Efficient and Effective Algorithms for A Family of Influence Maximization Problems with A Matroid Constraint

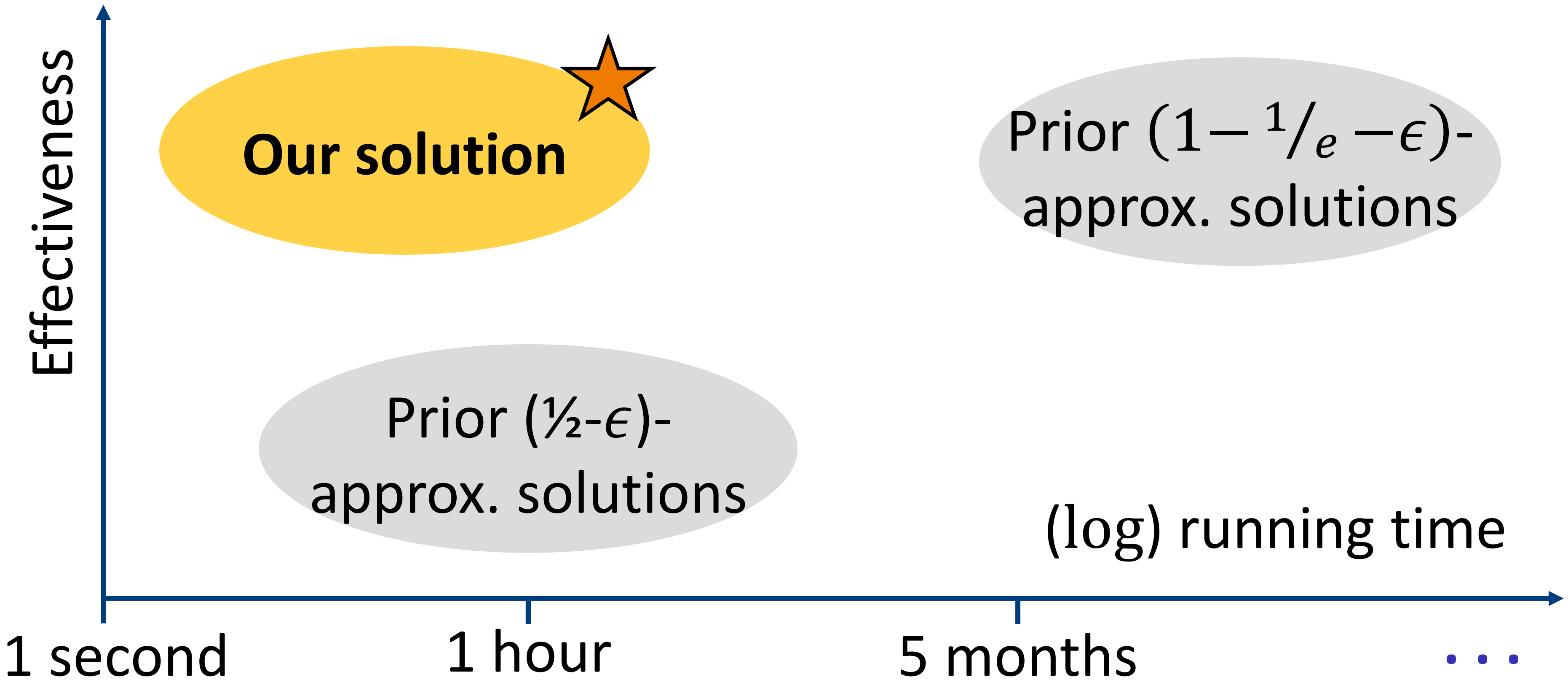
[Research track – Social Networks]

Yiqian Huang, Shiqi Zhang, Laks V.S. Lakshmanan,
Wenqing Lin, Xiaokui Xiao, Bo Tang

NUS National University of Singapore
PyroWis
UBC THE UNIVERSITY OF BRITISH COLUMBIA
Tencent
南方科技大学 SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

TL; DR

- We study **influence maximization** problems with a **matroid** constraint
- We give a **fast** solution with a $(\mathbf{1 - 1/e - \epsilon})$**-approximation**



Roadmap

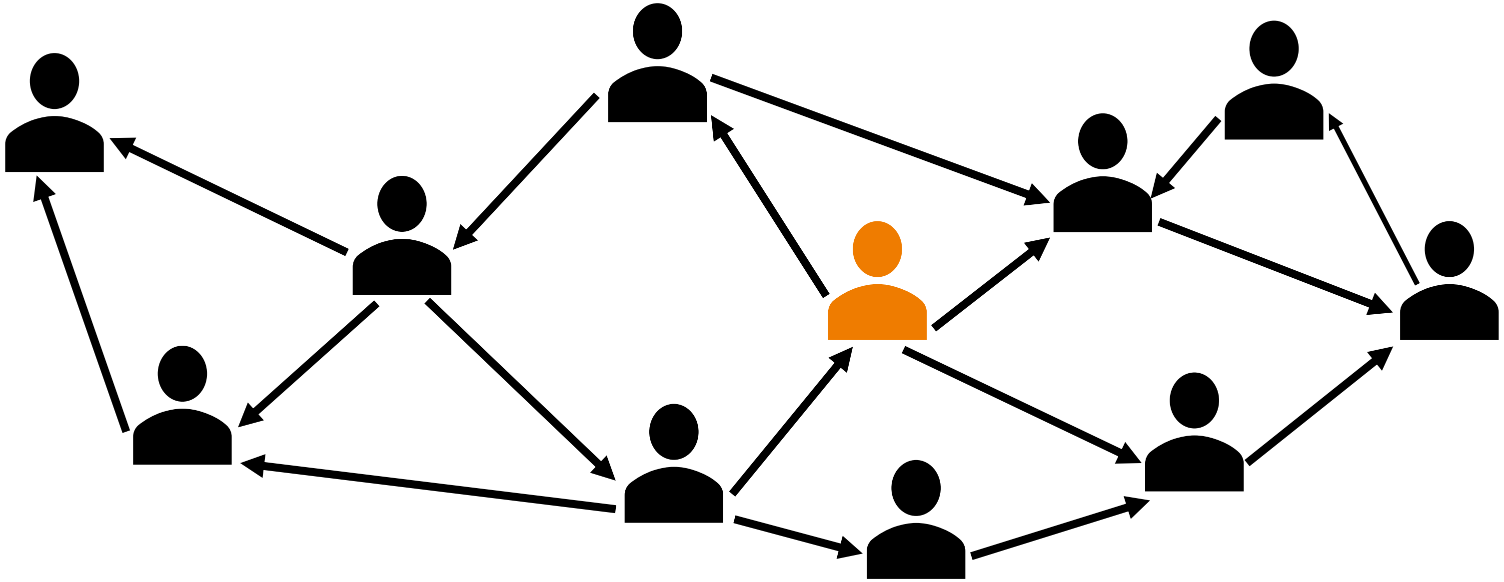
- Influence Maximization (IM)
- Why matroid constraints
- Our idea & guarantees
- Experiments

Roadmap

- **Influence Maximization (IM)**
- Why matroid constraints
- Our idea & guarantees
- Experiments

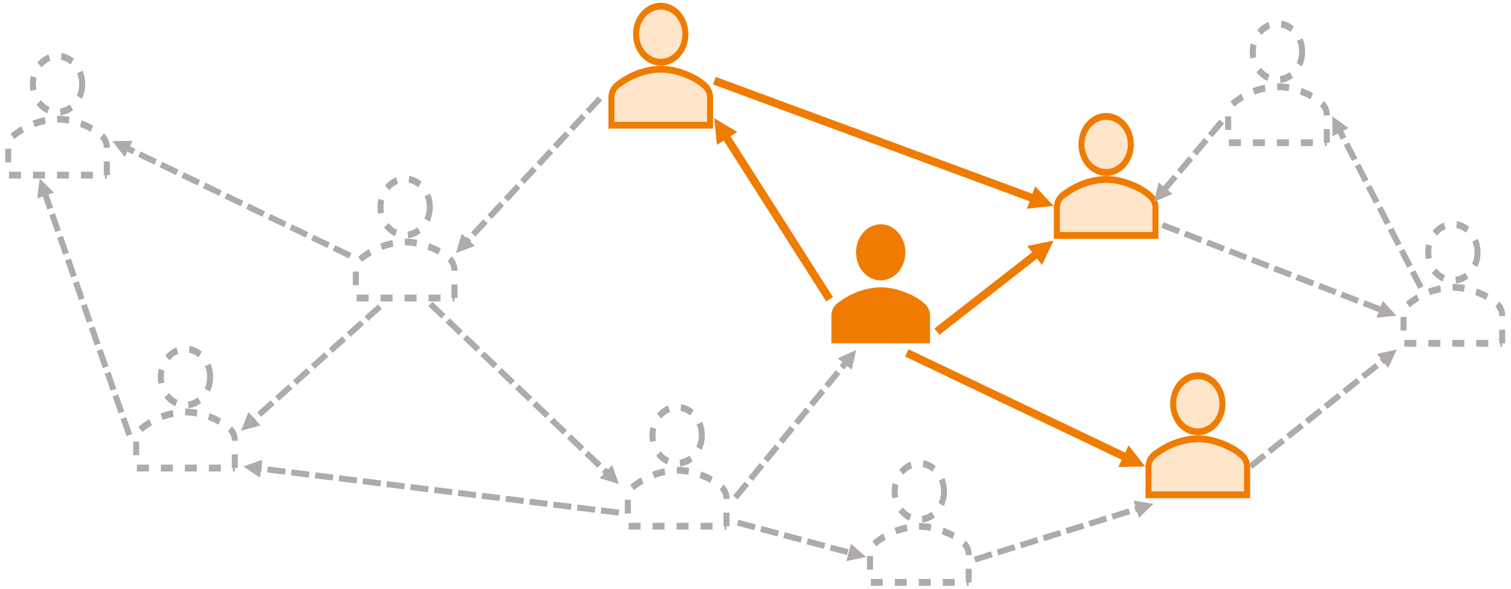
Influence Maximization (IM)

- In a social network, each user (say ) has *influence*



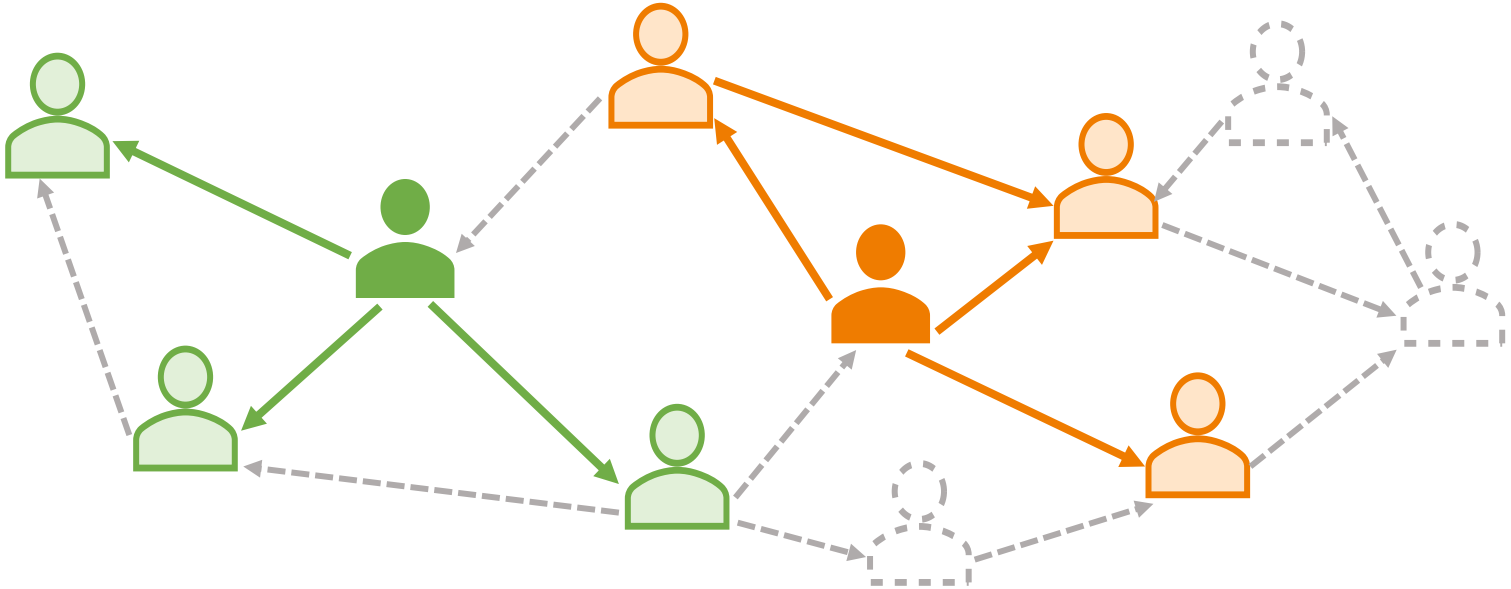
Influence Maximization (IM)

- In a social network, each user (say 👤) has *influence*
- Think of the influence as a (random) set of users they can reach (i.e., 👤)



Influence Maximization (IM)

- In a social network, each user (say 👤) has *influence*
- Think of the influence as a (random) set of users they can reach (i.e., 👤)
- IM aims to select k users to **maximize their joint influence**



Example ($k = 2$; result of selecting 👤 👤)

Influence Maximization (IM)

- In a social network, each user (say) has *influence*
- Think of the influence as a (random) set of users they can reach (i.e.,)
- IM aims to select k users to **maximize their joint influence**
- Applied in *viral marketing*, *recommendation*, *power grids management*, etc.

Influence Maximization (IM)

- In a social network, each user (say) has *influence*
- Think of the influence as a (random) set of users they can reach (i.e.,)
- IM aims to select k users to **maximize their joint influence**
- Applied in *viral marketing*, *recommendation*, *power grids management*, etc.
- **Example (viral marketing with ad promotion):**

= TikTok influencers who we want to endorse an ad

= Others who will watch the ad **(the more, the better for us!)**

k = our budget for launching the campaign

Roadmap

- Influence Maximization (IM)
- **Why matroid constraints**
- Our idea & guarantees
- Experiments

Why Matroid Constraints

- In (classical) IM, we select *a single set of* k *users*

Why Matroid Constraints

- In (classical) IM, we select *a single set of k users*
- Real-world scenarios often need **multiple** such sets

Why Matroid Constraints

- In (classical) IM, we select *a single set of* k *users*
- Real-world scenarios often need **multiple** such sets
- A **matroid** $(U, \mathbb{I})$ can model constraints across them

Why Matroid Constraints

- **Example 1 (Classical IM):**

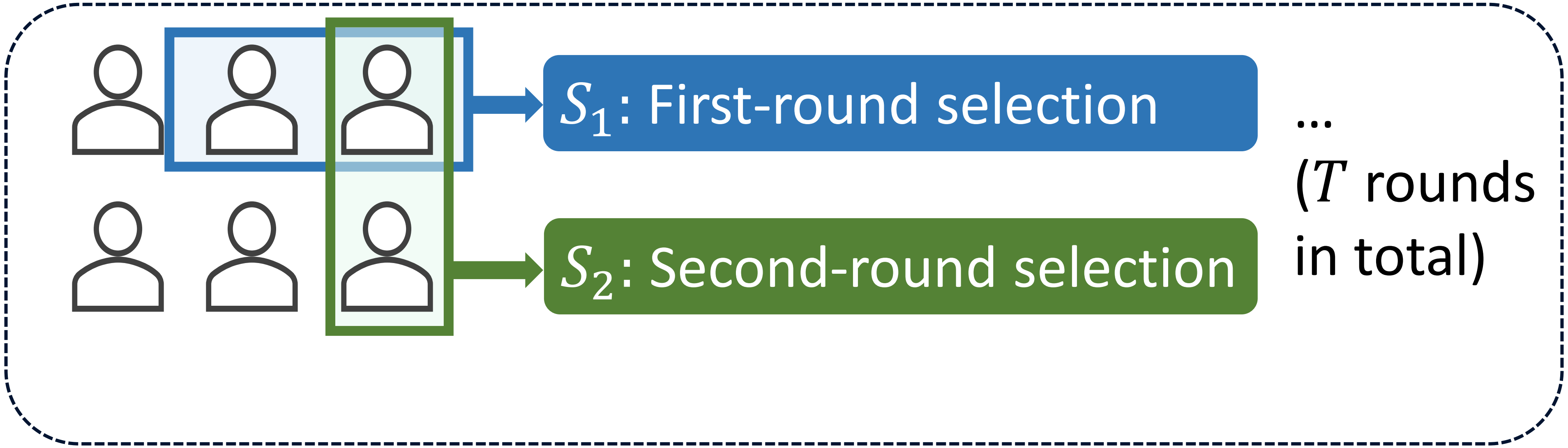
A *uniform matroid* $(U, \mathbb{I})$ specifies:

U: **all users**

$\mathbb{I}$: $|\boldsymbol{S}| \leq \boldsymbol{k}$ (at most k users)

Why Matroid Constraints

- **Example 2 (Multi-round):**

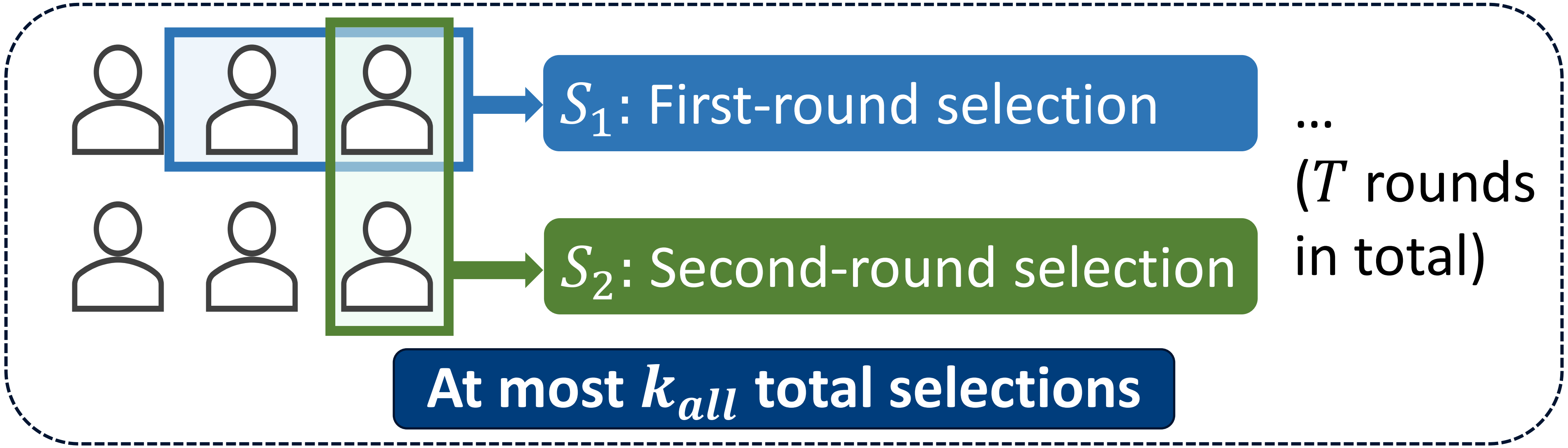


A *partition matroid* $(U, \mathbb{I})$ specifies:

U: **all users × [T rounds]** $\mathbb{I}$: $\forall \boldsymbol{i}, |\boldsymbol{S_i}| \leq \boldsymbol{k_i}$ (at most k_i users in round i)

Why Matroid Constraints

- **Example 3 (Multi-round + an overall budget):**

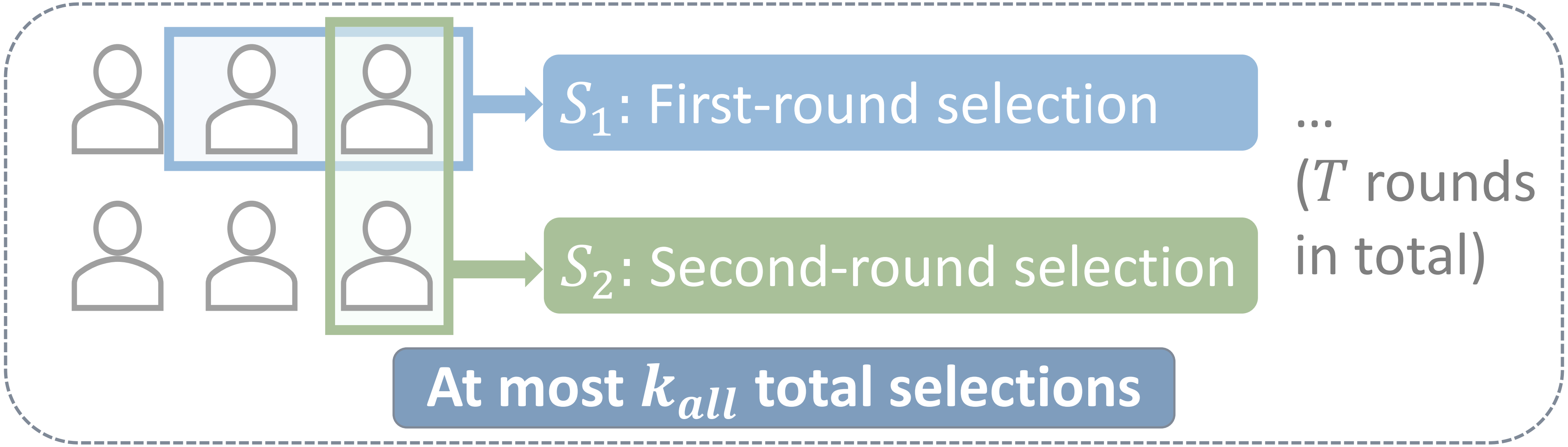


A *(general) matroid* $(U, \mathbb{I})$ specifies:

U: **all users** $\times$ **[T rounds]** $\mathbb{I}$: $\forall \boldsymbol{i}, |\boldsymbol{S_i}| \leq \boldsymbol{k_i}$ **and** $\sum_{i=1}^{T} |\boldsymbol{S_i}| \leq \boldsymbol{k_{all}}$

Why Matroid Constraints

- **Example 3 (Multi-round + an overall budget):**



A *(general) matroid* $(U, \mathbb{I})$ specifies:

U: **all users** $\times$ T **rounds** $\quad \mathbb{I}$: $\forall \boldsymbol{i}, |\boldsymbol{S_i}| \leq \boldsymbol{k_i}$ **and** $\sum_{i=1}^{T} |\boldsymbol{S_i}| \leq \boldsymbol{k_{all}}$

- **Problem definition:**

Maximize $\sigma(S)$ *(expected influence)* s.t. $S \subseteq U, S \in \mathbb{I}$.

Roadmap

- Influence Maximization (IM)
- Why matroid constraints
- **Our idea & guarantees**
- Experiments

TL; DR (Recap)

- We study influence maximization problems with a matroid constraint
- We give a **fast** solution with a $(\mathbf{1 - 1/e - \epsilon})$**-approximation**

Let's get into some math.

Recap of Existing Results

(1) This problem is an instance of **monotone submodular maximization under a matroid constraint** *(thus NP-Hard)*

Recap of Existing Results

(1) This problem is an instance of **monotone submodular maximization under a matroid constraint** *(thus NP-Hard)*

(2) Existing $(1 - 1/e - \epsilon)$-approximation algorithms adopt hill-climbing on the following *multilinear extension* F of $\sigma(\cdot)$:

$$F([x_1 \ \ldots \ x_{|U|}]) = \sum_{S \subseteq U} \prod_{i \in S} x_i \prod_{j \notin S} (1 - x_j) \cdot \sigma(S).$$

Recap of Existing Results

(1) This problem is an instance of **monotone submodular maximization under a matroid constraint** *(thus NP-Hard)*

(2) Existing $(1 - 1/e - \epsilon)$-approximation algorithms adopt hill-climbing on the following *multilinear extension* F of $\sigma(\cdot)$:

$$F([x_1 \ \dots \ x_{|U|}]) = \sum_{S \subseteq U} \prod_{i \in S} x_i \prod_{j \notin S} (1 - x_j) \cdot \sigma(S)$$

- Prior work (Calinescu et al. *SIAM J. Comput.'11*; Badanidiyuru et al. *SODA'14*) *estimates* $F([x_1 \ \dots \ x_{|U|}])$ by sampling, as it involves all $2^{|U|}$ possible S

Gruia Calinescu, Chandra Chekuri, Martin Pal, and Jan Vondrák. 2011. Maximizing a monotone submodular function subject to a matroid constraint. SIAM J. Comput. 40, 6 (2011), 1740–1766.
Ashwinkumar Badanidiyuru and Jan Vondrák. 2014. Fast algorithms for maximizing submodular functions. In SODA. 1497–1514.

Our Idea & Guarantees

(1) Adapt the previous approach (RR sets) to rewrite $\sigma(S)$ to set cover:

$$\sigma(S) \approx \mathrm{C} \cdot \sum_R \mathbf{1}\{S \cap R \neq \emptyset\}$$ *(R denotes RR set; C is constant)*

(2) Hill-climbing on $F(\cdot)$, with **fast & accurate assessment of its value**.

How?

Our Idea & Guarantees

(1) Adapt the previous approach (RR sets) to rewrite $\sigma(S)$ to set cover:

$$\sigma(S) \approx \mathrm{C} \cdot \sum_R \mathbf{1}\{S \cap R \neq \emptyset\}$$ *(R denotes RR set; C is constant)*

(2) Hill-climbing on $F(\cdot)$, with **<u>fast & accurate assessment of its value</u>.**

<u>How?</u> Plug the set cover into $F(\cdot)$:

$$F([x_1 \; \dots \; x_{|U|}]) \approx \mathrm{C} \cdot \sum_R \underbrace{\sum_{S \subseteq U} \prod_{i \in S} x_i \prod_{j \notin S} (1 - x_j)\, \mathbf{1}\{S \cap R \neq \emptyset\}}_{\spadesuit}.$$

Observation: $\spadesuit = (1 - \prod_{i \in R}(1 - x_i))$ **(a computable form)**.

Our Idea & Guarantees

- Example: Multi-Round IM (n users; T rounds in total, k users per round)
- Approximating $\sigma(\cdot)$ needs $L = \sum_R |\mathrm{R}|$ time in total (Tang et al. *SIGMOD'14*)

Algorithm	Time Complexity	Approximation
Calinescu et al. (*SIAM J. Comput.'11*)	$O(n^7 \cdot L)$	$1 - 1/e - \epsilon$
Badanidiyuru et al. (*SODA'14*)	$\mathrm{O}(nkT \cdot \epsilon^{-4} \cdot \mathrm{L})$	
Our algorithm **(AMP)**	$O(k \cdot \epsilon^{-1} \cdot L)$	

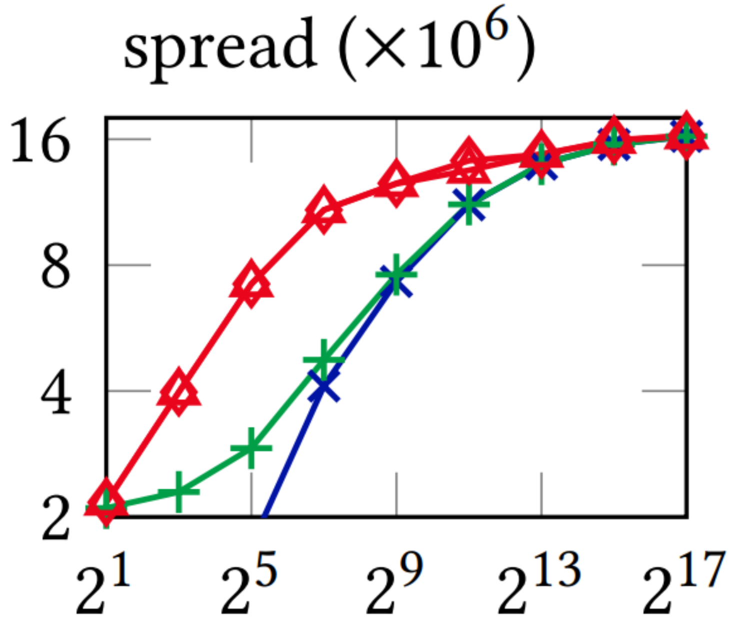
Tang Youze, Xiaokui Xiao, and Yanchen Shi. 2014. Influence maximization: Near-optimal time complexity meets practical efficiency. SIGMOD.

Roadmap

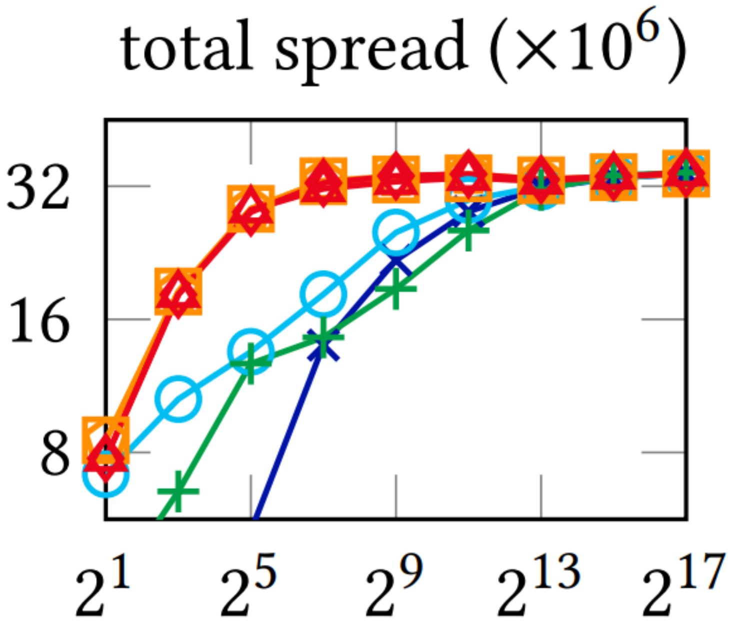
- Influence Maximization (IM)
- Why matroid constraints
- Our idea & guarantees
- **Experiments**

Experiments

- **Offline experiment:** 7 public datasets and 4 problem instances
- Our solution **AMP** (main algorithm *given RR sets;* **red△** & **orange□**):



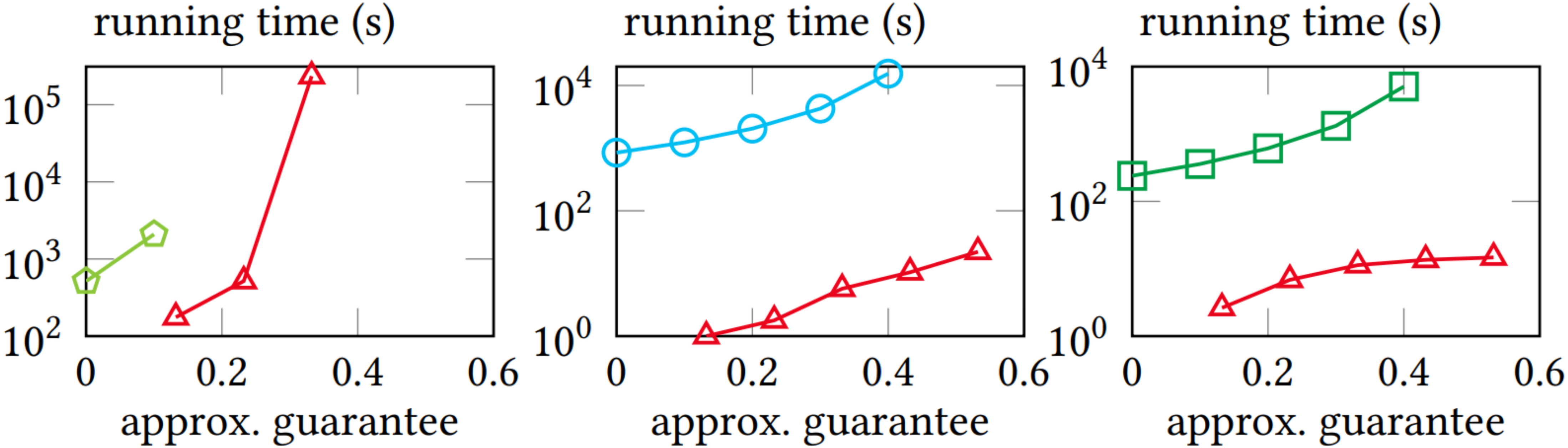
(a) IM



(b) Multi-round IM

Experiments

- **Offline experiment:** 7 public datasets and 4 problem instances
- Our solution **RAMP** (RR set generation + main algorithm; **red△**):



(a) RM (Twitter-large) **(b) MRIM (Wiki)** **(c) AdvIM (Wiki)**

(n=41.7M m=1.5B) (n=7.1K m=103.6K)

Experiments

- **Online deployment:** a Tencent game
- **Task:** recommend in-game products (maps) to each user
- **Scale:** 0.7 million users and 400 maps

Metric: Increase in engaged user–map pairs
(vs. control group, $\mathrm{K} = 10^3$)

Algorithm	RM-A	RAMP (ours)
Δ of pairs	20K	**160K**

Thanks!

Our paper

Definition Of Matroid

- A matroid is a pair $(U, \mathbb{I})$ where U is a finite set and $\mathbb{I}$ is a collection of subsets of U satisfying the following axioms:

1. $\mathbb{I} \neq \emptyset$
2. If $I \in \mathbb{I}$ and $J \subseteq I$, then $J \in \mathbb{I}$
3. If $I, J \in \mathbb{I}$ and $|I| < |J|$, then there exists $\mathrm{x} \in J \setminus I$ such that $I \cup \{\mathrm{x}\} \in \mathbb{I}$