Online Stochastic Ad Allocation: Simultaneous and Bicriteria Approximations

Vahab Mirrokni

December 13, 2013

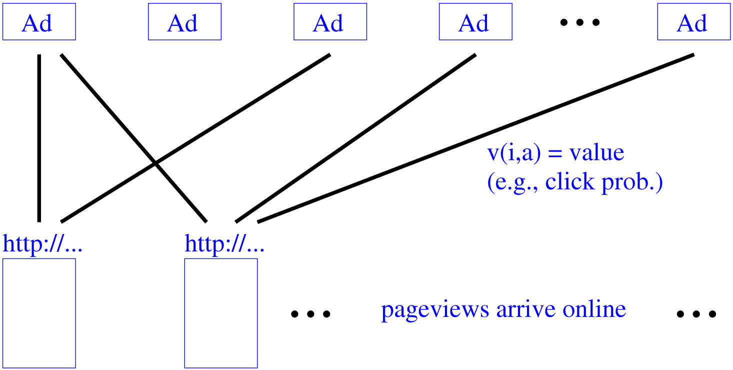
Based on

- SODA'12 paper with Shayan OveisGharan (Stanford), Morteza ZadiMoghaddam (MIT),
- WINE'13 paper with Nitish Korula (Google Research) and Morteza ZadiMoghaddam(MIT).

Outline: Online Allocation

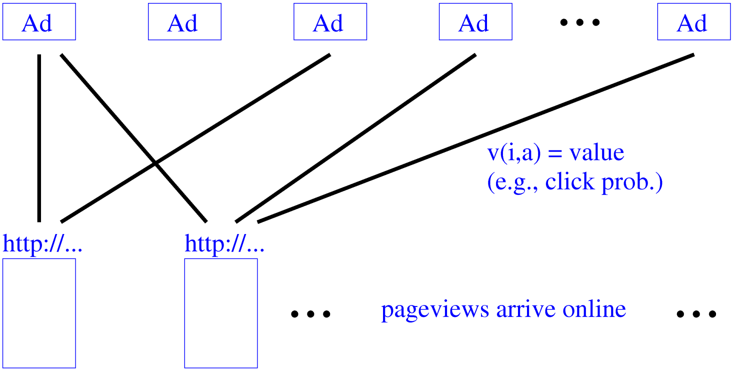
- Online Stochastic Assignment Problems
  - Online Adversarial Arrival Model: Primal-Dual Algorithms
  - Random Order: Dual Algorithms
  - IID with known Distribution: Primal Algorithms
  - Experimental Results
- Simultaneous Stochastic and Adversarial Approximations
  - Motivation and Our Results
  - Hardness Result
  - Factor revealing LP approach
- Multi-objective (Bicriteria) Online Matching

Online Ad Allocation



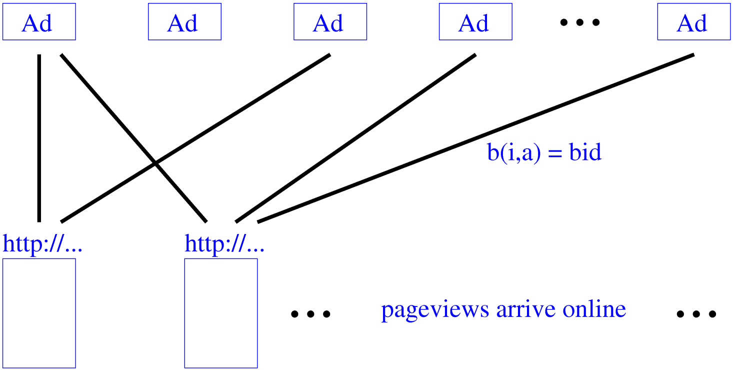
- When a page arrives, assign an eligible ad.
  - value of assigning page i to ad a: v_{ia}

Online Ad Allocation



- When a page arrives, assign an eligible ad.
  - value of assigning page i to ad a: v_{ia}
- Online Weighted Matching [Display Ads (DA)] problem:
  - Maximize value of ads served: $\max \sum_{i,a} v_{ia} x_{ia}$
  - Capacity of ad a: $\sum_{i \in A(a)} x_{ia} \leq C_a$

Online Ad Allocation



- When a page arrives, assign an eligible ad.
  - revenue from assigning page i to ad a: b_{ia}
- Budgeted Allocation ["AdWords" (AW)] problem:
  - Maximize revenue of ads served: $\max \sum_{i,a} b_{ia} x_{ia}$
  - Budget of ad a: $\sum_{i \in A(a)} b_{ia} x_{ia} \leq B_a$

General Form of LP

$$
\begin{aligned}
\max \sum_{i,a} v_{ia} x_{ia} & & & \\
\sum_{a} x_{ia} &\leq 1 & & (\forall\ i) \\
\sum_{i} s_{ia} x_{ia} &\leq C_a & & (\forall\ a) \\
x_{ia} &\geq 0 & & (\forall\ i, a)
\end{aligned}
$$

Online Matching:	Disp. Ads (DA):	AdWords (AW):
$v_{ia} = s_{ia} = 1$	$s_{ia} = 1$	$s_{ia} = v_{ia}$

Arrival Models

- **Adversarial Arrival Model:** Primal-dual Approach

  "ALG is α-competitive?" if for each H, $\frac{\mathrm{ALG}(H)}{\mathrm{OPT}(H)} \geq \alpha$

Arrival Models

- **Adversarial Arrival Model:** Primal-dual Approach

  "ALG is α-competitive?" if for each H, $\frac{\text{ALG}(H)}{\text{OPT}(H)} \geq \alpha$

- **Random Order or IID with unknown dist.:** Dual Approach

  "ALG is α-approximation?" if w.h.p., $\frac{\text{ALG}(H)}{\text{OPT}(H)} \geq \alpha$

  or

  "ALG is α-approximation?" if $\frac{E[\text{ALG}(H)]}{E[\text{OPT}(H)]} \geq \alpha$

Arrival Models

- **Adversarial Arrival Model:** Primal-dual Approach

  "ALG is α-competitive?" if for each H, $\frac{\mathrm{ALG}(H)}{\mathrm{OPT}(H)} \geq \alpha$

- **Random Order or IID with unknown dist.:** Dual Approach

  "ALG is α-approximation?" if w.h.p., $\frac{\mathrm{ALG}(H)}{\mathrm{OPT}(H)} \geq \alpha$

  or

  "ALG is α-approximation?" if $\frac{E[\mathrm{ALG}(H)]}{E[\mathrm{OPT}(H)]} \geq \alpha$

- **iid with known distributions:** Primal Approach

Arrival Models

- **Adversarial Arrival Model:** Primal-dual Approach
  - Unweighted: 1-1/e-competitive algorithm [KarpVV]
  - Weighted (Large Capacity): 1-1/e-competitive algorithm[MehtaSVV]
  - Extensions: BJN,FKMMP

Arrival Models

- **Adversarial Arrival Model:** Primal-dual Approach
  - Unweighted: 1-1/e-competitive algorithm [KarpVV]
  - Weighted (Large Capacity): 1-1/e-competitive algorithm[MehtaSVV]
  - Extensions: BJN,FKMMP
- **Random Order or IID with unknown dist.:** Dual Approach
  - Unweighted: 0.69-competitive algorithm [MahdianY]
  - Weighted (Large Capacity): $1 - \epsilon$-competitive by applying a Dual Approach: learn dual variables and use them later [DevanurH].
  - Extensions: AWY,FHKMS,VJV

Arrival Models

- **Adversarial Arrival Model:** Primal-dual Approach
  - Unweighted: 1-1/e-competitive algorithm [KarpVV]
  - Weighted (Large Capacity): 1-1/e-competitive algorithm[MehtaSVV]
  - Extensions: BJN,FKMMP
- **Random Order or IID with unknown dist.:** Dual Approach
  - Unweighted: 0.69-competitive algorithm [MahdianY]
  - Weighted (Large Capacity): $1 - \epsilon$-competitive by applying a Dual Approach: learn dual variables and use them later [DevanurH].
  - Extensions: AWY,FHKMS,VJV
- **iid with known distributions:** Primal Approach
  - Unweighted: 0.72-competitive [FeldmanMMM & ManshadiOS]
  - Weighted: 0.66-competitive [HaeuplerMZ]

Greedy Algorithm

Assign impression to an advertiser
maximizing Marginal Gain

Greedy Algorithm

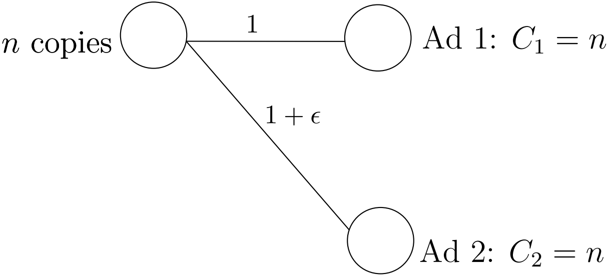
Assign impression to an advertiser
maximizing Marginal Gain

- Competitive Ratio: 1/2. [NWF78]
  - Follows from submodularity of the value function.

Greedy Algorithm

Assign impression to an advertiser
maximizing Marginal Gain

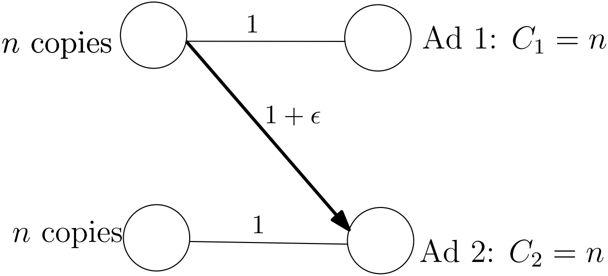
- Competitive Ratio: 1/2. [NWF78]
  - Follows from submodularity of the value function.



Greedy Algorithm

Assign impression to an advertiser
maximizing Marginal Gain

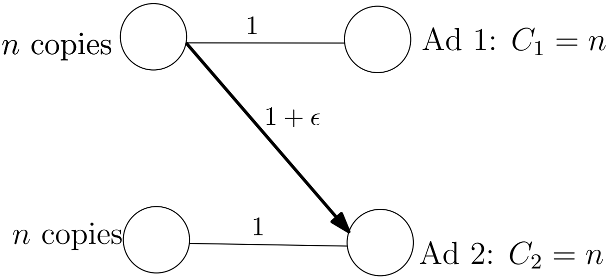
- Competitive Ratio: 1/2. [NWF78]
  - Follows from submodularity of the value function.



Greedy Algorithm

Assign impression to an advertiser
maximizing Marginal Gain

- Competitive Ratio: 1/2. [NWF78]
  - Follows from submodularity of the value function.



Primal-dual Algorithms

- AdWords Problem: [MSVV05] The following is a $1 - \frac{1}{e}$-aprx:

  Assign impression to an advertiser a:

  $$\text{maximizing } b_{ia}(1 - e^{\frac{\text{spent}_a}{B_a} - 1}),$$

Primal-dual Algorithms

- AdWords Problem: [MSVV05] The following is a $1 - \frac{1}{e}$-aprx:

  Assign impression to an advertiser a:

  maximizing $b_{ia}(1 - e^{\frac{\text{spent}_a}{B_a} - 1})$,

- Display Ads problem:

  Assign impression to an advertiser a:

  maximizing (imp. value - β_a),

  - Greedy: $\beta_a =$ min. impression assigned to a.
  - Better (pd-avg): $\beta_a =$ average value of top C_a impressions assigned to a.
  - Optimal (pd-exp): order value of edges assigned to a: $v(1) \geq v(2) \ldots \geq v(C_a)$:

$$\beta_a = \frac{1}{C_a(e-1)} \sum_{j=1}^{C_a} v(j)(1 + \frac{1}{C_a})^{j-1}.$$

  - Thm: pd-exp achieves optimal competitive Ratio: $1 - \frac{1}{e} - \epsilon$ if $C_a > O(\frac{1}{\epsilon})$. [FKMMP09]

Dual-base Algorithm For Random Order

$$\max \sum_{i,a} v_{ia} x_{ia}$$

$$\begin{aligned} \sum_a x_{ia} &\le 1 & (\forall\ i) \\ \sum_i x_{ia} &\le C_a & (\forall\ a) \\ x_{ia} &\ge 0 & (\forall\ i, a) \end{aligned}$$

$$\min \sum_a C_a \beta_a + \sum_i z_i$$

$$\begin{aligned} z_i &\ge v_{ia} - \beta_a & (\forall i, a) \\ \beta_a, z_i &\ge 0 & (\forall i, a) \end{aligned}$$

Dual-base Algorithm For Random Order

$$\max \sum_{i,a} v_{ia} x_{ia}$$

$$\begin{aligned} \sum_a x_{ia} &\leq 1 && (\forall\ i) \\ \sum_i x_{ia} &\leq C_a && (\forall\ a) \\ x_{ia} &\geq 0 && (\forall\ i, a) \end{aligned}$$

$$\min \sum_a C_a \beta_a + \sum_i z_i$$

$$\begin{aligned} z_i &\geq v_{ia} - \beta_a && (\forall i, a) \\ \beta_a, z_i &\geq 0 && (\forall i, a) \end{aligned}$$

Fact: If opt. β_a^* are known, assigning i to argmax$(v_{ia} - \beta_a^*)$ is OPT.

- Proof: Comp. slackness. Given β_a^*, compute x^* as follows: $x_{ia}^* = 1$ if $a = \text{argmax}(v_{ia} - \beta_a^*)$.

Dual-base Algorithm For Random Order

$$\max \sum_{i,a} v_{ia} x_{ia}$$
$$\begin{aligned} \sum_a x_{ia} &\leq 1 && (\forall\ i) \\ \sum_i x_{ia} &\leq C_a && (\forall\ a) \\ x_{ia} &\geq 0 && (\forall\ i, a) \end{aligned}$$

$$\min \sum_a C_a \beta_a + \sum_i z_i$$
$$\begin{aligned} z_i &\geq v_{ia} - \beta_a && (\forall i, a) \\ \beta_a, z_i &\geq 0 && (\forall i, a) \end{aligned}$$

Algorithm:

- Learn a dual variable β_a for each ad a, by solving the dual program on a sample, e.g., sample first ϵ fraction sample of page-views.
- Assign each impression i to ad a that maximizes $v_{ia} - \beta_a$.

Experiments: setup

- Real ad impression data from several large publishers
- 200k - 1.5M impressions in simulation period
- 100 - 2600 advertisers
- Edge weights = predicted click probability
- Algorithms:
  - greedy: maximum marginal value
  - pd-avg, pd-exp: pure online primal-dual from [FKMMP09].
  - dualbase: training-based primal-dual [FHKMS10]
  - hybrid: convex combo of training based, pure online.
  - lp-weight: optimum efficiency

Experimental Evaluation: Summary

Algorithm	Avg Efficiency%	Provable Ratio
opt	100	1
greedy	69	1/2
pd-avg	77	1/2
pd-exp	82	1-1/e
dualbase	87	$1-\epsilon$
hybrid	89	?

In Production

- Algorithms inspired by these techniques are in use at Google DoubleClick display ad serving system, delivering billions of ads per day.

In Production

- Algorithms inspired by these techniques are in use at Google DoubleClick display ad serving system, delivering billions of ads per day.

- Smooth Delivery of Display Ads (Bhalgat, Feldman, M.)
  - Model this with multiple nested capacity constraints.
- Page-based Ad Allocation (Korula, M., Yan)
  - Assign multiple ads per page with configuration constraints.
- Display Ad Allocation with Ad Exchange (Balseiro, Feldman, M., Muthukrishnan)
  - Maximize quality of reservation ads and revenue of Ad Exchange.

In Production

- Algorithms inspired by these techniques are in use at Google DoubleClick display ad serving system, delivering billions of ads per day.

- Smooth Delivery of Display Ads (Bhalgat, Feldman, M.)
  - Model this with multiple nested capacity constraints.
- Page-based Ad Allocation (Korula, M., Yan)
  - Assign multiple ads per page with configuration constraints.
- Display Ad Allocation with Ad Exchange (Balseiro, Feldman, M., Muthukrishnan)
  - Maximize quality of reservation ads and revenue of Ad Exchange.

- Final Algorithm: Adversarial or Stochastic? $\Rightarrow$ **hybrid.**

In practice: Adversarial or Stochastic?

- Stochastic? No. Traffic spikes contradict with the simple random arrival assumption.
- Adversarial? No. Too pessimistic. Some forecasts are useful.
- In practice, hybrid algorithms are used:
  - Use some stochastic predictions,
  - but adapt if predictions fail!

In practice: Adversarial or Stochastic?

- Stochastic? No. Traffic spikes contradict with the simple random arrival assumption.
- Adversarial? No. Too pessimistic. Some forecasts are useful.
- In practice, hybrid algorithms are used:
  - Use some stochastic predictions,
  - but adapt if predictions fail!
- Simultaneous Approximation: Adversarial algorithms that are robust against traffic spikes and perform better when the stochastic information is valid.
- (a,b)-approximation:
  - a-competitive for adversarial.
  - b-competitive for random order.

In practice: Adversarial or Stochastic?

- Stochastic? No. Traffic spikes contradict with the simple random arrival assumption.
- Adversarial? No. Too pessimistic. Some forecasts are useful.
- In practice, hybrid algorithms are used:
  - Use some stochastic predictions,
  - but adapt if predictions fail!
- Simultaneous Approximation: Adversarial algorithms that are robust against traffic spikes and perform better when the stochastic information is valid.
- (a,b)-approximation:
  - a-competitive for adversarial.
  - b-competitive for random order.
- Dealing with Traffic Spikes: Primal and Dual techniques fail in the adversarial models $\Rightarrow (0, b) - \textit{approximation}$.

Simultaneous adversarial & stochastic optimization

- (a,b)-approximation:
  - a-competitive for adversarial.
  - b-competitive for random order.
- Assuming $C_a \gg \max s_{ia}$, are there algorithms that achieve good approximation factors for both adversarial and stochastic models simultaneously?

Simultaneous adversarial & stochastic optimization

- (a,b)-approximation:
  - a-competitive for adversarial.
  - b-competitive for random order.
- Assuming $C_a \gg \max s_{ia}$, are there algorithms that achieve good approximation factors for both adversarial and stochastic models simultaneously?
- Yes for unweighted edges!(M.,OveisGharan, ZadiMoghaddam)
  - PD-EXP algorithm achieves $(1-\epsilon, 1-\frac{1}{e})$-approximation.

Simultaneous adversarial & stochastic optimization

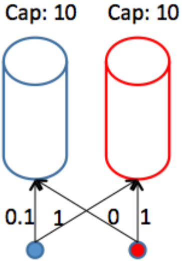
- (a,b)-approximation:
  - a-competitive for adversarial.
  - b-competitive for random order.
- Assuming $C_a \gg \max s_{ia}$, are there algorithms that achieve good approximation factors for both adversarial and stochastic models simultaneously?
- Yes for unweighted edges!(M.,OveisGharan, ZadiMoghaddam)
  - PD-EXP algorithm achieves $(1-\epsilon, 1-\frac{1}{e})$-approximation.
- No for weighted edges! (M.,OveisGharan,ZadiMoghaddam)
  - Achieving $(0.97, 1-\frac{1}{e})$-approximation is impossible.
  - PD-EXP achieves $(0.76, 1-\frac{1}{e})$-approximation.

Hardness Result

- Any $1 - \epsilon$-competitive algorithm for stochastic input can not have a competitive ratio better than $4\sqrt{\epsilon}$ in the adversarial model.
  - Achieving $(4\sqrt{\epsilon}, 1 - \epsilon)$-approximation is impossible.
- In particular, any $1 - 1/e$ competitive algorithm in the adversarial case has competitive ratio at most 97.6% for the random order.

Hardness Result Idea

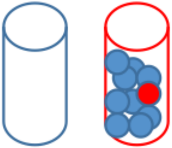
- We have two ads (bins) with the same capacity of 10, generate a random permutation of 100 blue balls and 10 red balls.
- The optimum offline solution fills up both bins completely.

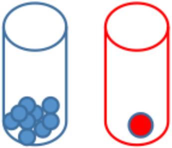


Hardness Result Idea

- Look at the first 10 balls of the random permutation.

- Adversarial Input vs. Stochastic Input





PD-EXP Algorithm for AdWords

- Let
$$f(r(x)) = 1 - e^{r(x)-1},$$
to each advertiser x where $r(x) = \frac{\text{spent(x)}}{B_x}$.
- Allocate a new arrived online node y to the advertiser x maximizing that
$$w(x,y)f(r(x)) = w(x,y)(1 - e^{r(x)-1}).$$
- [MSVV]: PD-EXP is a $1 - 1/e$-competitive algorithm for the worst-case scenario, i.e., adversarial model.

PD-EXP in random order

- PD-EXP has a scoring function $f(r(x))$ for different bins.
- Define a potential function to be the sum of capacities times the anti-derivative of function f for all bins:

$$\phi(t) := \sum_x c(x) \int_{r=0}^{r(t,x)} f(r(x))dr = \sum_x c(x)F(r(t,x)).$$

where $c(x)$ is the capacity of bin x and $r(t,x)$ is the ratio of the spent budget of x at time t.

Changes in Potential Function

- If we allocate item i to bin x with weight $w(i,x)$ in this bin at time t, $\phi(t)$ changes as follows:

$$\Delta(\phi) = c(x)[d(\int_{r=0}^{r}(t,x)f(r(x)))/dr]\Delta(r),$$

  where $\Delta(r)$ is the change in the ratio of bin x.
  So the total change is:

$$\Delta(\phi) = c(x)f(r(t,x))w(i,x)/c(x) = w(i,x)f(r(t,x)).$$

Potential Function Interpretation of the Algorithm

- We conclude that algorithm PD-EXP is trying to greedily maximize the increase in the potential function.
- So in each small fraction of time, the potential function is increased at least as much as the increase in potential based on the optimum solution assignment.

Factor-revealing Mathematical Program

- We have n balls, so t is in $[n]$.
- F is the antiderivative of f, and o_j is how much value is assigned to bin j in the optimum solution.

$$
\begin{aligned}
&\text{MP : minimize } \frac{1}{\text{OPT}} \sum_j r_j(n) c_j \\
&\sum_j c_j F(r_j(t)) = \phi(t) && \forall t \in [n], \\
&\epsilon \sum_j o_j f(r_j((k+1)n\epsilon)) \leq \phi((k+1)n\epsilon) - \phi(kn\epsilon) && \forall k \in [\tfrac{1}{\epsilon} - 1], \\
&o_j \leq c_j && \forall j \in [m], \\
&\sum_j o_j = \text{OPT}, \\
&r_j(t) \leq r_j(t+1) && \forall j, t \in [n-1], \\
&r_j(n) \leq 1 && \forall j \in [m].
\end{aligned}
$$

Converting Factor-revealing MP to Poly-size LP

- We discretize time and spent budget ratios into k parts, so we have k variables for values of potential function at different time intervals.
- We have variable $c[r,t]$ that denotes the sum of capacities of bins with rations in range $[r, r+1/k]$ at time t. Similarly we define $o[r,t]$.

LP minimize $\frac{1}{1-1/e}\left\{\phi(\frac{1}{\epsilon}) - \sum_{\rho=0}^{1/\epsilon-1} c_{\rho,k}(\rho\epsilon/e - e^{\rho\epsilon-1})\right\}$

$\sum_{\rho=0}^{1/\epsilon-1} c_{\rho,k}(\rho\epsilon - e^{\rho\epsilon-1}) \leq \phi(k)$ $\quad \forall k \in [\frac{1}{\epsilon}]$

$\sum_{\rho=0}^{1/\epsilon-1} \epsilon o_{\rho,k+1}(1 - e^{(\rho+1)\epsilon-1}) \geq \phi(k+1) - \phi(k)$ $\quad \forall k \in [\frac{1}{\epsilon} - 1]$

$o_{\rho,k} \leq c_{\rho,k}$ $\quad \forall \rho \in [\frac{1}{\epsilon} - 1], k \in [\frac{1}{\epsilon}]$

$\sum_{i=0}^{1/\epsilon-1} o_{\rho,k} = 1$ $\quad \forall k \in [\frac{1}{\epsilon}]:$

$\sum_{l=i}^{1/\epsilon-1} c_{l,k} \leq \sum_{l=i}^{1/\epsilon-1} c_{l,k+1}$ $\quad \forall i \in [\frac{1}{\epsilon} - 1], k \in [\frac{1}{\epsilon} - 1]$

Converting Factor-revealing MP to poly-size LP

- It gets more work to design a factor revealing LP using this mathematical program that yields 0.76 competitive ratio.
- We discretize both time and ratios into $k = \frac{1}{\epsilon}$ intervals. For $\frac{1}{\epsilon} = 30$, the LP proves 0.73 competitive ratio, for $\frac{1}{\epsilon} = 250$, it proves 0.76.

Summary: Robust Models

- Summary: Simultaneous Approximations
  - Unweighted: $(1 - 1/e, 0.70) \rightarrow (1 - 1/e, 1 - \epsilon)$.
  - Weighted: $(1 - 1/e, 1 - 1/e) \rightarrow (1 - 1/e, 0.76)$.
  - Impossible to get $(1 - 1/e, 0.97)$ for weighted.
- Other robust stochastic models?
  - Approximation factor as a function of accuracy of prediction?

Summary: Robust Models

- Summary: Simultaneous Approximations
  - Unweighted: $(1 - 1/e, 0.70) \to (1 - 1/e, 1 - \epsilon)$.
  - Weighted: $(1 - 1/e, 1 - 1/e) \to (1 - 1/e, 0.76)$.
  - Impossible to get $(1 - 1/e, 0.97)$ for weighted.
- Other robust stochastic models?
  - Approximation factor as a function of accuracy of prediction?
- Adaptively increase/decrease β's using a controller.
  - Tan and Srikant show asymptotic optimality in an iid model.
- Handling bursty random arrival models:
  - Periodic re-optimization achieves constant-factor approximation even in a bursty random arrival model [Ciacon and Farias]

Bicriteria Online Matching

- Multiple-Objective Optimization: Weight, Cardinality, Revenue, Social Welfare.
- e.g. Two Objectives: Assign ads to impressions, maximizing both weight and cardinality of the allocation respecting capacities.

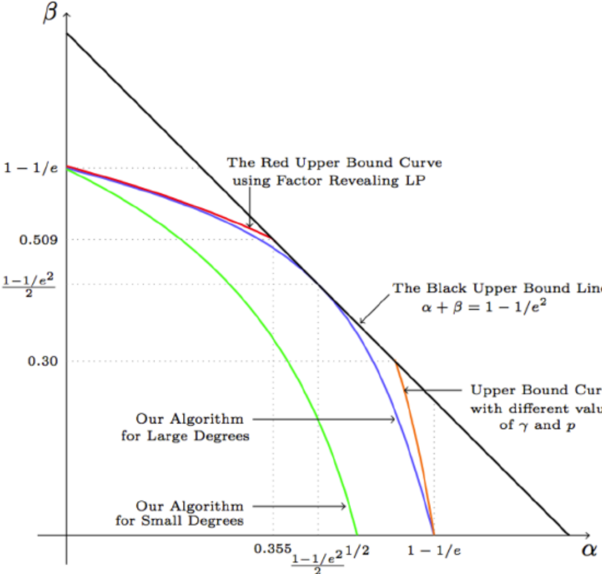
Bicriteria Online Matching

- Multiple-Objective Optimization: Weight, Cardinality, Revenue, Social Welfare.
- e.g. Two Objectives: Assign ads to impressions, maximizing both weight and cardinality of the allocation respecting capacities.
- Bicriteria Online Matching: (α, β)-approximation:
  - α-competitive for weight: Online Weighted Matching (DA)
  - β-competitive for cardinality: Online Matching
- Simple Algorithm: with probability 1/2, optimize for one objective: $((1-1/e)/2, (1-1/e)/2)$-approximation.

Bicriteria Online Matching

- Multiple-Objective Optimization: Weight, Cardinality, Revenue, Social Welfare.
- e.g. Two Objectives: Assign ads to impressions, maximizing both weight and cardinality of the allocation respecting capacities.
- Bicriteria Online Matching: (α, β)-approximation:
  - α-competitive for weight: Online Weighted Matching (DA)
  - β-competitive for cardinality: Online Matching
- Simple Algorithm: with probability 1/2, optimize for one objective: $((1-1/e)/2, (1-1/e)/2)$-approximation.
- Korula, M., ZadiMoghaddam: Bicriteria Online Matching
  - Improve to factor $((1-1/e^2)/2, (1-1/e^2)/2)$ for large capacity, and to $((1-1/e^2)/2, (1-1/e)/2)$ for small cap.
  - The above result is almost tight.

Bicriteria Online Matching: (Korula, M., ZadiMoghaddam)



Research Directions

- Open Problems:
  - Compete with online DP in the iid model: PTAS?
  - Small budgets and small degrees: Improve 1/2-approximation.
  - Bicriteria approximation for two weight functions?
- Other robust stochastic models?
  - Approximation factor as a function of accuracy of prediction?
- Online budget planning for repeated 2nd-price auctions?
  - Incentive-compatiblity: Clinching Auctions (Goel, M., Paes Leme, STOC12, SODA13)
  - (mean-field) equilibria? extensive-form or Nash equilibria?

Algorithms Research at Google NYC

- Advertiser (Bid) Optimization
- Online Stochastic Ad Allocation
- Mechanism Design for Ad Exchanges
- Large-scale Graph Mining
  - Develop large-scale algorithms to mine graphs with trillions of edges.
  - Distributed Frameworks: Map-Reduce, Pregel, or Asynchronous Message Passing.

Thank You