\min}^{-1} = \mathcal{O}(\mathrm{poly}(d/m))$:

$$R(T) \sim \sqrt{m^3 dT \log\frac{d}{m}}.$$

CombEXP with Approximate Projection

Exact projection with finitely many operations may be impossible
$\Longrightarrow$ CombEXP with approximate projection.

Proposition

Assume that the projection step of CombEXP *is solved up to accuracy*

$$\mathcal{O}\left(\frac{1}{n^2 \log^3(n)}\right), \quad \forall n \geq 1.$$

Then

$$R(T) \leq 2\sqrt{2m^3 T \left(d + \frac{m^{1/2}}{\lambda_{\min}}\right) \log \mu_{\min}^{-1}} + \mathcal{O}(1)$$

- The same regret scaling as for exact projection.
- Proof idea: Strong convexity of KL w.r.t. $\|\cdot\|_1$ + Properties of projection with KL

CombEXP with Approximate Projection

Exact projection with finitely many operations may be impossible
$\Longrightarrow$ CombEXP with approximate projection.

Proposition

Assume that the projection step of CombEXP *is solved up to accuracy*

$$\mathcal{O}\left(\frac{1}{n^2 \log^3(n)}\right), \quad \forall n \geq 1.$$

Then

$$R(T) \leq 2\sqrt{2m^3T\left(d + \frac{m^{1/2}}{\lambda_{\min}}\right)\log \mu_{\min}^{-1}} + \mathcal{O}(1)$$

- The same regret scaling as for exact projection.
- Proof idea: Strong convexity of KL w.r.t. $\|\cdot\|_1$ + Properties of projection with KL

CombEXP with Approximate Projection

Exact projection with finitely many operations may be impossible
$\Longrightarrow$ CombEXP with approximate projection.

Proposition

Assume that the projection step of CombEXP *is solved up to accuracy*

$$\mathcal{O}\left(\frac{1}{n^2 \log^3(n)}\right), \quad \forall n \geq 1.$$

Then

$$R(T) \leq 2\sqrt{2m^3 T \left(d + \frac{m^{1/2}}{\lambda_{\min}}\right) \log \mu_{\min}^{-1}} + \mathcal{O}(1)$$

- The same regret scaling as for exact projection.
- Proof idea: Strong convexity of KL w.r.t. $\|\cdot\|_1$ + Properties of projection with KL

CombEXP: Complexity

Theorem

Let

$$c = \#\ \textit{eq.}\ \mathrm{conv}(\mathcal{M}), \qquad s = \#\ \textit{ineq.}\ \mathrm{conv}(\mathcal{M}).$$

Then, if the projection step of CombEXP *is solved up to accuracy* $\mathcal{O}(n^{-2}\log^{-3}(n))$, $\forall n \geq 1$, CombEXP *after* T *rounds has time complexity*

$$\mathcal{O}(T[\sqrt{s}(c+d)^3\log(T) + d^4]).$$

- Box inequality constraints: $\mathcal{O}(T[c^2\sqrt{s}(c+d)\log(T) + d^4])$.
- Proof idea
  - Constructive proof of Carathéodory Theorem for decomposition
  - Barrier method for projection

$\textsc{CombEXP}$: Complexity

Theorem

Let

$$c = \#\ \textit{eq.}\ \mathrm{conv}(\mathcal{M}), \qquad s = \#\ \textit{ineq.}\ \mathrm{conv}(\mathcal{M}).$$

Then, if the projection step of $\textsc{CombEXP}$ *is solved up to accuracy* $\mathcal{O}(n^{-2}\log^{-3}(n))$, $\forall n \geq 1$, $\textsc{CombEXP}$ *after* T *rounds has time complexity*

$$\mathcal{O}(T[\sqrt{s}(c+d)^3\log(T) + d^4]).$$

- Box inequality constraints: $\mathcal{O}(T[c^2\sqrt{s}(c+d)\log(T) + d^4])$.
- Proof idea
  - Constructive proof of Carathéodory Theorem for decomposition
  - Barrier method for projection

CombEXP: Complexity

Theorem

Let

$$c = \#\ \textit{eq.}\ \mathrm{conv}(\mathcal{M}), \qquad s = \#\ \textit{ineq.}\ \mathrm{conv}(\mathcal{M}).$$

Then, if the projection step of CombEXP *is solved up to accuracy* $\mathcal{O}(n^{-2}\log^{-3}(n))$, $\forall n \geq 1$, CombEXP *after* T *rounds has time complexity*

$$\mathcal{O}(T[\sqrt{s}(c+d)^3\log(T) + d^4]).$$

- Box inequality constraints: $\mathcal{O}(T[c^2\sqrt{s}(c+d)\log(T) + d^4])$.
- Proof idea
  - Constructive proof of Carathéodory Theorem for decomposition
  - Barrier method for projection

Prior Work

Algorithm	**Regret (Symmetric Problems)**
Lower Bound (Audibert et al., 2013)	$\Omega\left(m\sqrt{dT}\right)$, if $d \geq 2m$
ComBand (Cesa-Bianchi & Lugosi, 2012)	$\mathcal{O}\left(\sqrt{m^3 dT \log \frac{d}{m}}\right)$
CombEXP	$\mathcal{O}\left(\sqrt{m^3 dT \log \frac{d}{m}}\right)$

- Both ComBand and CombEXP are off the LB by a factor $\sqrt{m \log(d/m)}$.
- ComBand relies on (approximate) sampling from $\mathcal{M}$ whereas CombEXP does convex optimization over $\mathrm{conv}(\mathcal{M})$.

Prior Work

Algorithm	**Regret (Symmetric Problems)**
Lower Bound (Audibert et al., 2013)	$\Omega\left(m\sqrt{dT}\right)$, if $d \geq 2m$
ComBand (Cesa-Bianchi & Lugosi, 2012)	$\mathcal{O}\left(\sqrt{m^3 dT \log\frac{d}{m}}\right)$
CombEXP	$\mathcal{O}\left(\sqrt{m^3 dT \log\frac{d}{m}}\right)$

- Both ComBand and CombEXP are off the LB by a factor $\sqrt{m\log(d/m)}$.
- ComBand relies on (approximate) sampling from $\mathcal{M}$ whereas CombEXP does convex optimization over $\mathrm{conv}(\mathcal{M})$.

Complexity Example: Matchings

Matchings in $\mathcal{K}_{m,m}$:

- $\mathrm{conv}(\mathcal{M})$ is the set of all doubly stochastic $m \times m$ matrices (*Birkhoff polytope*):

$$\mathrm{conv}(\mathcal{M}) = \left\{ Z \in \mathbb{R}_+^{m\times m} : \sum_{k=1}^{m} z_{ik} = 1,\ \forall i,\ \sum_{k=1}^{m} z_{kj} = 1,\ \forall j \right\}.$$

- $c = 2m$ and $s = m^2$ (box constraints).

Complexity of ComEXP: $\mathcal{O}(m^5 T \log(T))$

- Complexity of CombBand: $\mathcal{O}(m^{10} F(T))$ for some super-linear function $F(T)$ (need for approximating a permanent at each round).

Complexity Example: Matchings

Matchings in $\mathcal{K}_{m,m}$:

- $\mathrm{conv}(\mathcal{M})$ is the set of all doubly stochastic $m \times m$ matrices (*Birkhoff polytope*):

$$\mathrm{conv}(\mathcal{M}) = \left\{ Z \in \mathbb{R}_+^{m\times m} : \sum_{k=1}^{m} z_{ik} = 1,\ \forall i,\ \sum_{k=1}^{m} z_{kj} = 1,\ \forall j \right\}.$$

- $c = 2m$ and $s = m^2$ (box constraints).

Complexity of $\textsc{ComEXP}$: $\mathcal{O}(m^5 T \log(T))$

- Complexity of $\textsc{CombBand}$: $\mathcal{O}(m^{10} F(T))$ for some super-linear function $F(T)$ (need for approximating a permanent at each round).

Outline

Conclusion

- Stochastic combinatorial MABs
  - The first regret LB
  - ESCB: best performance in terms of regret
- Stochastic matroid bandits
  - The first explicit regret LB
  - KL-OSM: the first optimal algorithm
- Adversarial combinatorial MABs
  - CombEXP: the same regret as state-of-the-art but with lower computational complexity
- More in the thesis!

Conclusion

- Stochastic combinatorial MABs
  - The first regret LB
  - ESCB: best performance in terms of regret
- Stochastic matroid bandits
  - The first explicit regret LB
  - KL-OSM: the first optimal algorithm
- Adversarial combinatorial MABs
  - CombEXP: the same regret as state-of-the-art but with lower computational complexity
- More in the thesis!

Conclusion

- Stochastic combinatorial MABs
  - The first regret LB
  - ESCB: best performance in terms of regret
- Stochastic matroid bandits
  - The first explicit regret LB
  - KL-OSM: the first optimal algorithm
- Adversarial combinatorial MABs
  - CombEXP: the same regret as state-of-the-art but with lower computational complexity
- More in the thesis!

Future Directions: Stochastic

- Improvement to the proposed algorithms
  - Tighter regret analysis of ESCB-1 (order-optimality conjecture)
  - Can we amortize index computation?
- Analysis of THOMPSON SAMPLING for stochastic combinatorial MABs
- Stochastic combinatorial MABs under bandit feedback
- Projection-free optimal algorithm for bandit and semi-bandit feedbacks

Future Directions: Stochastic

- Improvement to the proposed algorithms
  - Tighter regret analysis of ESCB-1 (order-optimality conjecture)
  - Can we amortize index computation?
- Analysis of THOMPSON SAMPLING for stochastic combinatorial MABs
- Stochastic combinatorial MABs under bandit feedback
- Projection-free optimal algorithm for bandit and semi-bandit feedbacks

Future Directions: Stochastic

- Improvement to the proposed algorithms
  - Tighter regret analysis of ESCB-1 (order-optimality conjecture)
  - Can we amortize index computation?
- Analysis of THOMPSON SAMPLING for stochastic combinatorial MABs
- Stochastic combinatorial MABs under bandit feedback
- Projection-free optimal algorithm for bandit and semi-bandit feedbacks

Publications

- Combinatorial bandits revisited
  with R. Combes, A. Proutiere, and M. Lelarge **(NIPS 2015)**
- An optimal algorithm for stochastic matroid bandit optimization
  with A. Proutiere **(AAMAS 2016)**
- Spectrum bandit optimization
  with M. Lelarge and A. Proutiere **(ITW 2013)**
- Stochastic online shortest path routing: The value of feedback
  with Z. Zou, R. Combes, A. Proutiere, and M. Johansson **(Submitted to IEEE TAC)**

Thanks for your attention!

Publications

- Combinatorial bandits revisited
  with R. Combes, A. Proutiere, and M. Lelarge **(NIPS 2015)**
- An optimal algorithm for stochastic matroid bandit optimization
  with A. Proutiere **(AAMAS 2016)**
- Spectrum bandit optimization
  with M. Lelarge and A. Proutiere **(ITW 2013)**
- Stochastic online shortest path routing: The value of feedback
  with Z. Zou, R. Combes, A. Proutiere, and M. Johansson **(Submitted to IEEE TAC)**

Thanks for your attention!