Offline-to-Online Transformation via Blackwell Approachability

Negin Golrezaei, Sloan School of Management, MIT

Data-Driven Decision Processes Boot Camp, August 2022

The talk is mainly based on the following paper:

"Online Learning via Offline Greedy Algorithms: Applications in Market Design and Optimization"
with Rad Niazadeh, Joshua Wang, Fransisca Susan, and Ashwinkumar Badanidiyuru,

preliminary conference version: **ACM EC 2021**
full journal version: **Management Science (forthcoming)**

Applications in Modern Marketplaces

- Shared economies
  - Ridesharing
  - Food delivery

- Internet advertising

- Cloud platforms

- Shared economies
  - Ridesharing
  - Food delivery
- Internet advertising
- Cloud platforms

real-time interactions with users

Learning & Optimization

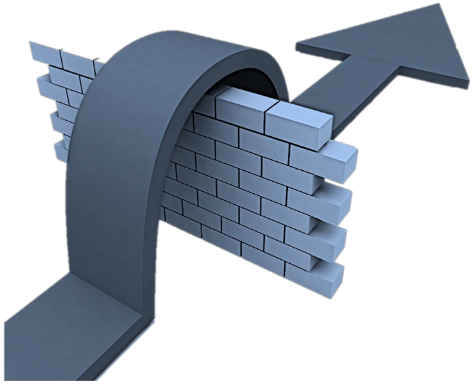
Future uncertainty
(*real-time aspect*)

Time-varying environments
(*dynamic aspect*)

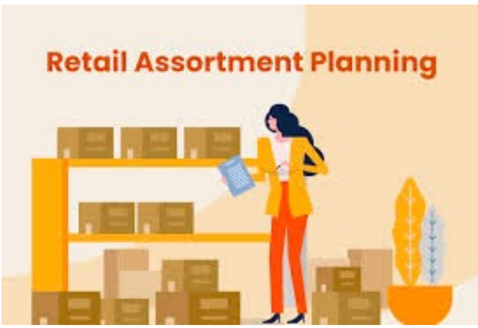
Computational complexity
(*combinatorial aspect*)

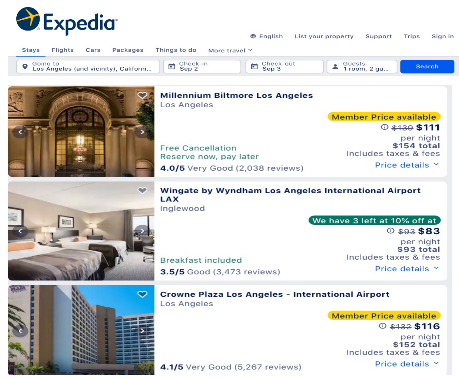
Economical Goals

High-level Goal: decision making by learning and optimization in online marketplaces, despite these challenges



Some Examples (and there are more)





Assortment planning: What items to offer to customers to maximize market share?

Product ranking: How to display products on online platforms?

Reserve price optimization: How to set reserve prices in auctions run to sell ad-views?

Challenges Marketplaces Face

1) **Future uncertainty**: Needs to learn the best course of action
2) **Time-varying environments:** The underlying environments keep changing
3) **Computational complexity:** exponentially many options to try

Assortment planning	Product ranking	Reserve price optimization
Demand is **uncertain** and **time-varying**	Customers' search behaviors are **uncertain** and **time-varying**	Advertisers' values are **uncertain** and **time-varying**
Number of assortments to try is **exponentially** large	Number of rankings to try is **exponentially** large	Number of reserve prices to try is **exponentially** large
NP hard (maximizing sub	**NP hard** (maximizing	**NP hard** [Roughgarden and

Running example in this talk:
Assortment planning to maximize market share

Assortment Planning: Maximizing Market Share

- There are n products
- Our goal is to choose set S with $|S| \leq K$ that maximizes market share
- $\boldsymbol{f(S) = \sum_{i \in S} \textbf{Prob}(\textbf{i is purchased } | S)}$ is the market share (demand) under set S: $f(\cdot)$ is a monotone submodular function
- **Offline problem:** We want to find $S^* = \text{argmax}_{|S| \leq K} f(S)$ **NP hard**
- **Online learning problem:**
  - In every round t∈ $[T]$, there is an unknown demand function $f_t(\cdot)$
  - Choose set S_t
    - Full information: observe $f_t(\cdot)$
    - Bandit: observe $f_t(S_t)$
  - Benchmark= in-hindsight optimal $\boldsymbol{\text{OPT} = \max_{S:|S| \leq K} \sum_t f_t(S)}$

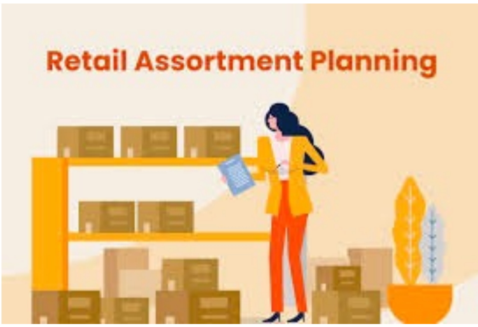
From Offline Optimization to Online Learning

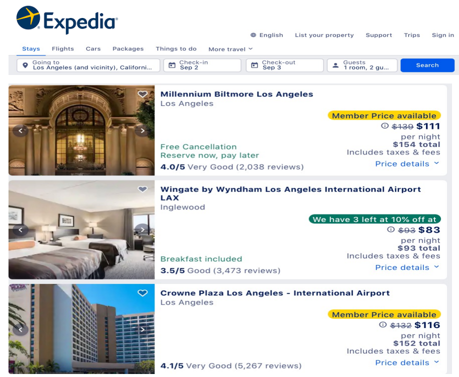
- Occasionally, even the offline optimization problem is NP-hard
  - We have access to approximation algorithms; greedy, LP relaxation & rounding, primal-dual, etc.
- Their performance guarantees only hold in the offline regime
- What about the online regime, with repeated interactions?

Agenda: developing generic tools to transform a large class of approximation algorithms used in marketplaces to their online variants, with almost no performance loss (over time)

We "mimic" the structure of offline approximation algorithm using Blackwell approachability

This Work: Iterative Greedy Algorithms





- **Assortment planning**
  Greedy ✓
- **Product ranking**
  Greedy ✓
- **Reserve price optimization**
  Greedy ✓

- A large class of problems in market design and revenue management admit ***greedy-type algorithms*** for their offline problem (with theoretical guarantees).

Greedy Algorithm for Assortment Planning Problem

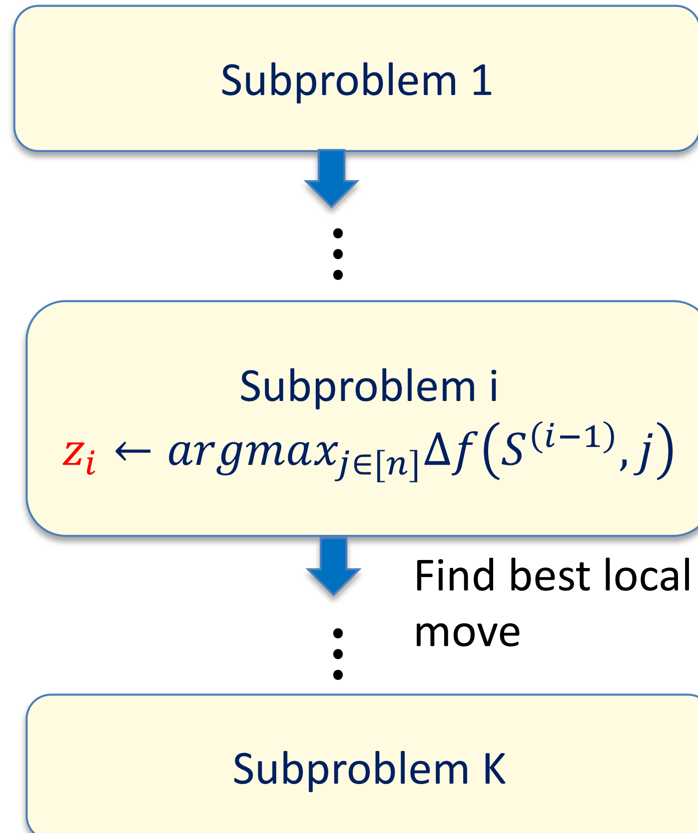
Greedy is (1-1/e)-approximation for maximizing monotone submodular functions subject to cardinality **[Nemhauser et al., 1987]**

Initialize $S^{(0)} = \{\}$
For subproblem $i \ = \ 1 \ to \ K$:

Greedily pick $z_i \in [n]$ such that
$$z_i \leftarrow argmax_{j\in[n]}\{f\left(S^{(i-1)} \cup \{j\}\right) - f\left(S^{(i-1)}\right)\}$$

Set $S^{(i)} \leftarrow S^{(i-1)} \cup \{z_i\}$
End
Return $S^{(K)}$

Choose a product with the maximum marginal market share $\Delta f(S^{(i-1)}, j)$



Research Question: Can we transform ***offline*** iterative greedy algorithms in a computationally efficient fashion to ***online*** algorithms with sublinear ***approximate regret***?

Approximate regret = *regret bound with respect to* γ *times the best in-hindsight benchmark, where* $\gamma \in [0,1]$ *is the approximation factor of greedy*

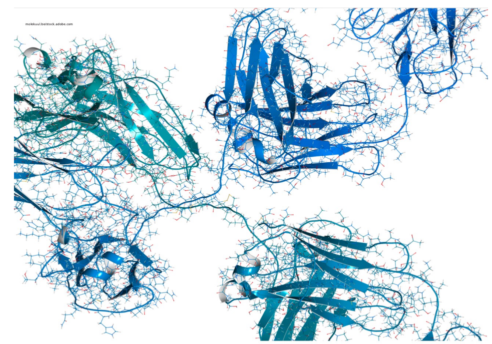
$$\gamma\text{-Regret} = \gamma \max_{S:|S|\leq K} \sum_t f_t(S) - \sum_{t\in[T]} f_t(S_t)$$

Power of Framework

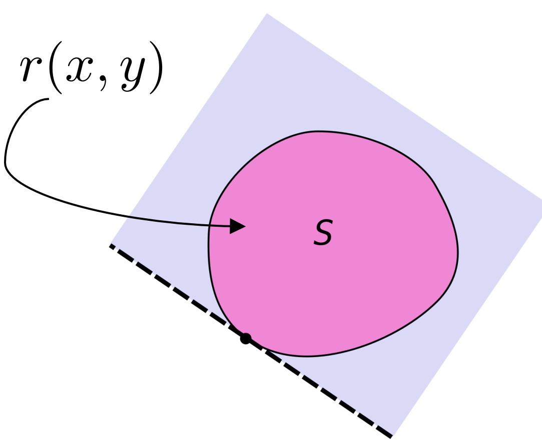
- Identifying simple generic conditions under which such a transformation exists
- Showing a large class of problems admit iterative greedy algorithms satisfying these conditions

Two Natural/General Conditions

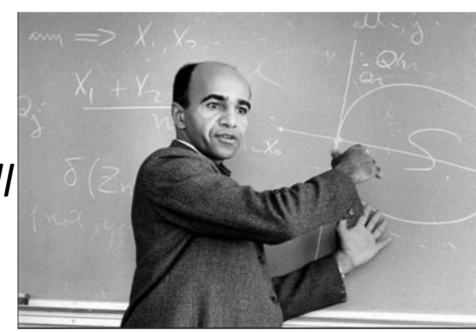
I. **Robustness of greedy to local errors:**
   - Small mistakes at each subproblem of greedy only harm the objective as much as the mistake; no error amplification



II. **Blackwell Locality:**
   - Each subproblem of the greedy algorithm can be cast as a ***Blackwell game***



David Blackwell (1919-2010)



Blackwell Games

Repeated Blackwell game: Repeated two-player (P1 and P2) zero-sum game with vector-valued biaffine payoff

Round t	
P1 plays x_t P2 plays y_t	P1 obtains $\mathbf{payoff}(\mathbf{x}_t, \mathbf{y}_t)$ P2 obtains $-\mathbf{payoff}(\mathbf{x}_t, \mathbf{y}_t)$

…

Round T	
P1 plays x_T P2 plays y_T	P1 obtains $\mathbf{payoff}(\mathbf{x}_T, \mathbf{y}_T)$ P2 obtains $-\mathbf{payoff}(\mathbf{x}_T, \mathbf{y}_T)$

P1 wants to approach a **convex *S*** and P2 does not want this to happen

$$d_\infty\left(\frac{1}{T}\sum_{t=1}^{T}\mathrm{payoff}(\mathrm{x}_t, \mathrm{y}_t)\,, S\right) = o(1)$$

If set S is “approachable” in a single-shot Blackwell game, in the repeated game, P1 can approach it using *Blackwell algorithm* AlgB

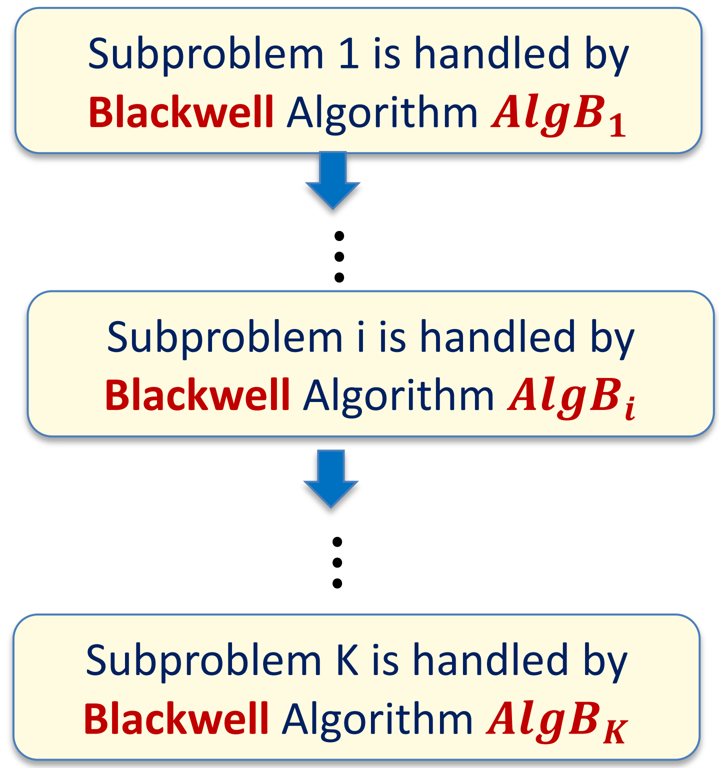
High-level Idea

Blackwell Locality: We cast each subproblem i as a Blackwell game

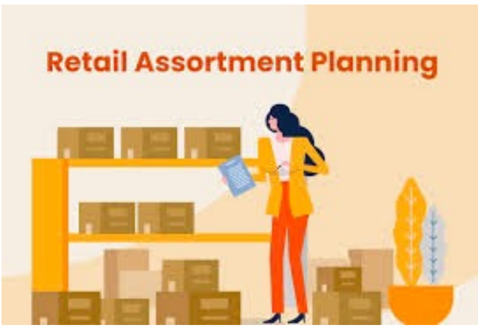
- With **biaffine payoff vector** and **approachable target convex set S**
- With the help of **AlgB**, we can generate a sequence of actions that are *"almost locally best on average over time"*

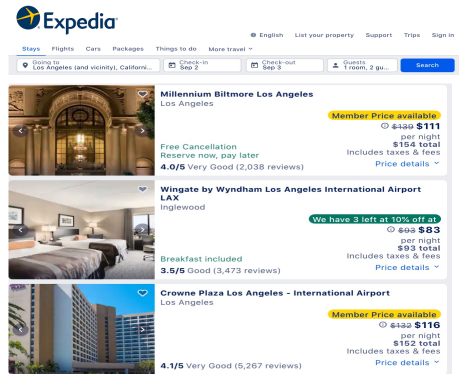
Robustness to local errors: Errors across subproblems don't get amplified

Online problem (round t)



Two Natural/General Conditions





- ◎ **Assortment planning**
  Greedy ✓
  Robustness ✓
  Blackwell locality ✓

- ◎ **Product ranking**
  Greedy ✓
  Robustness ✓
  Blackwell locality ✓

- ◎ **Reserve price optimization**
  Greedy ✓
  Robustness ✓
  Blackwell locality ✓

Contributions and Main Results

- Designing an efficient framework to transform offline iterative greedy algorithms to low-regret adversarial online learning algorithms via **Blackwell approachability**
  - *Vanishing approximate regret*
  - $O(\sqrt{T})$ *for* ***full information*** *and* $O(T^{2/3})$ *for* ***bandit information***
  - *Two conditions*: ***robustness to local errors*** *and* ***Blackwell locality***

- Wide-range of apps in operations & markets

Contributions and Main Results

- Our framework has a wide-range of applications

		Online Full-Information Setting		Online Bandit Setting	
Applications	γ	Our γ-Regret Bound	The Best Prior Bound	Our γ-Regret Bound	The Best Prior Bound
Product Ranking	1/2	$O(n\sqrt{T}\log n)$	-	$O(n^{5/3}T^{2/3}(\log n)^{1/3})$	-
Reserve Price Optimization	1/2	$O(n\sqrt{T}\log T)$	$O(n\sqrt{T}\log T)^{*}$	$O(n^{3/5}T^{4/5}(\log nT)^{1/3})$	-
Non-Monotone Set SM	1/2	$O(n\sqrt{T})$	$O(n\sqrt{T})^{\ddagger}$	$O(nT^{2/3})$	-
Non-Monotone Strong-DR SM	1/2	$O(n\sqrt{T}\log T)$	$\gamma = 1/4$, $O(T^{5/6})^{\S}$	$O(nT^{4/5}(\log T)^{1/3})$	$\gamma = \frac{1}{4}, O(T^{11/12})^{\S}$
Non-Monotone Weak-DR SM	1/2	$O(n\sqrt{T}\log T)$	-	$O(nT^{4/5}(\log T)^{1/3})$	-
Monotone Cont. SM (Strong-DR) in Downward Closed Convex Set	1-1/e	$O(\sqrt{Tn\log(n)})$	$O(\sqrt{T})^{\$}$	$O(nT^{5/6}(\log n)^{1/6})$	$O(nT^{8/9})^{@}$

$\sqrt{T}$ dependency

Discrete: $T^{\frac{2}{3}}$ dependency

*Roughgarden and Wang, 2019; ‡Roughgarden and Wang, 2018; §Thang and Srivastav, 2019; $ Chen et al 2018; @ Zhang et al 2020

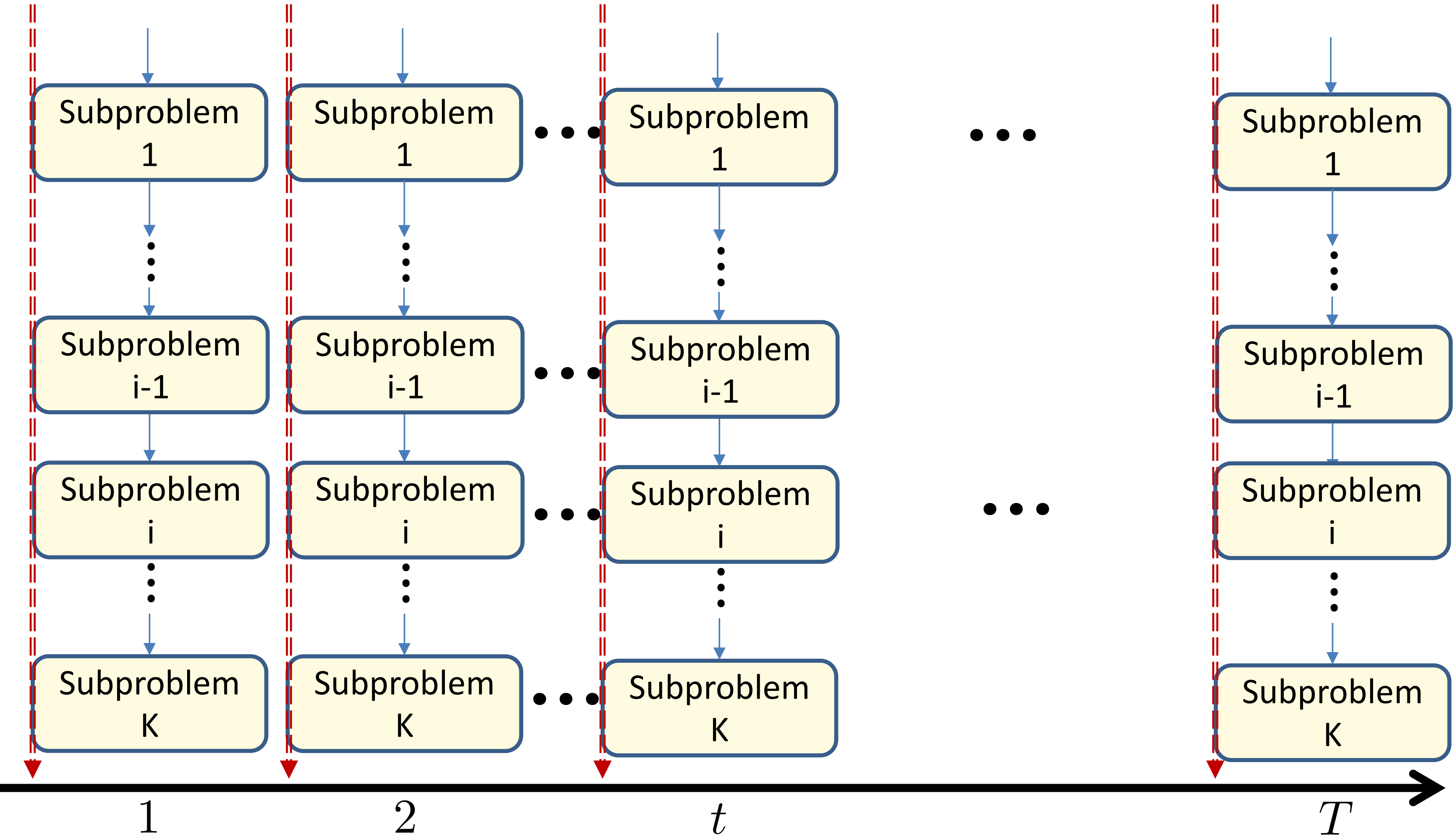
Related Work

Offline-to-online transformation for NP-hard combinatorial problems

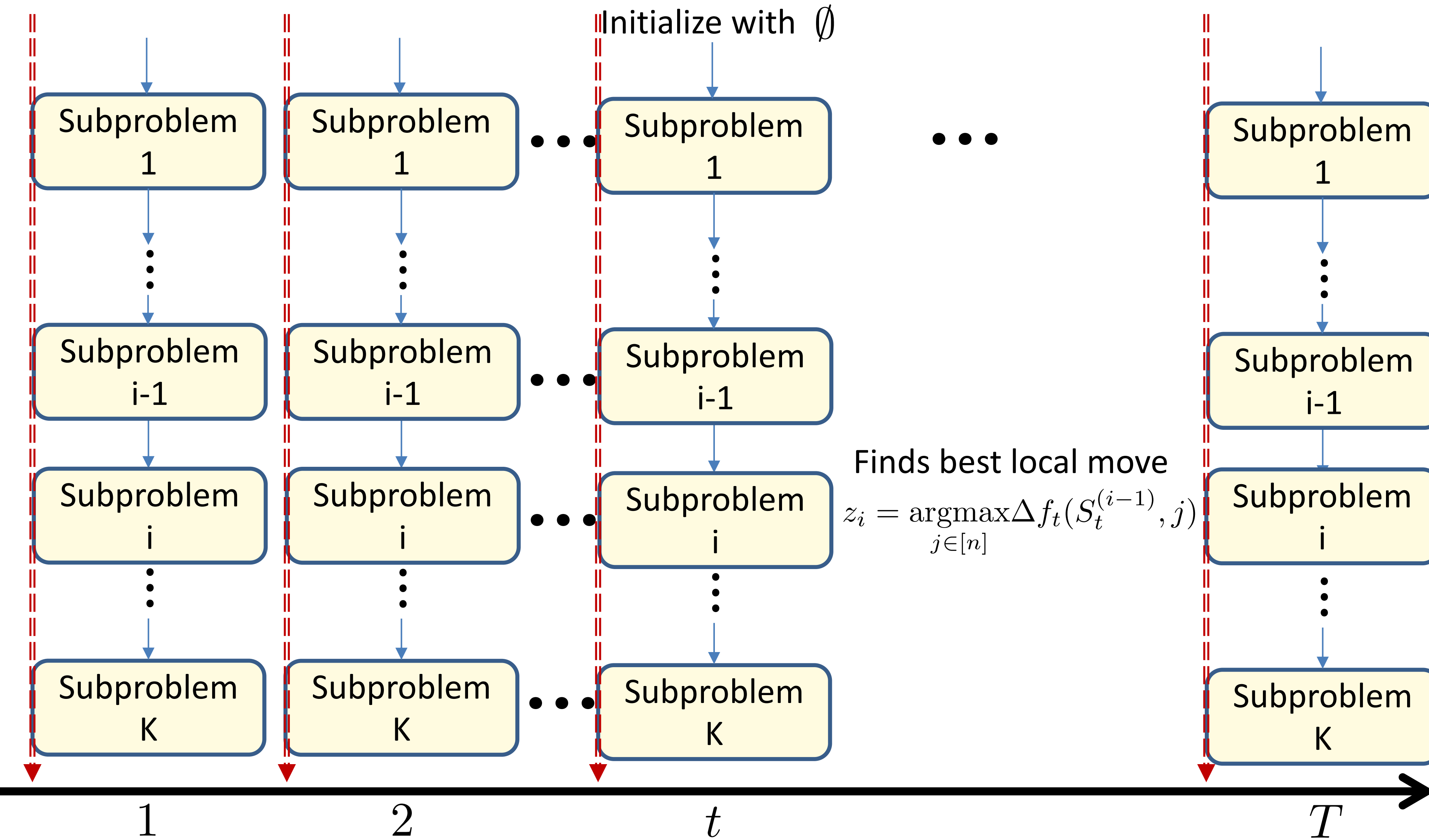
Offline-to-online transformation	• Hazan and Koren, 2016 – **negative results** for general comb. problems • Kalai and Vempala, 2005, Dudik et al., 2017 – learner can solve **offline problem efficiently** • Kakade et al., 2009 – NP-hard problem amenable to approximation, **linear rewards**
Combinatorial learning	• Audibert et al., 2014 – exponentially weighted avg. forecaster for full-info setting, tight regret, **linear rewards** • Bubeck et al., 2012, Hazan and Karnin, 2016 – efficient algorithm for the bandit setting, **linear rewards**
Our contribution	• NP-hard problems with **non-linear rewards** • Both bandit and full-information settings • Transform **offline greedy** algorithms to online

High-level Sketch of our Approach

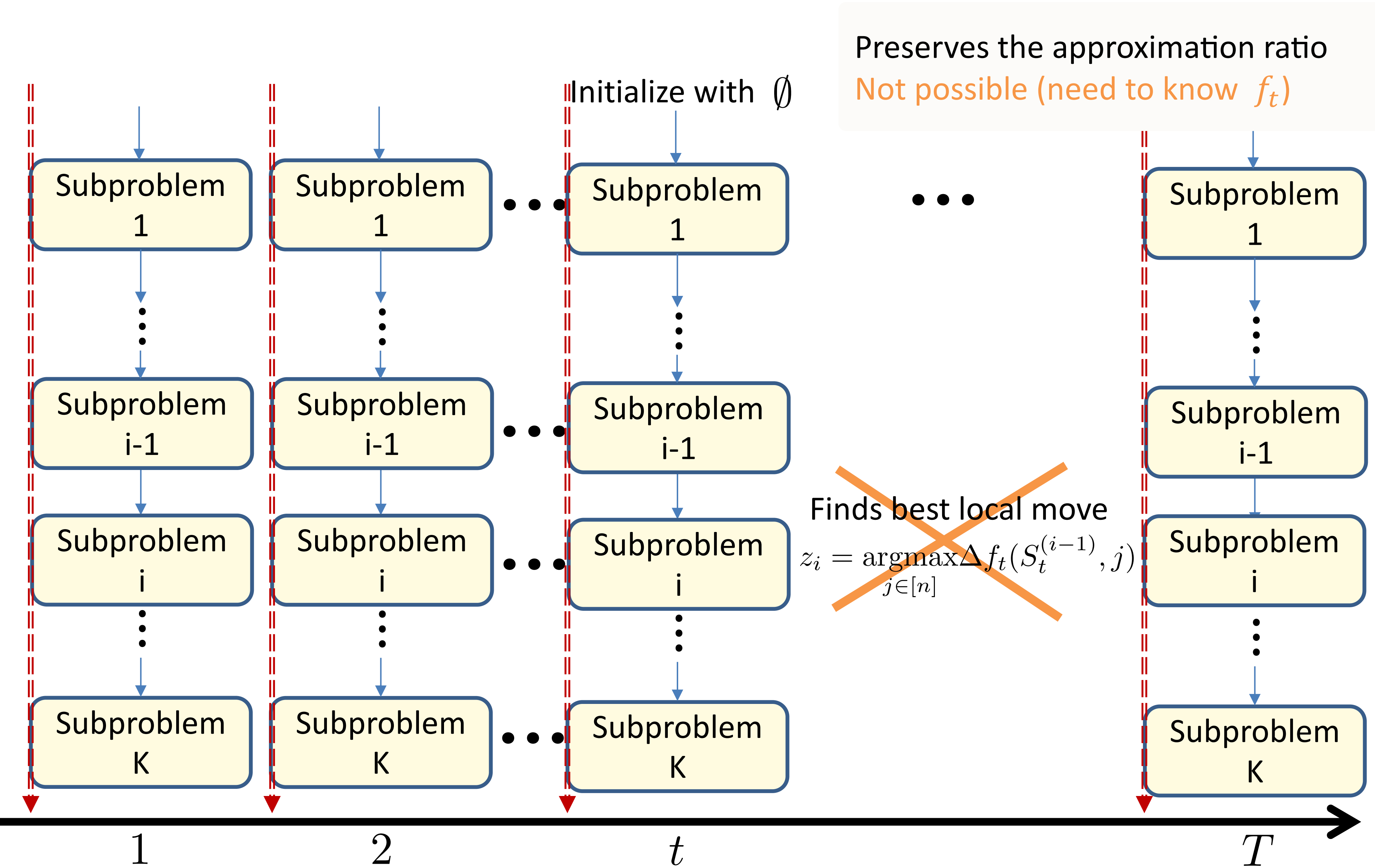
Mimicking the Greedy Algorithm



Mimicking the Greedy Algorithm



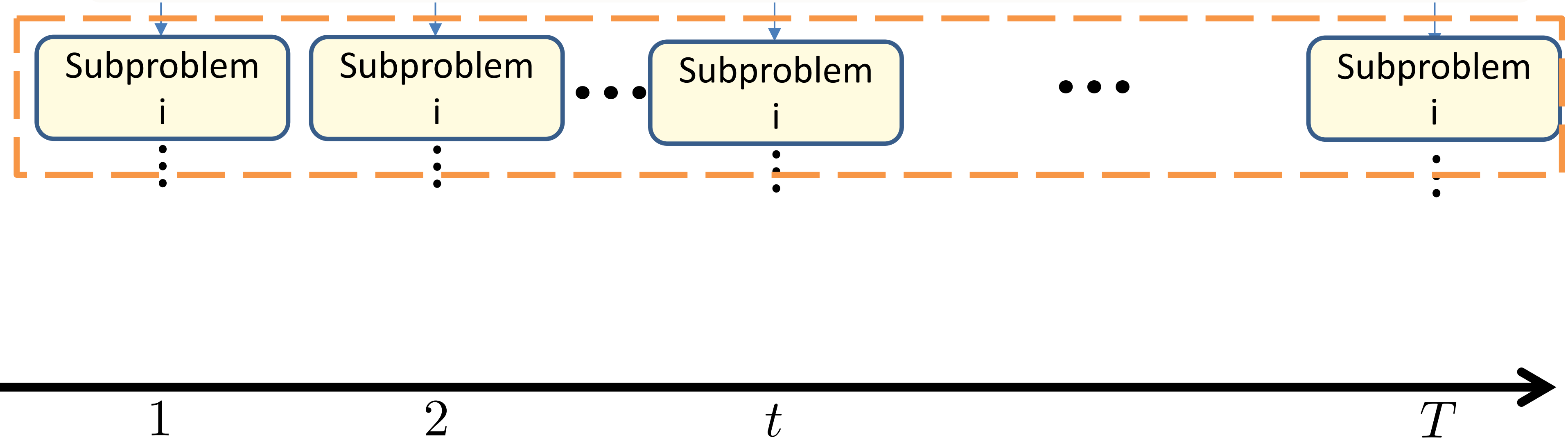
Mimicking the Greedy Algorithm



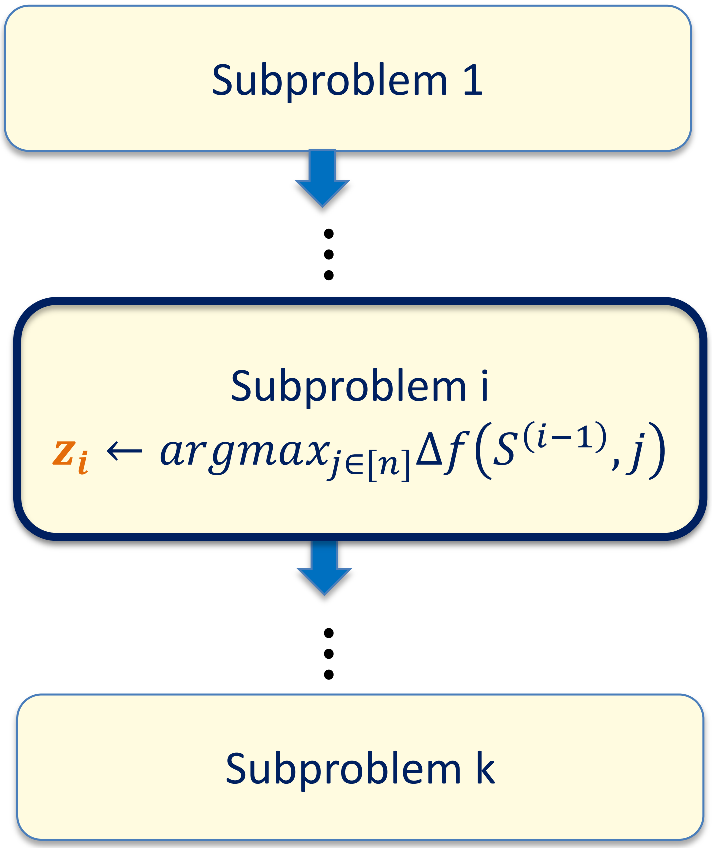
Mimicking the Greedy Algorithm

What is each subproblem i makes a sequence of actions that are (almost) locally best, but on an average sense over time?

This can be done under Blackwell locality condition!



Revisiting the Greedy Algorithm



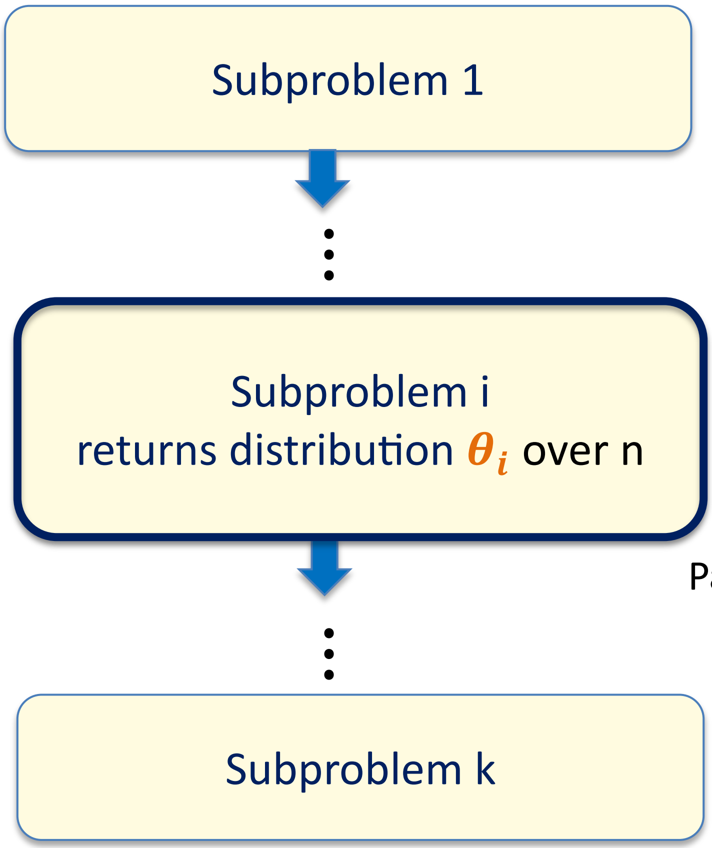
Greedy chooses product z_i that **maximizes** marginal market share $\boldsymbol{\Delta f(S^{(i-1)},\cdot)}$

$$\text{Payoff}(z_i, S^{(i-1)}, \Delta f) = \begin{pmatrix} \Delta f\left(S^{(i-1)}, z_i\right) - \Delta f\left(S^{(i-1)}, 1\right) \\ \Delta f\left(S^{(i-1)}, z_i\right) - \Delta f\left(S^{(i-1)}, 2\right) \\ \vdots \\ \Delta f\left(S^{(i-1)}, z_i\right) - \Delta f\left(S^{(i-1)}, n\right) \end{pmatrix} \geq \mathbf{0}$$

Issue: vector payoff is not linear in the greedy's decisions z_i!

$\Delta f(S, j) = f(S \cup \{j\}) - f(S)$ marginal market share of adding product j to set S

Revisiting the Greedy Algorithm



Greedy chooses distribution $\boldsymbol{\theta_i}$ on products that **maximizes** marginal market share $\boldsymbol{\Delta f(S^{(i-1)},\cdot)}$

$$\text{Payoff}(\theta_i, S^{(i-1)}, \Delta f) = \begin{pmatrix} \sum_{j\in[n]} \theta_{i,j} \Delta f(S^{(i-1)}, j) - \Delta f(S^{(i-1)}, \mathbf{1}) \\ \sum_{j\in[n]} \theta_{i,j} \Delta f(S^{(i-1)}, j) - \Delta f(S^{(i-1)}, \mathbf{2}) \\ \vdots \\ \sum_{j\in[n]} \theta_{i,j} \Delta f(S^{(i-1)}, j) - \Delta f(S^{(i-1)}, \boldsymbol{n}) \end{pmatrix} \geq \mathbf{0}$$

Vector payoff is now LINEAR in the greedy's decisions $\boldsymbol{\theta_i}$!

$\sum_{j\in[n]} \theta_{i,j} \Delta f(S^{(i-1)}, j)$ is the expected value of marginal market share at the greedy solution $\boldsymbol{\theta_i}$

Blackwell Locality:Casting Subproblems as Blackwell Games

- P1 is algorithm that returns $\boldsymbol{\theta_i}$
- P2 is the nature (ADV) that chooses $\boldsymbol{\Delta f\left(S^{(i-1)},.\right)}$
- Per period payoff vector is biaffine

$$\text{Payoff}(\boldsymbol{\theta_i}, S^{(i-1)}, \boldsymbol{\Delta f}) = \begin{pmatrix} \sum_{j\in[n]} \boldsymbol{\theta_{i,j}} \boldsymbol{\Delta f}\left(S^{(i-1)}, j\right) - \boldsymbol{\Delta f}\left(S^{(i-1)}, 1\right) \\ \sum_{j\in[n]} \boldsymbol{\theta_{i,j}} \boldsymbol{\Delta f}\left(S^{(i-1)}, j\right) - \boldsymbol{\Delta f}\left(S^{(i-1)}, 2\right) \\ \vdots \\ \sum_{j\in[n]} \boldsymbol{\theta_{i,j}} \boldsymbol{\Delta f}\left(S^{(i-1)}, j\right) - \boldsymbol{\Delta f}\left(S^{(i-1)}, n\right) \end{pmatrix} \geq \mathbf{0}$$

- Target set S is the positive orthant Payoff $\left(\boldsymbol{\theta_i}, S^{(i-1)}, \boldsymbol{\Delta f}\right) \geq \mathbf{0}$ and is approachable

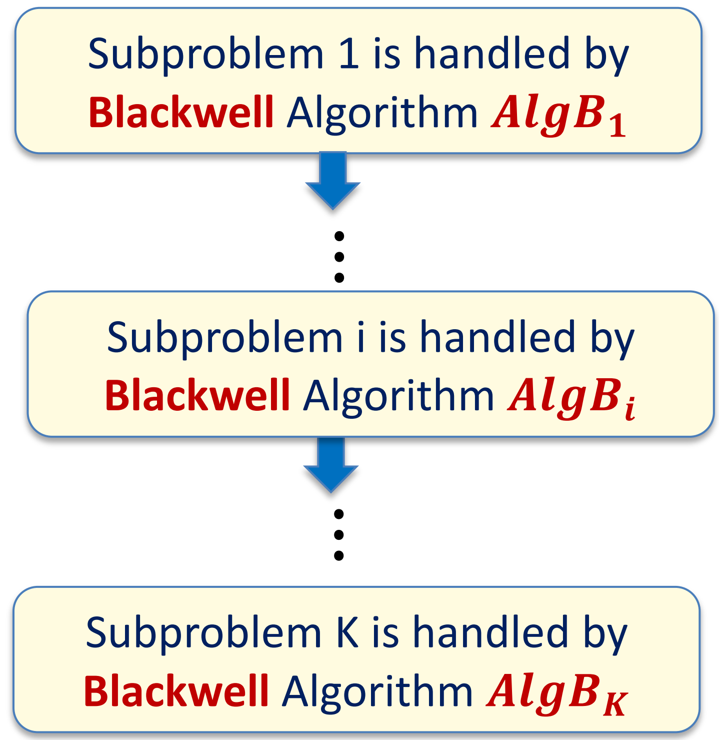
High-level Idea

Blackwell Locality: We cast each subproblem i as a Blackwell game

- With **biaffine payoff vector** and **approachable target convex set S**
- With the help of **AlgB**, we can generate a sequence of actions that are *"almost locally best on average over time"*

Robustness to local errors: Errors across subproblems don't get amplified

Online problem (round t)



Full Information

Theorem 1 (Full-information offline-to-online transformation) Suppose that an offline algorithm is

- **robust to local errors**, and
- **Blackwell local.**

Then, in the full information setting, there exists an online algorithm that runs in polynomial time and satisfies:

$$\gamma - \text{regret} \leq O\left(\boldsymbol{K}\boldsymbol{T}^{\mathbf{1}/\mathbf{2}}\right)$$

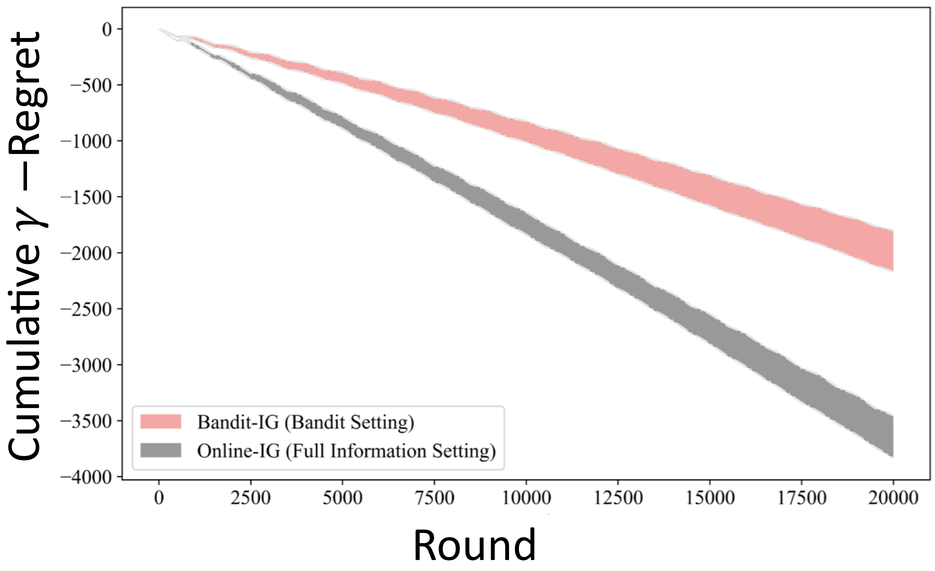
where K is the number of subproblems.

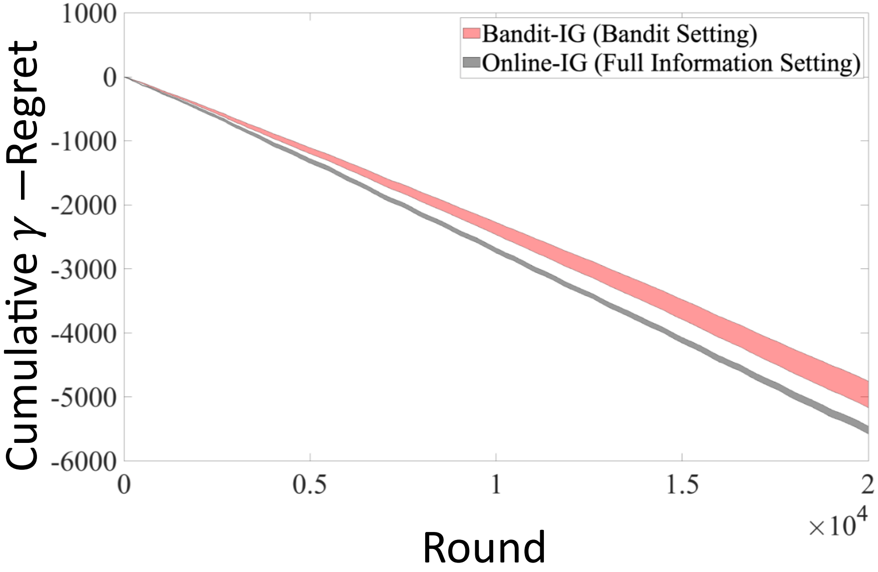
Blackwell local:

1) Defining **bi-affine** vector payoff for each subproblem
2) Defining an **approachable target set** for each subproblem

Applications

Product Ranking and Reserve Price Optimization





Product ranking

Reserve price optimization

Negative Cumulative γ −Regret (Doing much better than our theoretical results)

Takeaway

- Transform offline greedy algorithms to online ones using **Blackwell approachability**
  - Need the greedy algorithm to be robust to local errors and Blackwell local

- For full information setting, our algorithm has $\boldsymbol{O}\left(\sqrt{\boldsymbol{T}}\right)$ γ −regret
- For Bandit setting, our algorithm has $\boldsymbol{O}\left(\boldsymbol{T}^{2/3}\right)$ γ −regret

- Our framework is flexible and can be applied to many applications
  - Product ranking optimization in online platforms
  - Reserve price optimization in auctions
  - Submodular maximization

Link to the paper: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3613756

Email: golrezae@mit.edu