Tight Bounds on the Round Complexity of the Distributed Maximum Coverage Problem

Sepehr Assadi

University of Pennsylvania

Joint work with Sanjeev Khanna (Penn)

Maximum Coverage Problem

Given

- a collection of sets $S_1, \ldots, S_m$ from a universe $[n]$, and
- an integer parameter k

Find

- k sets whose union covers the most number of elements.

Maximum Coverage Problem

Given

- a collection of sets $S_1, \ldots, S_m$ from a universe $[n]$, and
- an integer parameter k

Find

- k sets whose union covers the most number of elements.

- A classical NP-hard optimization problem
- Wide range of applications in various domains
- An illustrative example of submodular maximization

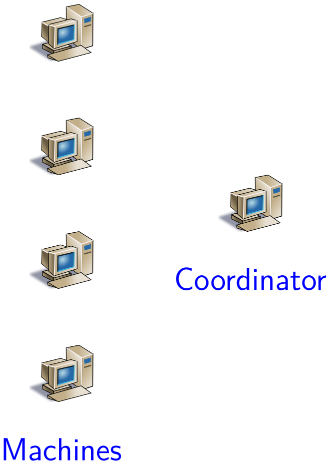
Distributed Maximum Coverage Problem

We are interested in the following distributed model:

Distributed Maximum Coverage Problem

We are interested in the following distributed model:

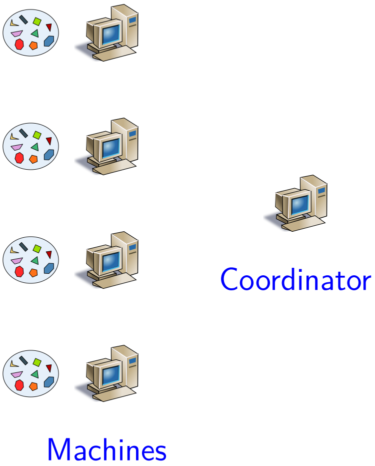
1. There are p machines plus an additional coordinator.



Distributed Maximum Coverage Problem

We are interested in the following distributed model:

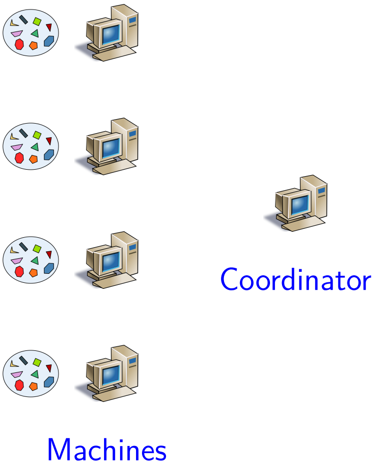
1. There are p machines plus an additional coordinator.
2. Each input set appears in exactly one machine.



Distributed Maximum Coverage Problem

We are interested in the following distributed model:

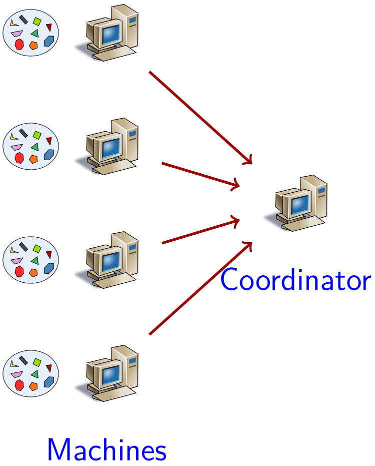
1. Communication happens in rounds. In each round,



Distributed Maximum Coverage Problem

We are interested in the following distributed model:

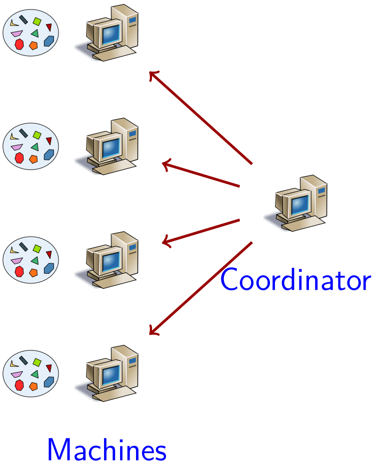
1. Communication happens in rounds. In each round,
   - Every machine simultaneously sends a message to the coordinator.



Distributed Maximum Coverage Problem

We are interested in the following distributed model:

1. Communication happens in rounds. In each round,
   - Every machine simultaneously sends a message to the coordinator.
   - Next, the coordinator responds with a single message to each machine.



Distributed Maximum Coverage Problem

We are interested in the following distributed model:

1. Communication happens in rounds. In each round,
   - Every machine simultaneously sends a message to the coordinator.
   - Next, the coordinator responds with a single message to each machine.
2. After the last round, the coordinator outputs the answer.

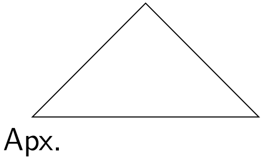
Efficiency

Main measures of efficiency in this model:

Efficiency

Main measures of efficiency in this model:

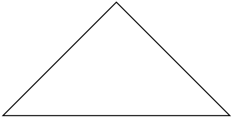
1. Approximation ratio of the returned solution.



Efficiency

Main measures of efficiency in this model:

1. Approximation ratio of the returned solution.
   - Ideally $\left(\frac{e}{e-1}\right)$ approximation.



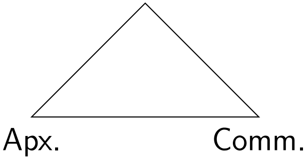
Apx.

Efficiency

Main measures of efficiency in this model:

1. Approximation ratio of the returned solution.
   - Ideally $\left(\frac{e}{e-1}\right)$ approximation.

2. Communication cost of the protocol.

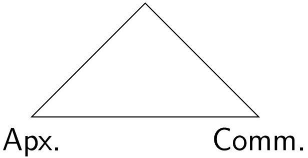


Efficiency

Main measures of efficiency in this model:

1. Approximation ratio of the returned solution.
   - Ideally $\left(\frac{e}{e-1}\right)$ approximation.

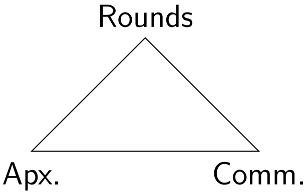
2. Communication cost of the protocol.
   - Ideally $\widetilde{O}(n)$ communication.



Efficiency

Main measures of efficiency in this model:

1. Approximation ratio of the returned solution.
   - Ideally $\left(\frac{e}{e-1}\right)$ approximation.
2. Communication cost of the protocol.
   - Ideally $\widetilde{O}(n)$ communication.
3. Number of rounds of computation.



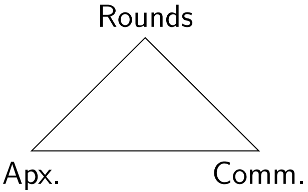
Efficiency

Main measures of efficiency in this model:

1. Approximation ratio of the returned solution.
   - Ideally $\left(\frac{e}{e-1}\right)$ approximation.

2. Communication cost of the protocol.
   - Ideally $\widetilde{O}(n)$ communication.

3. Number of rounds of computation.
   - Ideally $O(1)$ rounds.



Motivation Behind the Model

Why this distributed model?

Motivation Behind the Model

Why this distributed model?

1. A natural abstraction of distributed computing that focuses on number of rounds of parallel computation.

Motivation Behind the Model

Why this distributed model?

1. A natural abstraction of distributed computing that focuses on number of rounds of parallel computation.

2. Closely related to other computational models such as dynamic streams and MapReduce model.

Motivation Behind the Model

Why this distributed model?

1. A natural abstraction of distributed computing that focuses on number of rounds of parallel computation.
2. Closely related to other computational models such as dynamic streams and MapReduce model.
3. Studying powers and limitations of many popular algorithmic techniques such as linear sketching, composable coresets, and sample-and-prune, through the lens of communication complexity.

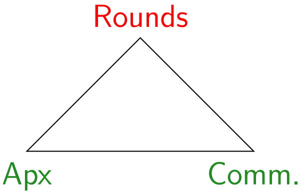
Previous Results

Two main categories:

Previous Results

Two main categories:

> Communication efficient protocols achieving $\left(\frac{e}{e-1}\right)$ approximation in large number of rounds.

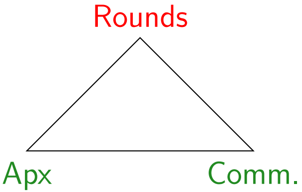


Previous Results

Two main categories:

Communication efficient protocols achieving $\left(\frac{e}{e-1}\right)$ approximation in large number of rounds.

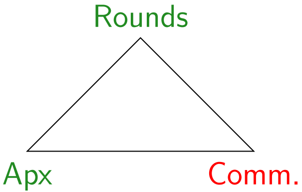
- $\widetilde{O}(n)$ communication and $\Omega(\log n)$ rounds [Kumar et al., 2013, Badanidiyuru et al., 2014, McGregor and Vu, 2017].



Previous Results

Two main categories:

Round efficient protocols achieving $O(1)$-approximation with large communication cost.

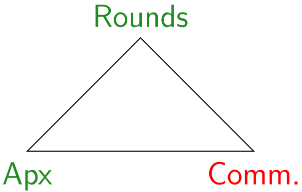


Previous Results

Two main categories:

Round efficient protocols achieving $O(1)$-approximation with large communication cost.

- $O(1)$ rounds and $k \cdot m^{\Omega(1)}$ communication [Kumar et al., 2013].



Motivating Question

Does there exist a truly efficient distributed protocol for maximum coverage, that is, a protocol that achieves $\widetilde{O}(n)$ *communication,* $O(1)$ *rounds, and* $O(1)$ *approximation?*

Motivating Question

Does there exist a truly efficient distributed protocol for maximum coverage, that is, a protocol that achieves $\widetilde{O}(n)$ communication, $O(1)$ rounds, and $O(1)$ approximation?

Any barrier?

Motivating Question

Does there exist a truly efficient distributed protocol for maximum coverage, that is, a protocol that achieves $\widetilde{O}(n)$ communication, $O(1)$ rounds, and $O(1)$ approximation?

Any barrier?

Not possible in one round using a restricted types of algorithms, namely, composable coresets [Indyk et al., 2014].

Motivating Question

Does there exist a truly efficient distributed protocol for maximum coverage, that is, a protocol that achieves $\widetilde{O}(n)$ *communication,* $O(1)$ *rounds, and* $O(1)$ *approximation?*

Any barrier?

Not possible in one round using a restricted types of algorithms, namely, composable coresets [Indyk et al., 2014].

At the same time, almost no multi-round lower bounds are known in this model...

Our Results

Our main result is a negative resolution of this question.

Our Results

Our main result is a negative resolution of this question.

Theorem

Any $\mathrm{poly}(n)$ *communication protocol that achieves* $O(1)$ *approximation requires* $\Omega\left(\frac{\log n}{\log\log n}\right)$ *rounds of communication.*

Our Results

Our main result is a negative resolution of this question.

Theorem

Any $\mathrm{poly}(n)$ *communication protocol that achieves* $O(1)$ *approximation requires* $\Omega\left(\frac{\log n}{\log\log n}\right)$ *rounds of communication.*

In general,

Theorem

For any integer $r \geq 1$*, any* r*-round protocol for distributed maximum coverage either incurs* $k \cdot m^{\Omega(1/r)}$ *communication per machine or has approximation ratio* $k^{\Omega(1/r)}$ *(here* k *and* n *are polynomially related).*

Our Results

We complement our lower bound by proving that its bounds are almost tight.

Our Results

We complement our lower bound by proving that its bounds are almost tight.

Theorem

For any integer $r \geq 1$, there are r-round protocols for distributed maximum coverage that achieve,

1. $\left(\frac{e}{e-1}\right)$*-approximation with* $k \cdot m^{O(1/r)}$ *communication,*
2. $O(r \cdot k^{1/r+1})$*-approximation with* $\widetilde{O}(n)$ *communication.*

Further Applications

Our results imply new algorithms and lower bounds for dynamic streams and MapReduce model.

Further Applications

Our results imply new algorithms and lower bounds for dynamic streams and MapReduce model.

1. An $\Omega(\log n)$-pass lower bound for $O(1)$-approximation semi-streaming algorithms in dynamic streams.

Further Applications

Our results imply new algorithms and lower bounds for dynamic streams and MapReduce model.

1. An $\Omega(\log n)$-pass lower bound for $O(1)$-approximation semi-streaming algorithms in dynamic streams.
   - In contrast, $O(1)$-approximation single-pass semi-streaming algorithms exists in insertion-only streams and even sliding windows [Badanidiyuru et al., 2014, McGregor and Vu, 2017, Chen et al., 2016, Epasto et al., 2017].

Further Applications

Our results imply new algorithms and lower bounds for dynamic streams and MapReduce model.

1. An $\Omega(\log n)$-pass lower bound for $O(1)$-approximation semi-streaming algorithms in dynamic streams.
    - In contrast, $O(1)$-approximation single-pass semi-streaming algorithms exists in insertion-only streams and even sliding windows [Badanidiyuru et al., 2014, McGregor and Vu, 2017, Chen et al., 2016, Epasto et al., 2017].
2. An improved $\left(\frac{e}{e-1}\right)$-approximation algorithm in the MapReduce model.

The Lower Bound

Theorem

For any integer $r \geq 1$, any r-round protocol for distributed maximum coverage either incurs $k \cdot m^{\Omega(1/r)}$ communication per machine or has approximation ratio $k^{\Omega(1/r)}$.

High Level Approach

First Part:

1. Design a hard input distribution for one-round protocols.

High Level Approach

First Part:

1. Design a hard input distribution for one-round protocols.
   - Each machine has one special set to contribute to the optimum solution.

High Level Approach

First Part:

1. Design a hard input distribution for one-round protocols.
   - Each machine has one special set to contribute to the optimum solution.
   - This special set is hidden in the large collection of input sets to this machine.

High Level Approach

First Part:

1. Design a hard input distribution for one-round protocols.
   - Each machine has one special set to contribute to the optimum solution.
   - This special set is hidden in the large collection of input sets to this machine.
   - To convey information about its special set, each machine needs to convey information about most of its input sets.

High Level Approach

First Part:

1. Design a hard input distribution for one-round protocols.
    - Each machine has one special set to contribute to the optimum solution.
    - This special set is hidden in the large collection of input sets to this machine.
    - To convey information about its special set, each machine needs to convey information about most of its input sets.
2. Prove a communication lower bound for this distribution.

High Level Approach

First Part:

1. Design a hard input distribution for one-round protocols.
   - Each machine has one special set to contribute to the optimum solution.
   - This special set is hidden in the large collection of input sets to this machine.
   - To convey information about its special set, each machine needs to convey information about most of its input sets.

2. Prove a communication lower bound for this distribution.
   - We use information theoretic machinery to analyze this distribution.

High Level Approach

Second Part:

We apply the previous idea recursively:

High Level Approach

Second Part:

We apply the previous idea recursively:

1. We first create many instances of the coverage problem on a smaller universe with fewer machines.

High Level Approach

Second Part:

We apply the previous idea recursively:

1. We first create many instances of the coverage problem on a smaller universe with fewer machines.
2. Each machine is participating in many such instances among which, one is special but unknown to the machine.

High Level Approach

Second Part:

We apply the previous idea recursively:

1. We first create many instances of the coverage problem on a smaller universe with fewer machines.
2. Each machine is participating in many such instances among which, one is special but unknown to the machine.
3. We pack all these instances into one larger instance of the coverage problem such that solving special instances is necessary for any efficient solution.

High Level Approach

Second Part:

We apply the previous idea recursively:

1. We first create many instances of the coverage problem on a smaller universe with fewer machines.
2. Each machine is participating in many such instances among which, one is special but unknown to the machine.
3. We pack all these instances into one larger instance of the coverage problem such that solving special instances is necessary for any efficient solution.
4. As special instances are hidden in the first round, one needs to solve them in the remaining rounds which is hard by induction.

High Level Approach

Second Part:

We apply the previous idea recursively:

1. We first create many instances of the coverage problem on a smaller universe with fewer machines.
2. Each machine is participating in many such instances among which, one is special but unknown to the machine.
3. We pack all these instances into one larger instance of the coverage problem such that solving special instances is necessary for any efficient solution.
4. As special instances are hidden in the first round, one needs to solve them in the remaining rounds which is hard by induction.

Main tool: a generalization of the multi-party round-elimination technique of [Alon et al., 2015].

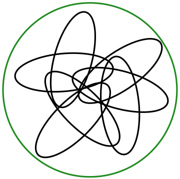
Analysis Sketch for r-Round Protocols

To prove a lower bound for r-round protocols, we create instances with the following parameters:

1. Number of elements is n_r.
2. Number of input sets is $n_r^{O(r)}$.
3. Parameter k_r and number of machines p_r are equal to each other.

Analysis Sketch for r-Round Protocols

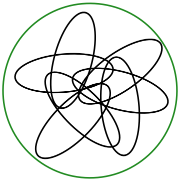
- $\mathcal{S}$ is a large collection of sets each of size n_{r-1} over $\ll n_r$ elements.



Collection $\mathcal{S}$

Analysis Sketch for r-Round Protocols

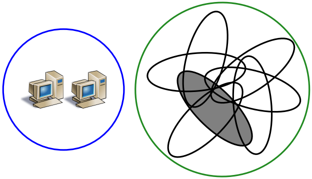
- $\mathcal{S}$ is a large collection of sets each of size n_{r-1} over $\ll n_r$ elements.
- Machines are partitioned into blocks, each of size p_{r-1}.



Collection $\mathcal{S}$

Analysis Sketch for r-Round Protocols

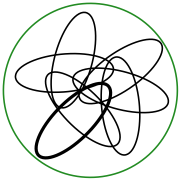
- $\mathcal{S}$ is a large collection of sets each of size n_{r-1} over $\ll n_r$ elements.
- Machines are partitioned into blocks, each of size p_{r-1}.
- Machines in each block are playing in $|\mathcal{S}|$ many instances of $(r-1)$-round problem, each over a universe $S_j \in \mathcal{S}$.



A block of machines and a universe $S_j \in \mathcal{S}$

Analysis Sketch for r-Round Protocols

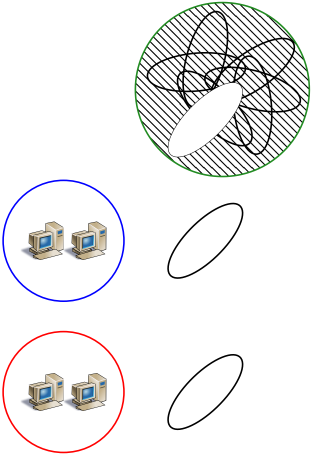
- $\mathcal{S}$ is a large collection of sets each of size n_{r-1} over $\ll n_r$ elements.
- Machines are partitioned into blocks, each of size p_{r-1}.
- Machines in each block are playing in $|\mathcal{S}|$ many instances of $(r-1)$-round problem, each over a universe $S_j \in \mathcal{S}$.
- We pick one of the instances in $\mathcal{S}$ as special uniformly at random.



Special instance

Analysis Sketch for r-Round Protocols

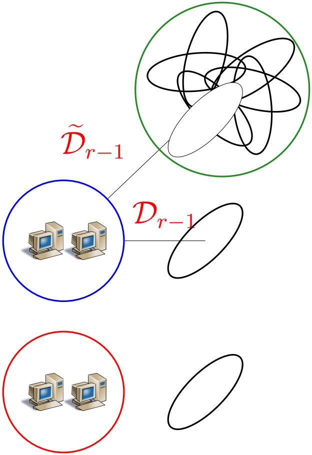
- $\mathcal{S}$ is a large collection of sets each of size n_{r-1} over $\ll n_r$ elements.
- Machines are partitioned into blocks, each of size p_{r-1}.
- Machines in each block are playing in $|\mathcal{S}|$ many instances of $(r-1)$-round problem, each over a universe $S_j \in \mathcal{S}$.
- We pick one of the instances in $\mathcal{S}$ as special uniformly at random.
- Across the blocks, elements in special instance are unique, while other elements are shared.



Global view

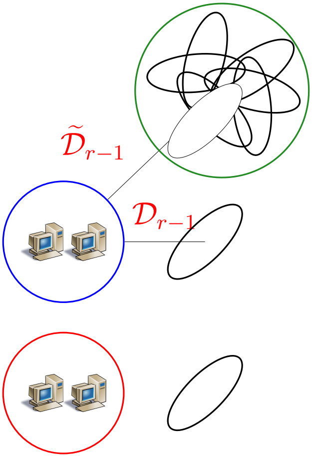
Analysis Sketch for r-Round Protocols

1. The machines need to solve the $(r-1)$-round instance between their blocks and their special instance.



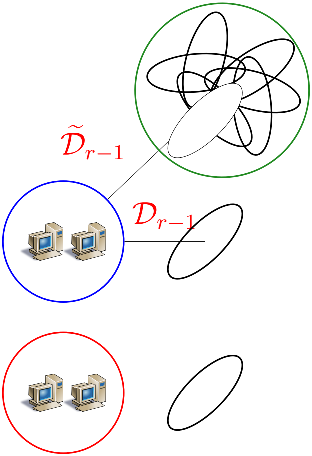
Analysis Sketch for r-Round Protocols

1. The machines need to solve the $(r-1)$-round instance between their blocks and their special instance.
2. The first message M of a low communication cost protocol π does not reveal any useful information about the special instance.



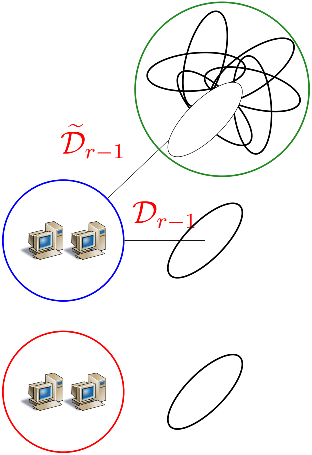
Analysis Sketch for r-Round Protocols

1. The machines need to solve the $(r-1)$-round instance between their blocks and their special instance.
2. The first message M of a low communication cost protocol π does not reveal any useful information about the special instance.
3. If π can solve $\mathcal{D}_r$ in r rounds, then $\pi \mid M$ should be able to solve $\mathcal{D}_{r-1}$ in $r-1$ rounds.



Analysis Sketch for r-Round Protocols

1. The machines need to solve the $(r-1)$-round instance between their blocks and their special instance.
2. The first message M of a low communication cost protocol π does not reveal any useful information about the special instance.
3. If π can solve $\mathcal{D}_r$ in r rounds, then $\pi \mid M$ should be able to solve $\mathcal{D}_{r-1}$ in $r-1$ rounds.
4. We can obtain a low communication cost protocol π' for solving $\mathcal{D}_{r-1}$ in $r-1$ rounds by simulating $\pi \mid M$.



Analysis Sketch for r-Round Protocols

What is left to prove?

Analysis Sketch for r-Round Protocols

What is left to prove?

- The machines really have to solve their special instance.

Analysis Sketch for r-Round Protocols

What is left to prove?

- The machines really have to solve their special instance.
- We need to design the set $\mathcal{S}$ carefully to satisfy this property.

Analysis Sketch for r-Round Protocols

What is left to prove?

- The machines really have to solve their special instance.
- We need to design the set $\mathcal{S}$ carefully to satisfy this property.

We achieve this using a randomly generated set-system in the spirit of the edifice construction of [Chakrabarti and Wirth, 2016].

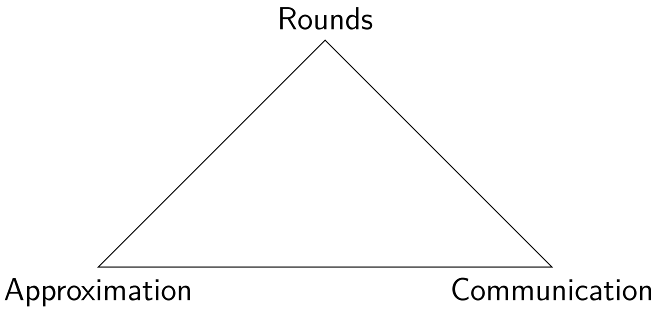
Analysis Sketch for r-Round Protocols

By optimizing the ratio between the parameters, we obtain:

A lower bound of $k_r \cdot m_r^{\Omega(1/r)}$ communication for $\approx k_r^{1/2r}$ approximation in r rounds.

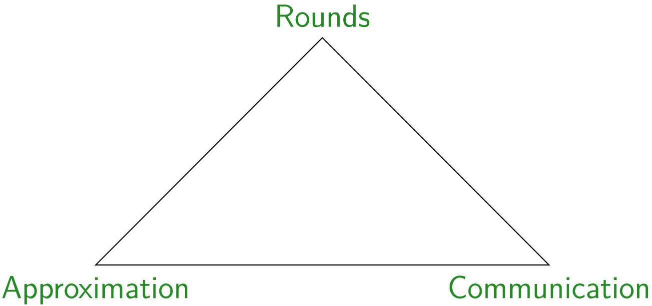
Summary

The efficiency triangle:



Summary

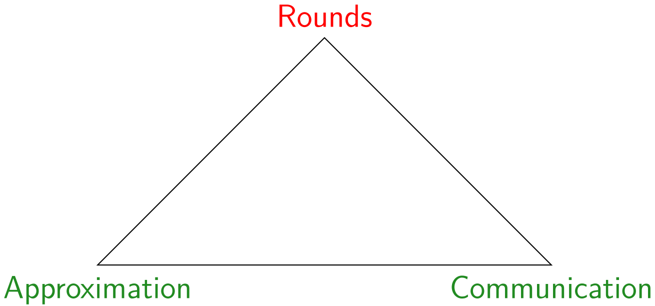
The efficiency triangle:



This paper: Impossible to be efficient in all three measures simultaneously!

Summary

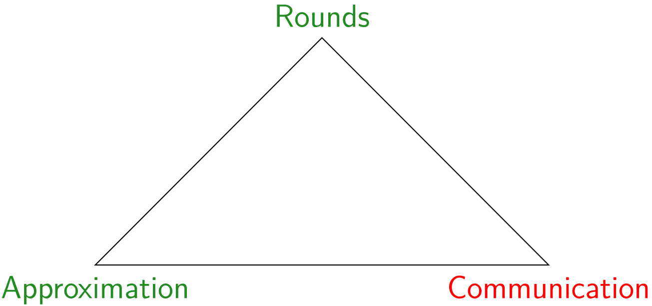
The efficiency triangle:



[Badanidiyuru et al., 2014, McGregor and Vu, 2017].

Summary

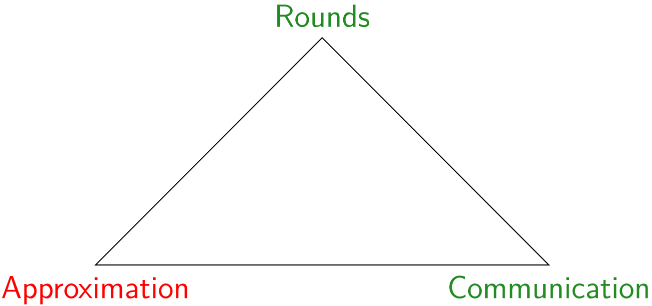
The efficiency triangle:



[Kumar et al., 2013] and this paper.

Summary

The efficiency triangle:



This paper.

Open Problems

- Round-approximation tradeoffs for other distributed problems.

Open Problems

- Round-approximation tradeoffs for other distributed problems.
  - Approximating maximum matching?

Open Problems

- Round-approximation tradeoffs for other distributed problems.
  - Approximating maximum matching?
- Dynamic semi-streaming algorithms for maximum coverage in constant number of rounds.

Open Problems

- Round-approximation tradeoffs for other distributed problems.
  - Approximating maximum matching?
- Dynamic semi-streaming algorithms for maximum coverage in constant number of rounds.
  - Our results imply $\left(\frac{e}{e-1}\right)$-approximation in $O(\log n)$ passes.
  - $k^{O(1/r)}$-approximation in r-passes?
  - $O(\sqrt{k})$-approximation in one pass?

Open Problems

- Round-approximation tradeoffs for other distributed problems.
  - Approximating maximum matching?
- Dynamic semi-streaming algorithms for maximum coverage in constant number of rounds.
  - Our results imply $\left(\frac{e}{e-1}\right)$-approximation in $O(\log n)$ passes.
  - $k^{O(1/r)}$-approximation in r-passes?
  - $O(\sqrt{k})$-approximation in one pass?

Thank you!

Alon, N., Nisan, N., Raz, R., and Weinstein, O. (2015).
Welfare maximization with limited interaction.
In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 1499–1512.

Badanidiyuru, A., Mirzasoleiman, B., Karbasi, A., and Krause, A. (2014).
Streaming submodular maximization: massive data summarization on the fly.
In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 671–680.

Chakrabarti, A. and Wirth, A. (2016).
Incidence geometries and the pass complexity of semi-streaming set cover.

In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 1365–1373.

Chen, J., Nguyen, H. L., and Zhang, Q. (2016).
Submodular maximization over sliding windows.
CoRR, abs/1611.00129.

Epasto, A., Lattanzi, S., Vassilvitskii, S., and Zadimoghaddam, M. (2017).
Submodular optimization over sliding windows.
In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, pages 421–430.

Indyk, P., Mahabadi, S., Mahdian, M., and Mirrokni, V. S. (2014).
Composable core-sets for diversity and coverage maximization.

In *Proceedings of the 33rd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS'14, Snowbird, UT, USA, June 22-27, 2014*, pages 100–108.

Kumar, R., Moseley, B., Vassilvitskii, S., and Vattani, A. (2013).
Fast greedy algorithms in mapreduce and streaming.
In *25th ACM Symposium on Parallelism in Algorithms and Architectures, SPAA '13, Montreal, QC, Canada - July 23 - 25, 2013*, pages 1–10.

McGregor, A. and Vu, H. T. (2017).
Better streaming algorithms for the maximum coverage problem.
In *20th International Conference on Database Theory, ICDT 2017, March 21-24, 2017, Venice, Italy*, pages 22:1–22:18.