Cost sharing methods for Stochastic Optimization

Stefano Leonardi

"Sapienza" University of Rome

ADFOCS 2008

August 18 - 22, 2008

Outline

- Two stage stochastic optimization
  (Gupta, Pál, Ravi and Sinha, 2004)
- Online Stochastic Optimization
  (Garg, Gupta, L. & Sankowski, 2008)
- Conclusions

Two-stage Stochastic Optimization

Stochastic optimization

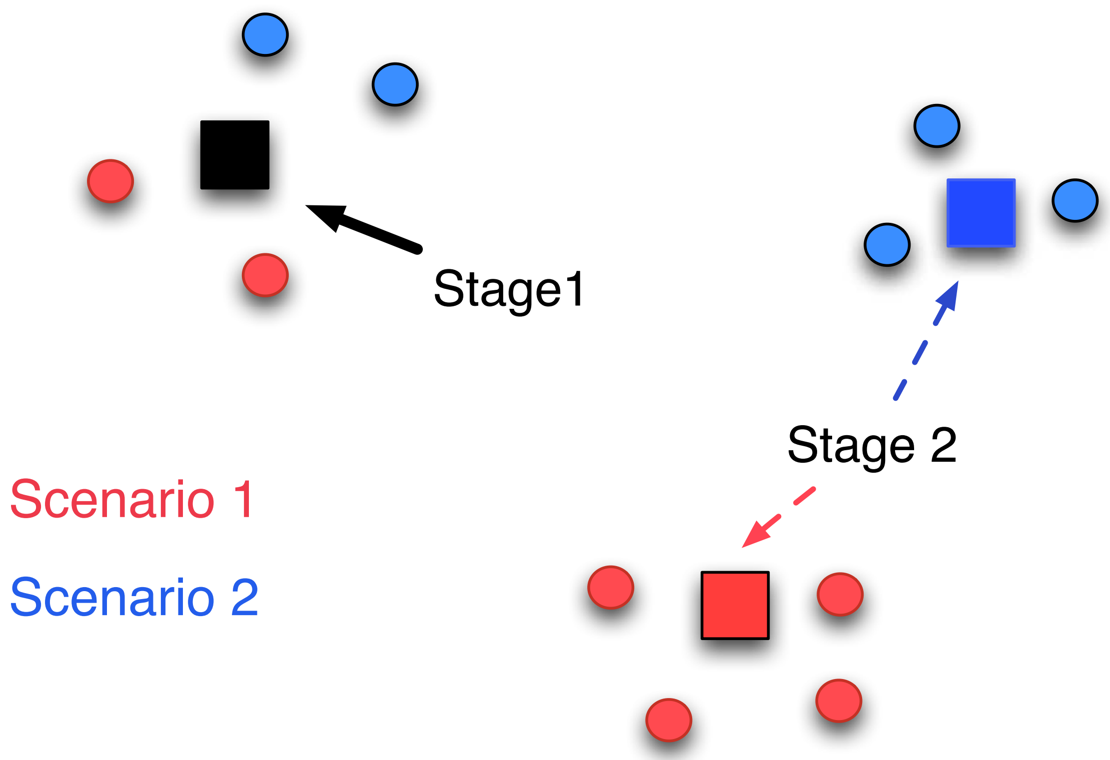
- Classical optimization assumes deterministic inputs
- Stochastic optimization handles uncertainty in data
- Modeled by a probability distribution π over actual realizations of data, called *scenarios*
- Stochastic linear programming dates back to early stages of optimization [Dantzig '55, Beale '61]
- We rather consider a class of stochastic integer combinatorial optimization problems

Two-stage stochastic optimization

- Two stages of decision making, with limited information in the first stage
- Probability distribution governing second-stage data and costs given in the first stage
- Solution can always be made feasible in the second stage
- Building in advance reduces costs, but not enough information is given
- Building in second stage when the actual scenario materializes is more expensive

Example: Facility location

- In a first stage open some facilities with opening costs f_i.
- When demand points materialize, open more facilities with inflated cost σf_i.



Modeling uncertainty

We assume the black-box model: The algorithm has the ability to sample from an *arbitrary* probability distribution

Other models also considered:

- K different scenarios part of the input [Karger & Minkoff00]. Also in the context of solving stochastic linear programs [Ravi & Sinha 03], [Immorlica, Karger, Minkoff & Mirrokni 04].
- Demand request j shows up independently with some probability p_j [Immorlica et al, 2004].

The model:

Combinatorial optimization problem Π:

- U: universe of clients
- X: set of elements we can purchase
- For a set $F \subseteq X$, $c(F)$: cost of element set F
- Given $S \subseteq U$, $Sols(S) \subseteq 2^X$ be the set of feasible solutions for client set S
- $OPT(S)$: solution in $Sols(S)$ of minimum cost
- $\sigma \geq 1$: inflation parameter
- Probability distribution $\pi : 2^U \to [0, 1]$

The model (contd)

First stage:

- An oracle draws samples from π in $poly(U)$
- Compute a first stage solution F_0 with cost $c(F_0)$

Second stage:

- One scenario $S \subseteq U$ materializes.
- Purchase a second stage solution F_S at cost $\sigma c(F_S)$ such that $F_0 \cup F_S \in Sols(S)$

Minimize the expected cost of the solution:

$$c(F_0) + \sum_{S \subseteq U} \pi(S) \sigma c(F_S)$$

Requirements

We require Π to satisfy subadditivity:

- For any $S, S' \subseteq U$, $S' \cup S'$ is also a valid set of clients for Π
- $Sols(S) \cup Sols(S') \in Sols(S \cup S')$

Steiner tree with connection to a root r is subadditive

If we drop the root r, e.g. connect all nodes in S, we loose subadditivity

Boost-and-Sample(Π)

An α approximate algorithm $\mathcal{A}$ used to compute the first stage solution

Augmentation algorithm $Aug_{\mathcal{A}}$ used to compute the second stage solution.

1. Draw σ independent samples $D_1, \ldots, D_\sigma$ from π. Let $D = \cup_i D_i$.
2. Use $\mathcal{A}$ to construct an α-approximate first stage solution $F_0 \in sols(D)$
3. Use $Aug_{\mathcal{A}}$ to compute F_S such that $F_0 \cup F_S \in Sols(S)$.

Theorem

Theorem 1 *If i) Π is subadditive and ii) α-apx alg. $\mathcal{A}$ for Π admits group β-strict cost-shares then exists an $\alpha + \beta$-apx for two-stage stochastic Π.*

Group β-strictness

$\xi : 2^U \times U \rightarrow \Re_{\geq 0}$ is a group β-strict cost-sharing function with respect to $\mathcal{A}$ if

1. For a set $S \subseteq U$, $\xi(S, j) > 0$ only for $j \in S$.
2. For a set $S \subseteq U$, $\sum_{j \in S} \xi(S, j) \leq c(OPT(S))$
3. If $S' = S \cup T$, then $\sum_{j \in T} \xi(S', j) \geq (1/\beta)$ x cost $(Aug_{\mathcal{A}}(S'/S))$

Applications: Steiner tree

Steiner tree: Prim's cost-shares are group 2-strict. Implies a 4-apx algorithm for 2-stage stochastic steiner tree.

Steiner t: FKLS cost-shares are 3 strict (not group strict). Implies a 5-apx for 2-stage stochastic Steiner forest in the independent decision model, e.g. terminal pair j shows up independently with probability p_j

Applications: Facility location

Metric facility location: Mettu-Plaxton algorithm for facility location is 3-apx. Pal-Tardos costs shares are 5.45 group strict for Mettu-Plaxton algorithm. Implies a 8.45 apx algorithm for metric facility location.

Vertex cover: There exists a 2-apx, 6-strict algorithm for vertex cover.

Proof of main Theorem

Bounding first stage cost by $\alpha E[OPT]$:

- Let $F_1 = F_0^* + F_{D_1}^* + \ldots + F_{D_\sigma}^*$
- By subadditivity, $F_1 \in Sols(D)$ and $E[c(F_0)] \leq \alpha E_D[c(F_1)]$
- Let $Z^* = c(F_0^*) + \sum_S \pi(S)\sigma c(F_S^*)$

$$
\begin{aligned}
E_D[c(F_1)] &\leq c(F_0^*) + E_D[\sum_{i=1}^{\sigma} c(F_{D_i}^*)] \\
&= c(F_0^*) + \sum_{i=1}^{\sigma} E_D[c(F_{D_i}^*)] \\
&= c(F_0^*) + \sigma \sum_S \pi(S) c(F_S^*) = Z^*
\end{aligned}
$$

Proof of main Theorem

Bounding second stage cost by $\beta E[OPT]$:

On lines similar to the bound of rental cost for Rent-or-buy network design

The argument needs to be extended to group strictness

Extensions

- Deal with non-uniform inflation factors [Gupta et al. 2007].
- Convert an arbitrary deterministic LP based approximation into a 2-stage stochastic approximation. [Schmoys and Swamy, 2004]
- Solve stochastic linear programming in the black-box model [Schmoys and Swamy, 2006].
- Multi-stage stochastic optimization.

On-line Stochastic Optimization

On-line Stochastic Optimization

We assume that the requests to the algorithm come from the universe U that is equipped with probability distribution $\pi : U \to [0,1]$. The input sequence ω is generated by drawing k independent requests from π.

Online model: the sequence ω is revealed element by element.

Approximation Ratio

In the stochastic case the aim is to minimize the expected cost payed by the algorithm.

Hence, we define the stochastic approximation ratio as:

$$\mathsf{SApx}(\mathsf{ALG}) = \max_{\pi} \max_{k} \frac{\mathbf{E}_{\omega \in \pi^k, r}[\mathsf{ALG}(\omega, r)]}{\mathbf{E}_{\omega \in \pi^k}[\mathsf{OPT}(\omega)]},$$

where:

- r denote the random choices of the algorithm.
- $\mathsf{OPT}(\omega)$ is the optimal (offline) solution.

Online and Stochastic Online

The *online algorithm* must satisfy an unpredictable sequence of requests, completing each request without being able to see the future.

The *online stochastic algorithm* knows that the sequence of requests comes from π. Additionally, it might be given knowledge of:

- distribution π — assumed here,
- sequence length k — is *length-aware*,
  - otherwise is *length-oblivious*.

Online Stochastic Steiner Tree

In the *online Steiner tree problem*:

- we are given a graph $G = (V, E)$,
- a root vertex r,
- and edge costs/lengths $c : E \longrightarrow \Re_{\geq 0}$,
- the request are vertices $v_1, v_2, \ldots$ from V
  - that need to be connected to r,
- the decisions are irrevocable.

(Garg, Gupta, L. & Sankowski, 2008)

Greedy Algorithm

In the classical online setting the best solution is greedy algorithm, i.e.,

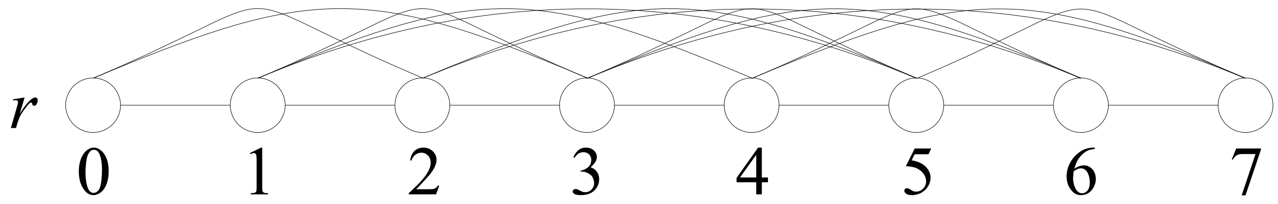
- we connect each request using the shortest path.

The greedy algorithm is $\Theta(\log n)$ competitive.

It remains as bad even when we are in the stochastic setting.

Greedy Algorithm

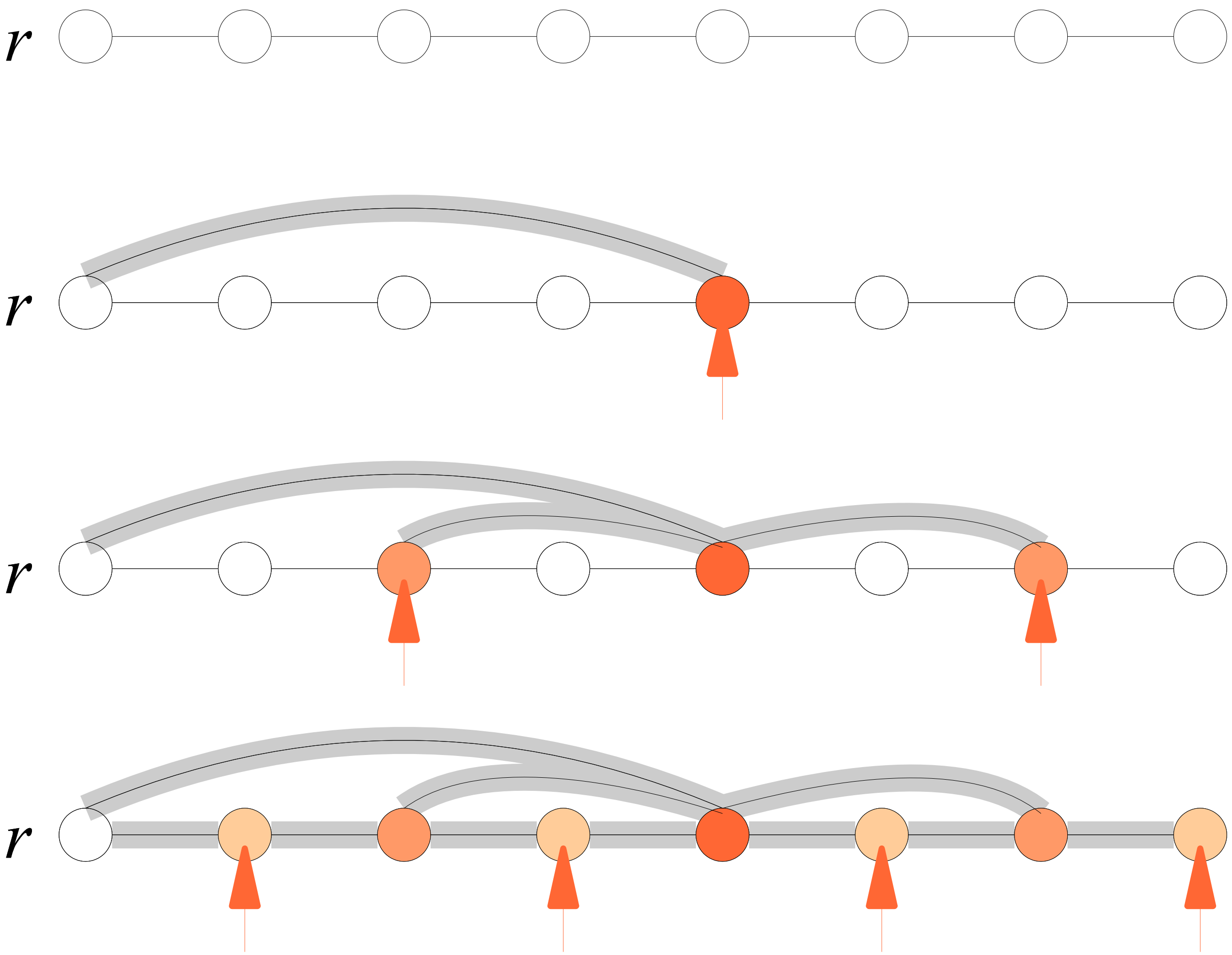
Consider a path of length n with unit length edges, with the vertices numbered $0, 1, \ldots, n$, vertex 0 being the root, and each other vertex having $\pi(i) = 1/n$.



Between each pair of nonadjacent vertices add an edge $\{i, j\}$ having length $\ell_{ij} = |i - j| - (i - j)^2/n^3$.

This ensures that the shortest path between any pair of nodes is the "short-cut" edge between them.

Greedy Algorithm



Greedy Algorithm

Now if we pick input nodes from π, then:

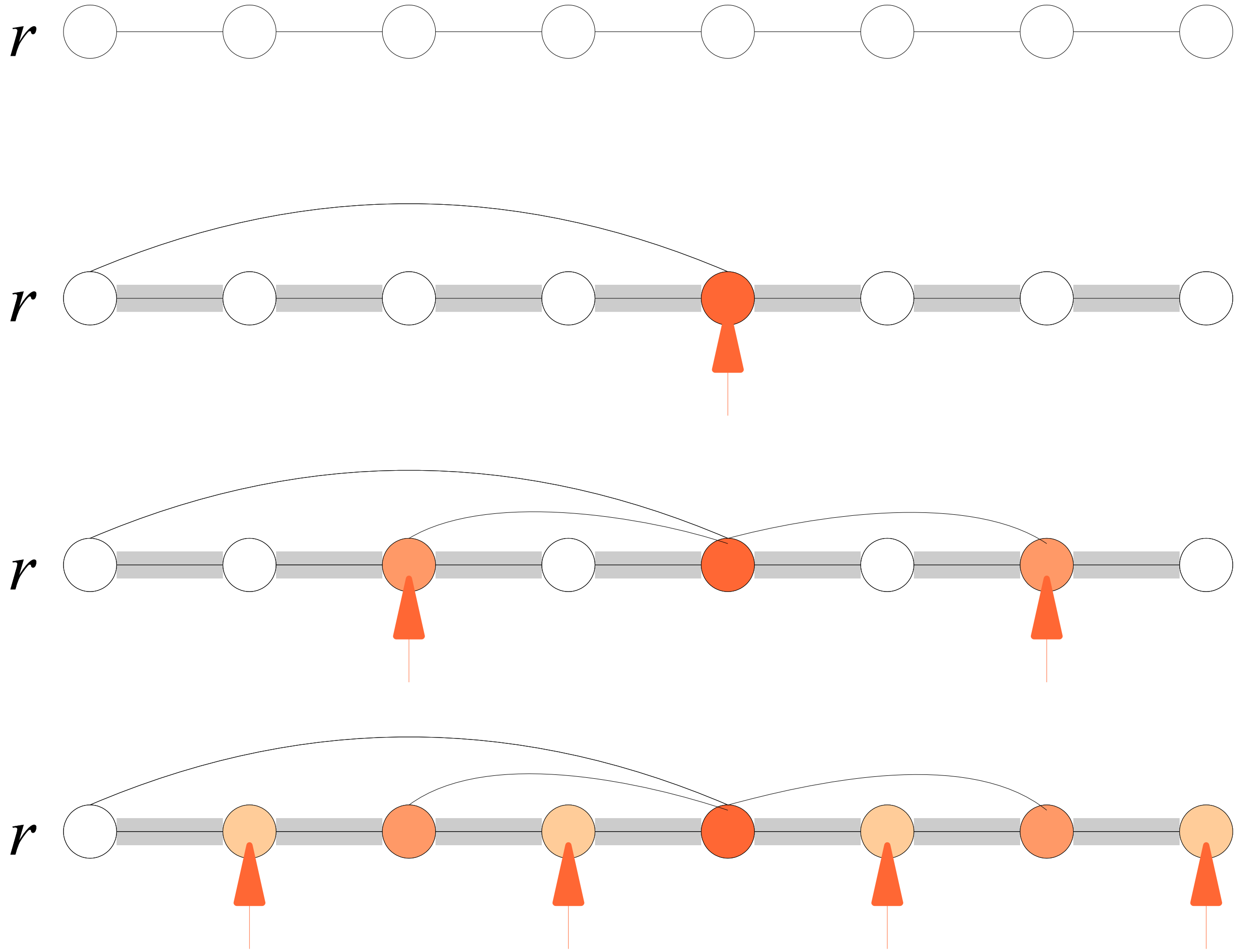
- the expected distance from t^{th} node to closest node is approximately $n/2t$.
- the greedy always buys "short-cut" edge.

We get expected cost of $\Theta(n \log k)$ for k vertices.

Even the completely naïve strategy of buying the *entire* path when the first input point arrives does well, i.e.,

- it costs n, which results in a $\mathsf{SApx} = n/(n/2) = 2$.

Naïve Algorithm



Doing Better than Greedy

Solution is motivated by the naive strategy.

At the beginning we build an ”dummy” solution, such that:

- its expected cost is related to the expected OPT,
- connection cost of the online vertices is related to the expected OPT.

We show that a Steiner tree on k random point is a good anticipatory solution.

Length-aware Algorithm

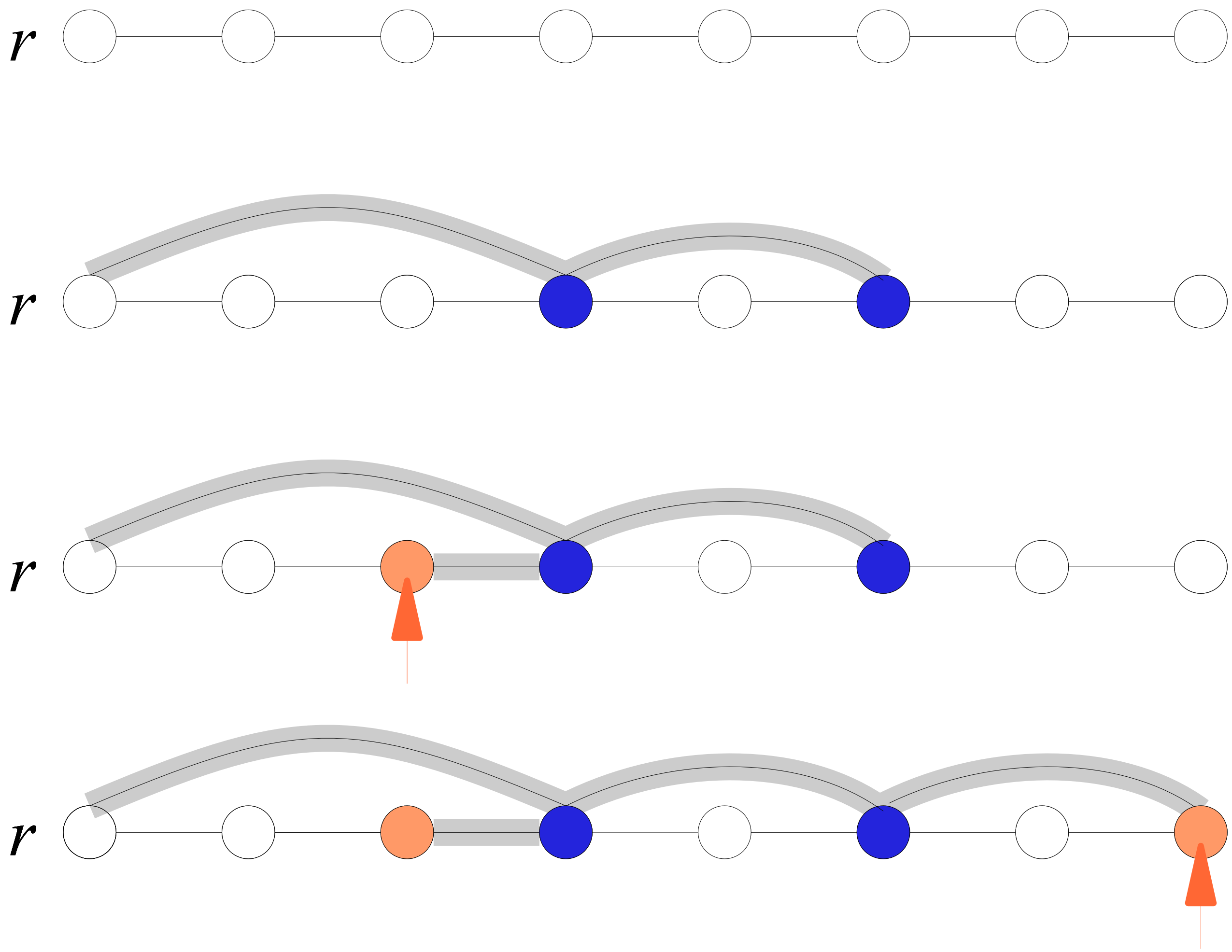
Assume that the sequence length k is known. Consider the following algorithm:

A1. Choose a set of vertices D by drawing from the distribution π independently k times.

A2. Construct a 2-approximate Steiner tree T_M over the set $D \cup \{r\}$.

A3. Run the *greedy algorithm* on actual nodes —namely, connect each input vertex to the closest vertex in the current tree.

The Algorithm



Length-aware Algorithm

Theorem 2 (Steiner Tree Lenght-aware) *The* SApx *ratio of the above algorithm is* 4*.*

The cost of the optimal Steiner tree on D has expected cost $\mathbf{E}[\mathsf{OPT}(\omega)]$, and hence the cost of the anticipatory solution is at most $2\mathbf{E}[OPT(\omega)]$.

Connection Cost

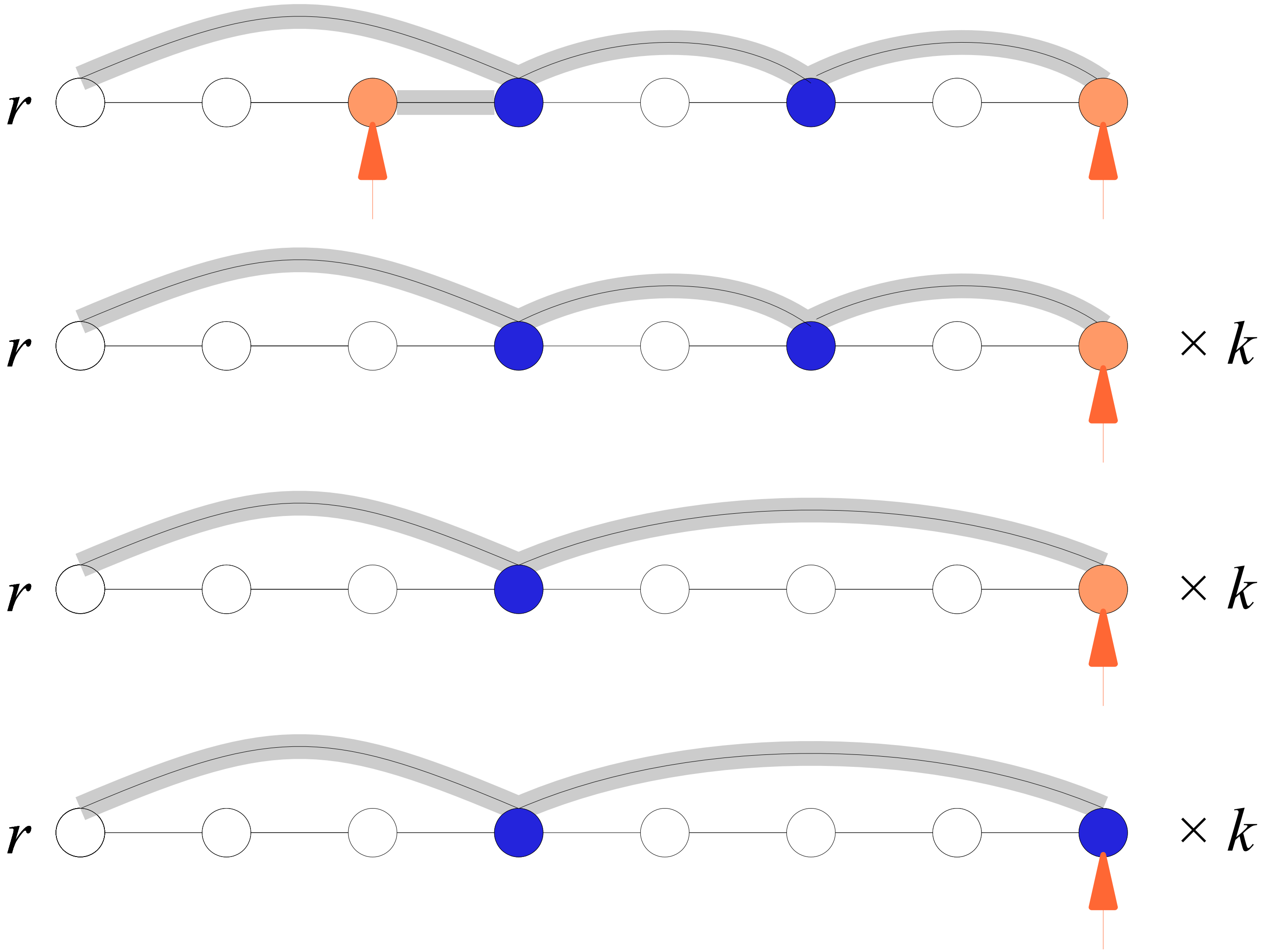
Theorem 3 *The expected cost of greedy connections is* $2\mathbf{E}[OPT(\omega)]$.

We bound $\mathbf{E}_{D\in\pi^k,\omega\in\pi^k}\left[\sum_{v\in\omega} c(v, D\cup\{r\})\right]$ by:

$$\begin{aligned}
& k\times\mathbf{E}_{v\in\pi,D\in\pi^k}[c(v, D\cup\{r\})] \\
\leq\ & k\times\mathbf{E}_{v\in\pi,D'\in\pi^{k-1}}[c(v, D'\cup\{r\})] \\
\leq\ & k\times\mathbf{E}_{D''\in\pi^k}[\frac{1}{k}\cdot MST(D''\cup\{r\})] \\
=\ & \mathbf{E}_{D\in\pi^k}[MST(D\cup\{r\})]
\end{aligned}$$

We can see v as a random vertex of D'' and charge it with the connection to its parent node in the MST.

Connection Cost



Length-oblivious Algorithm

Now we will remove the assumption that we know the number of points in the input sequence.

The idea of the algorithm is to scale on the expected cost of the solution.

In each scale we build an anticipatory solution costing about twice as much as before, and wait until we see as many vertices as in that solution.

Unknown Sequence Length

The cost in the last scale is at most 8, because the expected cost of anticipatory solution is at most two times bigger then the cost of k vertices.

Hence, the cost of all the scales is no more then 16 times the expected cost of optimal solution on k vertices.

Theorem 4 (Unknown Sequence Length) *There exists polynomial time length-oblivious algorithm for stochastic online Steiner tree with* $\mathsf{SApx} = O(1)$.

Other Results

One can show that some access to the distribution, and also the independence is necessary, i.e.:

- if the input consists of i.i.d. draws from a fixed but unknown distribution,
- or if the draws are not independent but drawn from some given Markov chain,

then there is an $\Omega(\frac{\log n}{\log \log n})$ lower bound for Steiner tree.

Can be generalized to subadditive problems provided that there exists a β-strict cost sharing methods for an α-approximate anticipatory solution: Uncapacitated Facility Location, Steiner Forest and Vertex Cover.

Expectation of Ratios

An objective function that is often more challenging to work with is the *expected ratio* (EoR):

$$\mathsf{EoR}(\mathsf{ALG}) = \max_{\pi} \max_{k} \mathbf{E}_{\omega \in \pi^k} \left[\frac{\mathbf{E}_r[\mathcal{A}(\omega, r)]}{\mathsf{OPT}(\omega)} \right].$$

We will again consider the case when the length of the input sequence is given to the algorithm.

Expectation of Ratios

Consider the following algorithm run in parallel with the greedy algorithm:

1. Sample L different k element multisets $D_1, \ldots, D_L$ from the distribution π.
2. For each i, find ρ-approximate Steiner tree T_i on the set $D_i \cup \{r\}$, but do not buy these edges.
3. Choose i^* such that the cost of T_{i^*} is the least, and buy these edges: i.e., set $S_0 = T_{i^*}$.
4. Connect the k input vertices greedily: connect the t^{th} input vertex v_t to the closest node in the tree S_{t-1}.

Expectation of Ratios

The following lemma is immediate from symmetry.

Lemma 5 *With probability at least* $1 - \frac{1}{L+1}$*, the cost of least expensive tree* T_{i^*} *is no more than* $\rho \mathsf{OPT}(R)$.

The following lemma follows from the fact that for each set of $\log n$ input vertices there is vertex from T_{i^*} nearby. Hence, the connection cost is no more then the greedy connection cost.

Lemma 6 *The cost of connecting the demand vertices is* $O(\mathsf{OPT}(R) \cdot \log\log(nL))$ *with probability at least* $1 - \frac{1}{n^2}$.

Expectation of Ratios

Theorem 7 *Setting $L = O(\log n)$, the expected competitive ratio of the above algorithm is $O(\log \log n)$.*

- Suppose either of the both lemmas fails: this happens with probability at most $\frac{1}{L+1} + \frac{L}{n^2} \leq \frac{2}{\log n}$. In this case, we pay greedy $O(\log n)$ cost, which contributes only constant to EoR.
- If neither of the two lemmas fail, we see that the cost of the algorithm is $O(\mathsf{OPT}(R) \cdot \log \log n)$.

Stochastic Competitive Ratio

In the stochastic case we can consider two different performance measures.

- the *ratio-of-expectations* (RoE):

$$\mathsf{RoE}(\mathsf{ALG}) = \max_{\pi} \max_{k} \frac{\mathbf{E}_{\omega \in \pi^k, r}[\mathsf{ALG}(\omega, r)]}{\mathbf{E}_{\omega \in \pi^k}[\mathsf{OPT}(\omega)]}.$$

- the *expectation-of-ratios* (EoR):

$$\mathsf{RoE}(\mathsf{ALG}) = \max_{\pi} \max_{k} \mathbf{E}_{\omega \in \pi^k, r}\left[\frac{\mathsf{ALG}(\omega, r)}{\mathsf{OPT}(\omega)}\right].$$

Stochastic Competitive Ratio

The ratios are incomparable, but the EoR seems harder.

Consider $L+1$ inputs.

$$\begin{aligned} \text{RoE} &= \frac{L \times 1 + 1 \times 2L^2}{L \times 1 + L} \\ &= \frac{1+2L}{2} \geq L, \\ \text{EoR} &= \frac{1}{L+1}\left(L \times \frac{1}{1} + \frac{L^2}{L}\right) \\ &= \frac{2L}{L+1} \leq 2. \end{aligned}$$

Consider $2L$ inputs.

$$\begin{aligned} \text{RoE} &= \frac{L \times L + L \times 2L}{L \times L + L} \\ &= \frac{2L^2}{L^2 + L} \leq 2, \\ \text{EoR} &= \frac{1}{2L}\left(L \times \frac{L}{L} + L \times \frac{2L}{1}\right) \\ &= \frac{L + 2L^2}{2L} \geq L. \end{aligned}$$

Stochastic Competitive Ratio

Theorem 8 (Known Sequence Length) *There exists polynomial time length-aware algorithm for stochastic online Steiner tree with* $\mathsf{EoR} = O(\log \log n)$.

The length-oblivious case remains unsolved.

Conclusions

The stochastic assumptions soften pessimistic online setting and allow to:

- beat the classical competitive $\Omega(\log n)$ lower bound for the online Steiner tree,

Interesting problem:

- we might be interested algorithms with expected approximation ratio.