Online Ad Serving: Theory and Practice

Vahab Mirrokni
(Three papers in collaboration with Googlers)

Google Research, New York

October 20, 2010

Contract-based Online Advertising

- Pageviews (impressions) instead of queries.
- Display/Banner Ads, Video Ads, Mobile Ads.

Contract-based Online Advertising

- Pageviews (impressions) instead of queries.
- Display/Banner Ads, Video Ads, Mobile Ads.
- Cost-Per-Impression (CPM).
- Not Auction-based: offline negotiations + Online allocations.

Contract-based Online Advertising

- Pageviews (impressions) instead of queries.
- Display/Banner Ads, Video Ads, Mobile Ads.
- Cost-Per-Impression (CPM).
- Not Auction-based: offline negotiations + Online allocations.

Display/Banner Ads:

- Q1, 2010: One Trillion Display Ads in US. $2.7 billion.
- Top Publishers: Facebook, Yahoo and Microsoft sites.
- Top Advertiser: AT&T, Verizon, Scottrade.

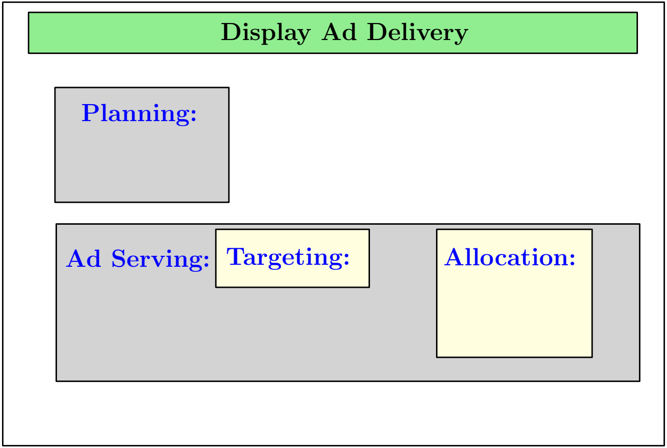
Contract-based Online Advertising

- Pageviews (impressions) instead of queries.
- Display/Banner Ads, Video Ads, Mobile Ads.
- Cost-Per-Impression (CPM).
- Not Auction-based: offline negotiations + Online allocations.

Display/Banner Ads:

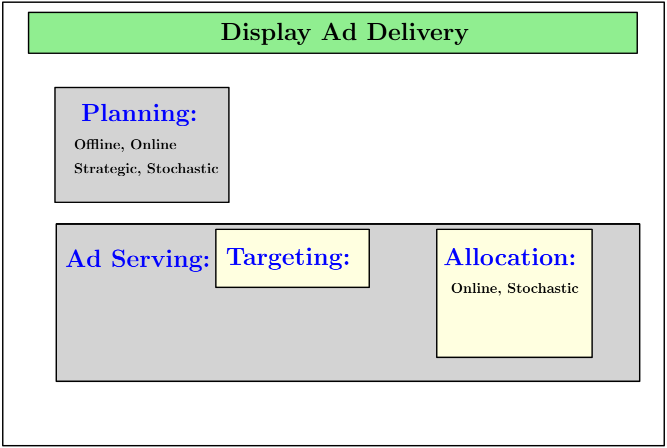
- Q1, 2010: One Trillion Display Ads in US. $2.7 billion.
- Top Publishers: Facebook, Yahoo and Microsoft sites.
- Top Advertiser: AT&T, Verizon, Scottrade.
- Ad Serving Systems e.g. Facebook, Google Doubleclick, AdMob.

Display Ad Delivery: Overview



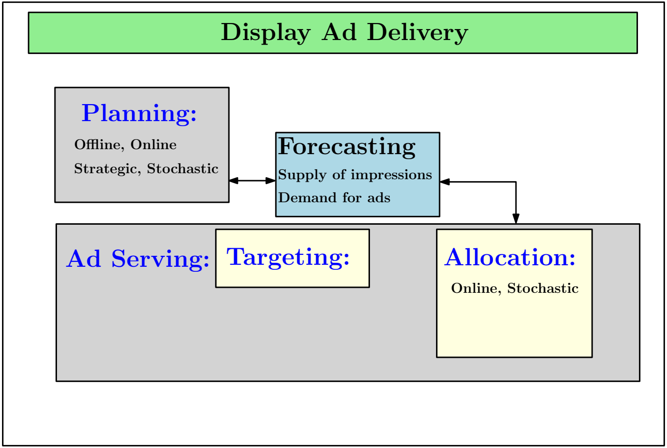
1. Planning: Contracts/Commitments with Advertisers.
2. Ad Serving:
   - Targeting: Predicting value of impressions.
   - Ad Allocation: Assigning Impressions to Ads Online.

Display Ad Delivery: Overview



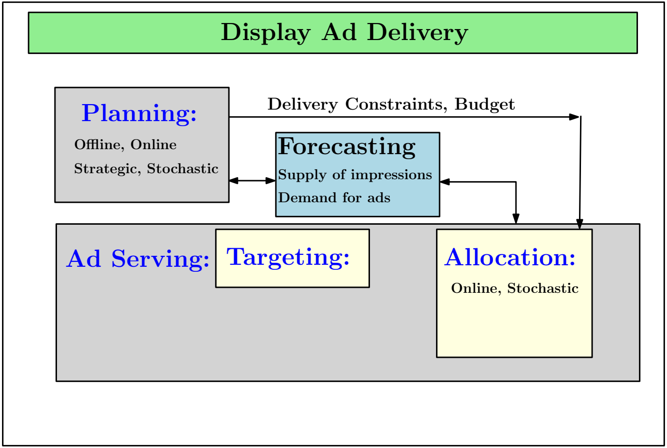
1. Planning: Contracts/Commitments with Advertisers.
2. Ad Serving:
    - Targeting: Predicting value of impressions.
    - Ad Allocation: Assigning Impressions to Ads Online.

Display Ad Delivery: Overview



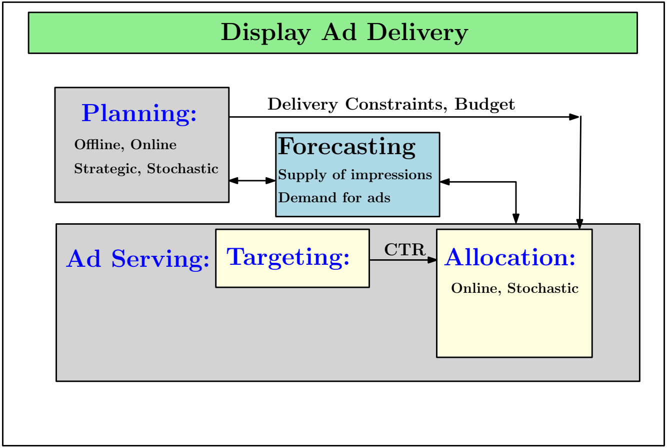
1. Planning: Contracts/Commitments with Advertisers.
2. Ad Serving:
   - Targeting: Predicting value of impressions.
   - Ad Allocation: Assigning Impressions to Ads Online.

Display Ad Delivery: Overview



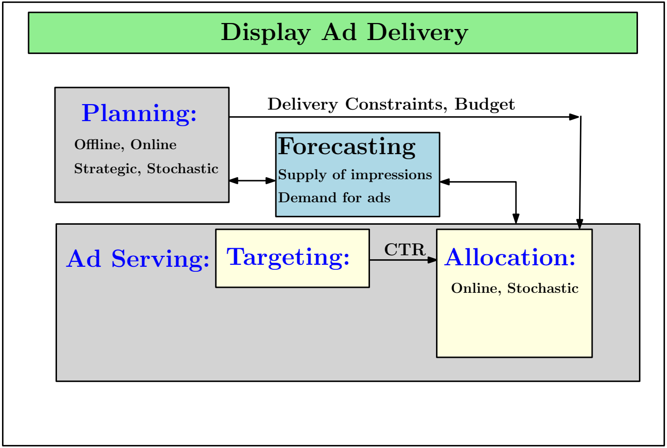
1. Planning: Contracts/Commitments with Advertisers.
2. Ad Serving:
   - Targeting: Predicting value of impressions.
   - Ad Allocation: Assigning Impressions to Ads Online.

Display Ad Delivery: Overview



1. Planning: Contracts/Commitments with Advertisers.
2. Ad Serving:
   - Targeting: Predicting value of impressions.
   - Ad Allocation: Assigning Impressions to Ads Online.

Display Ad Delivery: Overview



- **Objective Functions:**
  - Efficiency: Users and Advertisers. Revenue of the Publisher.
  - Smoothness, Fairness, Delivery Penalty.

Targeting

Estimating Value of an impression.

Targeting

Estimating Value of an impression.

- Behavioral Targeting
  - Interest-based Advertising.
  - Yan, Liu, Wang, Zhang, Jiang, Chen, 2009, How much can Behavioral Targeting Help Online Advertising?

Targeting

Estimating Value of an impression.

- Behavioral Targeting
  - Interest-based Advertising.
  - Yan, Liu, Wang, Zhang, Jiang, Chen, 2009, How much can Behavioral Targeting Help Online Advertising?
- Contextual Targeting
  - Information Retrieval (IR).
  - Broder, Fontoura, Josifovski, Riedel, A semantic approach to contextual advertising

Targeting

Estimating Value of an impression.

- Behavioral Targeting
  - Interest-based Advertising.
  - Yan, Liu, Wang, Zhang, Jiang, Chen, 2009, How much can Behavioral Targeting Help Online Advertising?
- Contextual Targeting
  - Information Retrieval (IR).
  - Broder, Fontoura, Josifovski, Riedel, A semantic approach to contextual advertising
- Creative Optimization
  - Experimentation

Predicting value of Impressions for Display Ads

- Estimating Click-Through-Rate (CTR).
  - Budgeted Multi-armed Bandit
- Probability of Conversion.

Predicting value of Impressions for Display Ads

- Estimating Click-Through-Rate (CTR).
  - Budgeted Multi-armed Bandit
- Probability of Conversion.
- Long-term vs. Short-term value of display ads?
  - Archak, Mirrokni, Muthukrishnan, 2010 Graph-based Models.
    - Computing Adfactors based on AdGraphs
    - Markov Models for Advertiser-specific User Behavior

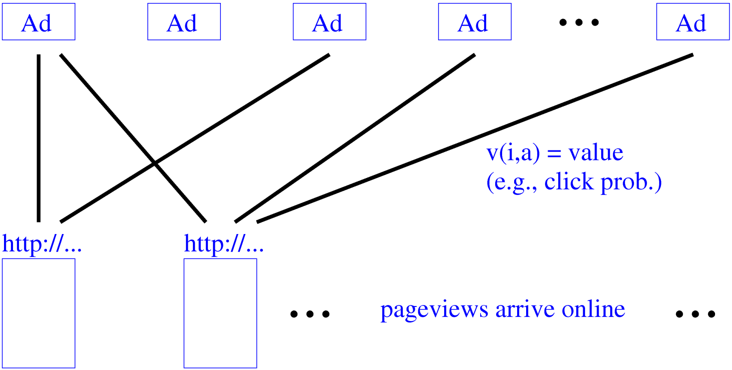
Contract-based Ad Delivery: Outline

- Basic Information
- Ad Planning: Reservation
- Ad Serving.
  - Targeting.
  - Online Ad Allocation

Outline: Online Allocation

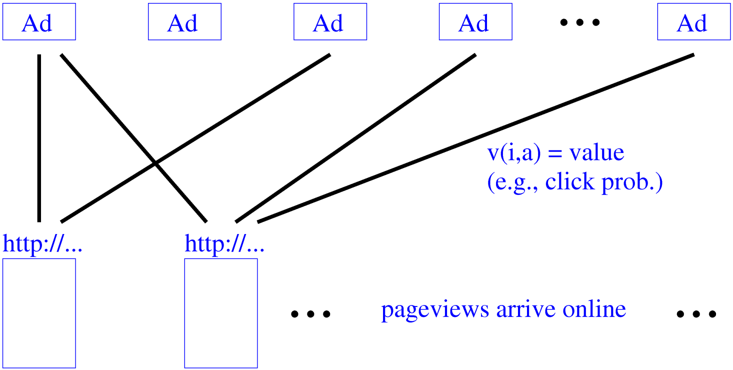
- Online Stochastic Assignment Problems
  - Online (Stochastic) Matching
  - Online Generalized Assignment (with free disposal)
  - Online Stochastic Packing
  - Experimental Results
- Online Learning and Allocation

Online Ad Allocation



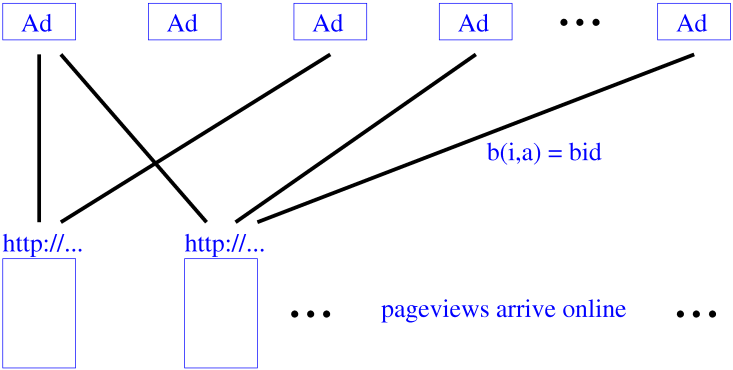
- When page arrives, assign an eligible ad.
  - value of assigning page i to ad a: v_{ia}

Online Ad Allocation



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

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

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

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

General Form of LP

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

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

	Online Matching: $v_{ia} = s_{ia} = 1$	Disp. Ads (DA): $s_{ia} = 1$	AdWords (AW): $s_{ia} = v_{ia}$
Worst-Case	Greedy: $\frac{1}{2}$, [KVV]: $1 - \frac{1}{e}$-aprx	Inapproximable	[MSVV,BJN]: $1 - \frac{1}{e}$-aprx

Ad Allocation: Problems and Models

	Online Matching: $v_{ia} = s_{ia} = 1$	Disp. Ads (DA): $s_{ia} = 1$	AdWords (AW): $s_{ia} = v_{ia}$
Worst Case	Greedy: $\frac{1}{2}$, [KVV]: $1 - \frac{1}{e}$-aprx	Inapproximable ?	[MSVV,BJN]: $1 - \frac{1}{e}$-aprx

Ad Allocation: Problems and Models

	Online Matching: $v_{ia} = s_{ia} = 1$	Disp. Ads (DA): $s_{ia} = 1$	AdWords (AW): $s_{ia} = v_{ia}$
Worst Case	Greedy: $\frac{1}{2}$, [KVV]: $1-\frac{1}{e}$-aprx	Inapproximable ?	[MSVV,BJN]: $1-\frac{1}{e}$-aprx
Stochastic (i.i.d.)		?	[DH09]: $1-\epsilon$-aprx, if $\text{OPT} \gg \max v_{ia}$

Stochastic i.i.d model:

- i.i.d model with known distribution
- random order model (i.i.d model with unknown distribution)

Ad Allocation: Problems and Models

	Online Matching: $v_{ia} = s_{ia} = 1$	Disp. Ads (DA): $s_{ia} = 1$	AdWords (AW): $s_{ia} = v_{ia}$
Worst Case	Greedy: $\frac{1}{2}$, [KVV]: $1 - \frac{1}{e}$-aprx	Inapproximable ?	[MSVV,BJN]: $1 - \frac{1}{e}$-aprx
Stochastic (i.i.d.)	[FMMM09]: 0.67-aprx i.i.d with known distribution	?	[DH09]: $1 - \epsilon$-aprx, if $\text{OPT} \gg \max v_{ia}$

Stochastic i.i.d model:

- i.i.d model with known distribution
- random order model (i.i.d model with unknown distribution)

Online Stochastic Matching: Motivation

- Pageview supply from the past should tell us something about the future [Parkes, Sandholm, SSA 2005][Abrams, Mendelevitch, Tomlin, EC 07] [Boutilier, Parkes, Sandholm, Walsh AAAI 08].

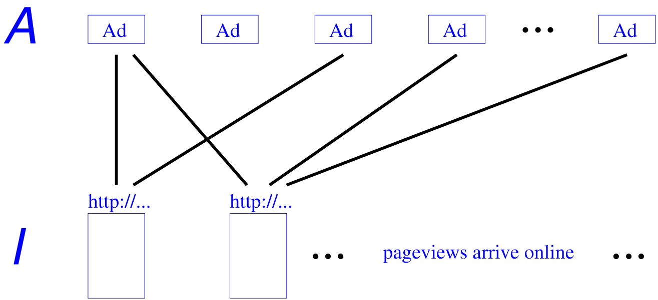
Online Stochastic Matching: Motivation

- Pageview supply from the past should tell us something about the future [Parkes, Sandholm, SSA 2005][Abrams, Mendelevitch, Tomlin, EC 07] [Boutilier, Parkes, Sandholm, Walsh AAAI 08].
- Primal Algorithm:
  - Construct an expected instance,
  - Compute an optimal solution to this expected instance,
  - Use this solution to guide the online allocation.

Online Stochastic Matching: Motivation

- Pageview supply from the past should tell us something about the future [Parkes, Sandholm, SSA 2005][Abrams, Mendelevitch, Tomlin, EC 07] [Boutilier, Parkes, Sandholm, Walsh AAAI 08].
- Primal Algorithm:
  - Construct an expected instance,
  - Compute an optimal solution to this expected instance,
  - Use this solution to guide the online allocation.
- Can we extend the theory of online algorithms to this architecture?

Online Stochastic Matching: iid (known dist.)



Given (offline):

- Bipartite graph $G = (A, I, E)$,
- Distribution D over I.

Online:

- n indep. draws from D.
- Must assign nodes upon arrival.

Primal Algorithm: “Two-suggested-matchings”

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

Simple Primal Algorithm:

- Find one matching in expected graph G offline, and try to apply it online.
- Tight $1 - \frac{1}{e}$-approximation.

Primal Algorithm: “Two-suggested-matchings”

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

Simple Primal Algorithm:

- Find one matching in expected graph G offline, and try to apply it online.
- Tight $1 - \frac{1}{e}$-approximation.

Better Algorithm: Two-Suggested-Matchings

- Offline: Find two disjoint matchings, blue(B) and red(R), on the expected graph G.
- Online: try the blue matching first, then if that doesn’t work, try the red one.

Primal Algorithm: "Two-suggested-matchings"

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

Simple Primal Algorithm:

- Find one matching in expected graph G offline, and try to apply it online.
- Tight $1 - \frac{1}{e}$-approximation.

Better Algorithm: Two-Suggested-Matchings

- Offline: Find two disjoint matchings, blue(B) and red(R), on the expected graph G.
- Online: try the blue matching first, then if that doesn't work, try the red one.
- Thm: Tight $\frac{1-2/e^2}{4/3-2/3e} \geq 0.67$

(Feldman, M., M., Muthukrishnan, 2009).

Background: Balls in bins

- Suppose n balls thrown into n bins, i.i.d. uniform.

Background: Balls in bins

- Suppose n balls thrown into n bins, i.i.d. uniform.
- # non-empty bins concentrates:

Background: Balls in bins

- Suppose n balls thrown into n bins, i.i.d. uniform.
- # non-empty bins concentrates:
  - $B =$ particular subset of bins.

Background: Balls in bins

- Suppose n balls thrown into n bins, i.i.d. uniform.
- # non-empty bins concentrates:
  - $B =$ particular subset of bins.
  - $s = \#$ bins in B with ≥ 1 ball.

Background: Balls in bins

- Suppose n balls thrown into n bins, i.i.d. uniform.
- # non-empty bins concentrates:
  - $B =$ particular subset of bins.
  - $s = \#$ bins in B with ≥ 1 ball.
  - Then w.h.p., $s \approx |B|(1 - \frac{1}{e})$.

Analysis: Two-suggested-matching Algorithm

- Proof Ideas: Balls-into-Bins concentration inequalities, structural properties of min-cuts, etc.

Analysis: Two-suggested-matching Algorithm

- Proof Ideas: Balls-into-Bins concentration inequalities, structural properties of min-cuts, etc.
- Bounding ALG: Classify $a \in A$ based on its neighbors in the blue and red matchings: A_{BR}, A_{BB}, A_B, A_R

$$ALG \geq \left(1 - \frac{1}{e^2}\right)|A_{BB}| + \left(1 - \frac{2}{e^2}\right)|A_{BR}| + \left(1 - \frac{3}{2e}\right)(|A_B| + |A_R|)$$

Analysis: Two-suggested-matching Algorithm

- Proof Ideas: Balls-into-Bins concentration inequalities, structural properties of min-cuts, etc.
- Bounding ALG: Classify $a \in A$ based on its neighbors in the blue and red matchings: A_{BR}, A_{BB}, A_B, A_R

$$ALG \geq \left(1 - \frac{1}{e^2}\right)|A_{BB}| + \left(1 - \frac{2}{e^2}\right)|A_{BR}| + \left(1 - \frac{3}{2e}\right)(|A_B| + |A_R|)$$

- Bounding OPT: Find min-cut in augmented expected graph G, and use it min-cut in G as a "guide" for cut in each scenario.

First Attempt: “Suggested matching”

1. Find a maximum matching in G.
2. Use that matching as nodes arrive online.

First Attempt: "Suggested matching"

1. Find a maximum matching in G.
2. Use that matching as nodes arrive online.

- Does no better than $1 - 1/e$.

First Attempt: “Suggested matching”

1. Find a maximum matching in G.
2. Use that matching as nodes arrive online.

- Does no better than $1 - 1/e$.
- Proof:
  - Suppose $G =$ complete graph.

First Attempt: "Suggested matching"

1. Find a maximum matching in G.
2. Use that matching as nodes arrive online.

- Does no better than $1 - 1/e$.
- Proof:
  - Suppose $G =$ complete graph.
  - Then $\text{OPT(H)} = n$.

First Attempt: "Suggested matching"

1. Find a maximum matching in G.
2. Use that matching as nodes arrive online.

- Does no better than $1 - 1/e$.
- Proof:
  - Suppose $G =$ complete graph.
  - Then OPT(H) $= n$.
  - But w.h.p. only $1 - 1/e$ fraction of I will ever arrive.
    $\implies$ ALG $\approx (1 - 1/e)n$.

First Attempt: “Suggested matching”

1. Find a maximum matching in G.
2. Use that matching as nodes arrive online.

- Does no better than $1 - 1/e$.
- Proof:
  - Suppose $G =$ complete graph.
  - Then $\text{OPT(H)} = n$.
  - But w.h.p. only $1 - 1/e$ fraction of I will ever arrive.
    $\implies \text{ALG} \approx (1 - 1/e)n$.
- In fact, this algorithm does achieve $1 - 1/e$ (in paper).

New ALG: “Two suggested matchings”

1. Offline: Find two disjoint matchings
2. Online: try the first one, then if that doesn’t work, try the second one.

New ALG: "Two suggested matchings"

Warmup: complete graph

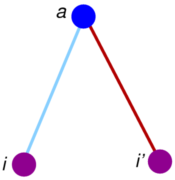
- Two disjoint perfect matchings: blue (1-ary), red (2-ary).

New ALG: “Two suggested matchings”

Warmup: complete graph

- Two disjoint perfect matchings: blue (1-ary), red (2-ary).
- Union of matchings = cycles with alt. blue and red edges

New ALG: "Two suggested matchings"

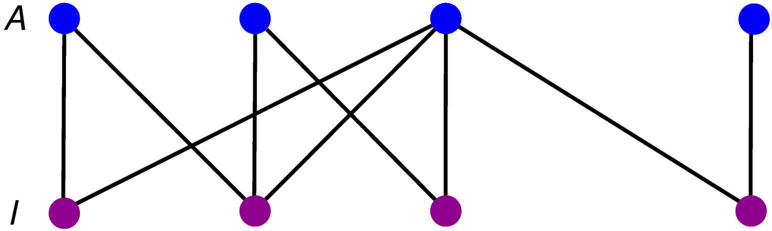


For particular node $a \in A$:

$$\begin{aligned}
\Pr[a \text{ is chosen }] &\geq \Pr[i \text{ arrives once, or } i' \text{ arrives twice}] \\
&= 1 - \Pr[i \text{ never arrives \& } i' \text{ arrives } \leq \text{ once}] \\
&= 1 - \left((1-2/n)^n + n(1/n)(1-2/n)^{n-1}\right) \\
&\approx 1 - 2/e^2
\end{aligned}$$

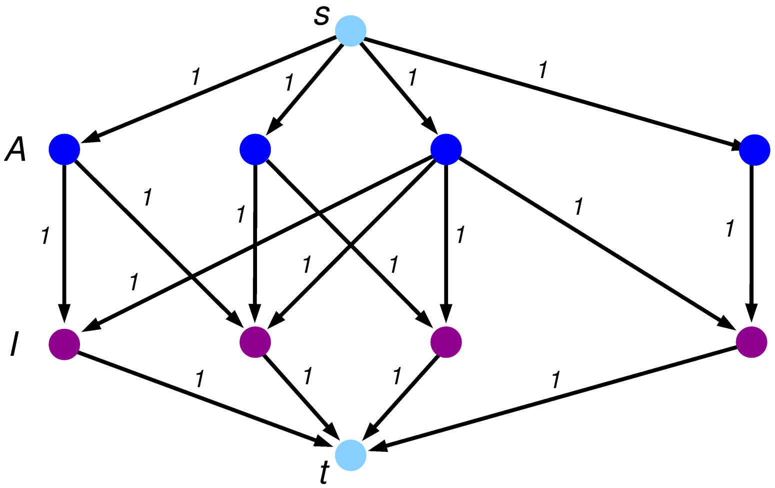
Thus, $E[\#\text{ nodes in } A \text{ chosen}] \approx (1-2/e^2)n \approx .729n$
(This also concentrates...)

Algorithm (Offline)



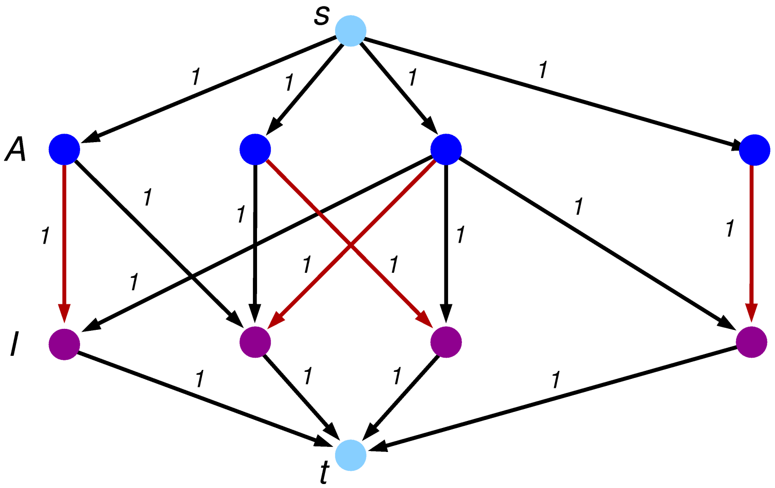
- How to find a matching with flow.

Algorithm (Offline)



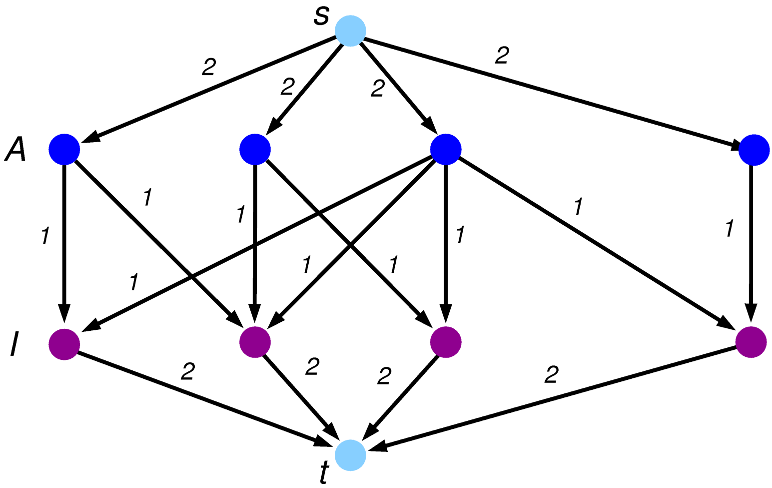
- How to find a matching with flow.

Algorithm (Offline)



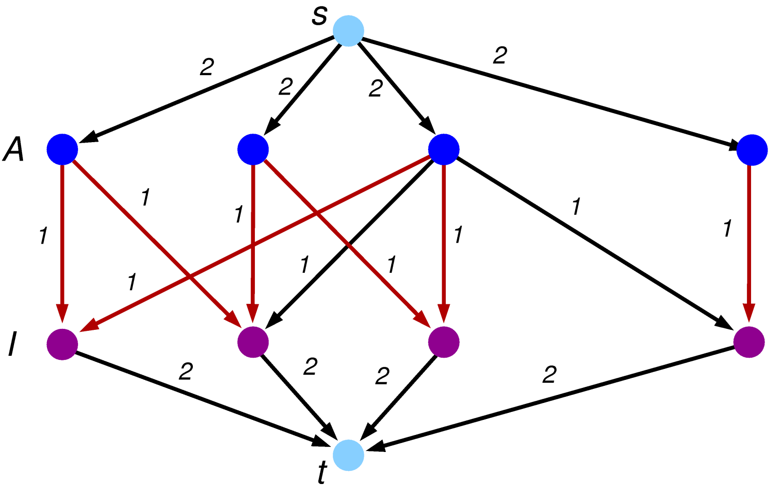
- How to find a matching with flow.

Algorithm (Offline)



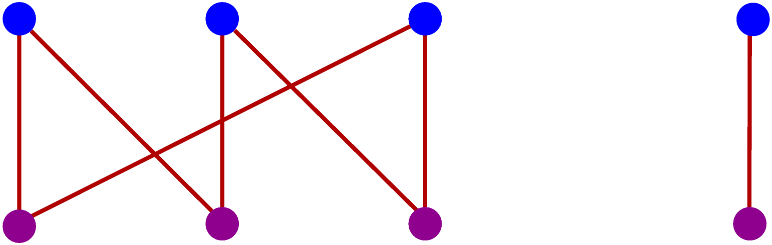
- Solve an "augmented flow" problem instead.

Algorithm (Offline)



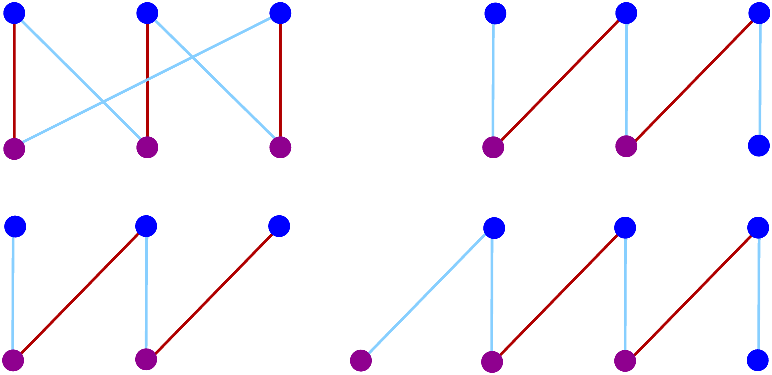
- Solve an "augmented flow" problem instead.

Algorithm (Offline)



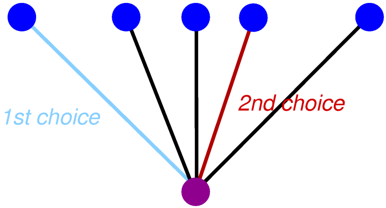
- Examine edges in flow.

Algorithm (Offline)



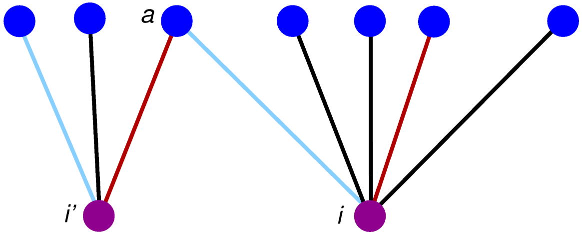
- Color the edges as shown

Algorithm (Online)



- When node $i \in I$ arrives:
  - Try the blue edge first, then the red edge.

Algorithm (Online)



- Consider a node $a \in A$:
  - $\Pr[a \text{ is chosen }] \geq \Pr[i \text{ arrives once, or } i' \text{ arrives twice}]$

Performance of the Algorithm

- Classify $a \in A$ based on its neighbors in the flow.

$$|\text{flow}| = 2|A_{BR}| + 2|A_{BB}| + |A_B| + |A_R|$$

Performance of the Algorithm

- Classify $a \in A$ based on its neighbors in the flow.

$$|\text{flow}| = 2|A_{BR}| + 2|A_{BB}| + |A_B| + |A_R|$$

- Using Balls-in-bins concentration results (Azuma's inequality):

Performance of the Algorithm

- Classify $a \in A$ based on its neighbors in the flow.

$$|\text{flow}| = 2|A_{BR}| + 2|A_{BB}| + |A_B| + |A_R|$$

- Using Balls-in-bins concentration results (Azuma's inequality):
  - $a \in A_B$. We get at least $|A_B|(1 - 1/e)$.

Performance of the Algorithm

- Classify $a \in A$ based on its neighbors in the flow.

$$|\text{flow}| = 2|A_{BR}| + 2|A_{BB}| + |A_B| + |A_R|$$

- Using Balls-in-bins concentration results (Azuma's inequality):
  - $a \in A_B$. We get at least $|A_B|(1 - 1/e)$.
  - $a \in A_{BR}$. We get at least $|A_{BR}|(1 - 2/e^2)$.

Performance of the Algorithm

- Classify $a \in A$ based on its neighbors in the flow.

$$|\text{flow}| = 2|A_{BR}| + 2|A_{BB}| + |A_B| + |A_R|$$

- Using Balls-in-bins concentration results (Azuma's inequality):
  - $a \in A_B$. We get at least $|A_B|(1 - 1/e)$.
  - $a \in A_{BR}$. We get at least $|A_{BR}|(1 - 2/e^2)$.
  - $a \in A_{BB}$. We get at least $|A_{BB}|(1 - 1/e^2)$.

Performance of the Algorithm

- Classify $a \in A$ based on its neighbors in the flow.

$$|\text{flow}| = 2|A_{BR}| + 2|A_{BB}| + |A_B| + |A_R|$$

- Using Balls-in-bins concentration results (Azuma's inequality):
  - $a \in A_B$. We get at least $|A_B|(1 - 1/e)$.
  - $a \in A_{BR}$. We get at least $|A_{BR}|(1 - 2/e^2)$.
  - $a \in A_{BB}$. We get at least $|A_{BB}|(1 - 1/e^2)$.
  - $a \in A_R$. We get at least $|A_R|(1 - 2/e)$.

Performance of the Algorithm

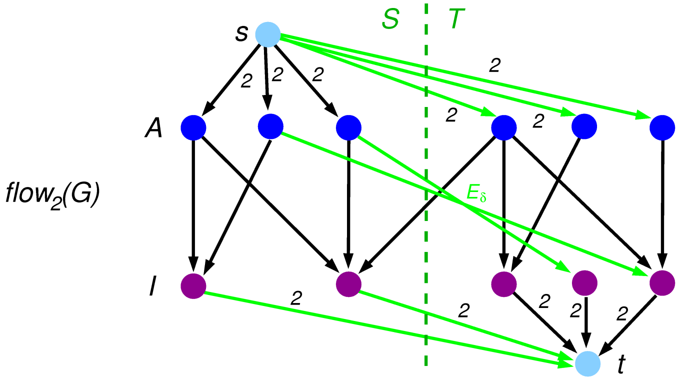
- Classify $a \in A$ based on its neighbors in the flow.

$$|\text{flow}| = 2|A_{BR}| + 2|A_{BB}| + |A_B| + |A_R|$$

- Using Balls-in-bins concentration results (Azuma's inequality):
  - $a \in A_B$. We get at least $|A_B|(1 - 1/e)$.
  - $a \in A_{BR}$. We get at least $|A_{BR}|(1 - 2/e^2)$.
  - $a \in A_{BB}$. We get at least $|A_{BB}|(1 - 1/e^2)$.
  - $a \in A_R$. We get at least $|A_R|(1 - 2/e)$.
- Bound on ALG in terms of flow (using $|B| \geq |R|$):

$$ALG \geq \left(1 - \frac{1}{e^2}\right)|A_{BB}| + \left(1 - \frac{2}{e^2}\right)|A_{BR}| + \left(1 - \frac{3}{2e}\right)(|A_B| + |A_R|)$$

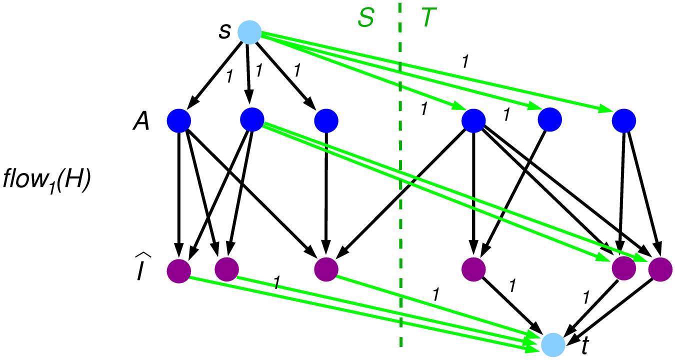
Bounding OPT



- Find min-cut in augmented flow graph (from G).
- E_δ is a matching.
- By max-flow min-cut,

$$|\text{flow}| = 2(|A_T| + |I_S|) + |E_\delta|.$$

Bounding OPT



- OPT $\leq$ cut(H). (Remember $H = (A, \hat{I}, \hat{E})$.)
- Use min-cut in G as "guide" for cut in H.
- W.h.p., $|I_S| \approx |\hat{I}_S|$. E_δ?
- For any node $a \in S$ with an edge in the cut in $\hat{E}(H)$, move it to $T \Rightarrow$ # nonempty nodes in $E_\delta \Rightarrow (1 - \frac{1}{e})E_\delta$.

Putting things together

- Eventually (after moving a few nodes around) you get
  - $OPT \lesssim |I_S| + |A_T| + (1 - 1/e)|E_\delta|$.

Putting things together

- Eventually (after moving a few nodes around) you get
  - $OPT \lesssim |I_S| + |A_T| + (1 - 1/e)|E_\delta|$.
- A lemma relating the decomposition to the cut gives
  - $|E_\delta| \leq \frac{2}{3}|A_{BR}| + \frac{4}{3}|A_{BB}| + |A_B| + \frac{1}{3}|A_R|$,

Putting things together

- Eventually (after moving a few nodes around) you get
  - $OPT \lesssim |I_S| + |A_T| + (1 - 1/e)|E_\delta|$.
- A lemma relating the decomposition to the cut gives
  - $|E_\delta| \leq \frac{2}{3}|A_{BR}| + \frac{4}{3}|A_{BB}| + |A_B| + \frac{1}{3}|A_R|$,
- which, when combined with
  - $|\text{flow}| = 2(|A_T| + |I_S|) + |E_\delta|$
  - $|\text{flow}| = 2|A_{BR}| + 2|A_{BB}| + |A_B| + |A_R|$,
  - $\text{ALG} \geq (1 - \frac{1}{e^2})|A_{BB}| + (1 - \frac{2}{e^2})|A_{BR}| + (1 - \frac{3}{2e})(|A_B| + |A_R|)$,
- gives
  - $\frac{ALG}{OPT} \geq min\{\frac{1-1/e^2}{5/3-4/3e}, \frac{1-2/e^2}{4/3-2/3e}, \frac{1-3/2e}{1-1/e}\}$
  - $\frac{ALG}{OPT} \geq .67$

Putting things together

- Eventually (after moving a few nodes around) you get
  - $OPT \lesssim |I_S| + |A_T| + (1 - 1/e)|E_\delta|$.
- A lemma relating the decomposition to the cut gives
  - $|E_\delta| \leq \frac{2}{3}|A_{BR}| + \frac{4}{3}|A_{BB}| + |A_B| + \frac{1}{3}|A_R|$,
- which, when combined with
  - $|\text{flow}| = 2(|A_T| + |I_S|) + |E_\delta|$
  - $|\text{flow}| = 2|A_{BR}| + 2|A_{BB}| + |A_B| + |A_R|$,
  - $\text{ALG} \geq (1 - \frac{1}{e^2})|A_{BB}| + (1 - \frac{2}{e^2})|A_{BR}| + (1 - \frac{3}{2e})(|A_B| + |A_R|)$,
- gives
  - $\frac{ALG}{OPT} \geq min\{\frac{1-1/e^2}{5/3-4/3e}, \frac{1-2/e^2}{4/3-2/3e}, \frac{1-3/2e}{1-1/e}\}$
  - $\frac{ALG}{OPT} \geq .67$
- The analysis is tight.

Ad Allocation: Problems and Models

	Online Matching: $v_{ia} = s_{ia} = 1$	Disp. Ads (DA): $s_{ia} = 1$	AdWords (AW): $s_{ia} = v_{ia}$
Worst Case	Greedy: $\frac{1}{2}$, [KVV]: $1 - \frac{1}{e}$-aprx	Inapproximable ?	[MSVV,BJN]: $1 - \frac{1}{e}$-aprx
Stochastic (i.i.d.)	[FMMM09]: 0.67-aprx i.i.d with known distribution	?	[DH09]: $1 - \epsilon$-aprx, if $\mathrm{OPT} \gg \max v_{ia}$

Ad Allocation: Problems and Models

	Online Matching: $v_{ia} = s_{ia} = 1$	Disp. Ads (DA): $s_{ia} = 1$	AdWords (AW): $s_{ia} = v_{ia}$
Worst Case	Greedy: $\frac{1}{2}$, [KVV]: $1 - \frac{1}{e}$-aprx	Inapproximable ?	[MSVV,BJN]: $1 - \frac{1}{e}$-aprx
Stochastic (i.i.d.)	[FMMM09]: 0.67-aprx i.i.d with known distribution	[FHKMS10,AWY]: $1 - \epsilon$-aprx, if $\mathrm{OPT} \gg \max v_{ia}$ and $n \gg m$	[DH09]: $1 - \epsilon$-aprx, if $\mathrm{OPT} \gg \max v_{ia}$

random order = i.i.d. model with unknown distribution

Stochastic DA: Dual Algorithm

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

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

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

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

Algorithm:

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

Stochastic DA: Dual Algorithm

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

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

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

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

Algorithm:

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

Stochastic DA: Dual Algorithm

Feldman, Henzinger, Korula, M., Stein 2010

Thm[FHKMS10,AWY]: W.h.p, this algorithm is a $(1-O(\epsilon))$-aprx, as long as each item has low value ($v_{ia} \leq \frac{\epsilon \mathrm{OPT}}{m \log n}$), and large capacity ($C_a \leq \frac{m \log n}{\epsilon^3}$)

Stochastic DA: Dual Algorithm

Feldman, Henzinger, Korula, M., Stein 2010

Thm[FHKMS10,AWY]: W.h.p, this algorithm is a $(1 - O(\epsilon))$-aprx, as long as each item has low value ($v_{ia} \leq \frac{\epsilon \mathrm{OPT}}{m \log n}$), and large capacity ($C_a \leq \frac{m \log n}{\epsilon^3}$)

Fact: If optimum β_a^* are known, this alg. finds OPT

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

Stochastic DA: Dual Algorithm

Feldman, Henzinger, Korula, M., Stein 2010

Thm[FHKMS10,AWY]: W.h.p, this algorithm is a $(1 - O(\epsilon))$-aprx, as long as each item has low value ($v_{ia} \leq \frac{\epsilon \mathrm{OPT}}{m \log n}$), and large capacity ($C_a \leq \frac{m \log n}{\epsilon^3}$)

Fact: If optimum β_a^* are known, this alg. finds OPT

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

Lemma: In the random order model, W.h.p., the sample β'_a are close to β_a^*.

- Extending DH09.

General Stochastic Packing LPs

- m fixed *resources* with capacity C_a
- *Items* i arrive online with *options* O_i, *values* v_{io}, *rsrc. use* s_{ioa}.
  - Choose $o \in O_i$, using up capacity s_{ioa} in *all* a.

Thm[FHKMS10,AWY]: W.h.p, the PD algorithm is a $(1 - O(\epsilon))$-aprx, as long as items have low value ($v_{io} \leq \frac{\epsilon \mathrm{OPT}}{\log n}$) and small size ($s_{ioa} \leq \frac{\epsilon^3 C_a}{\log n}$).

General Stochastic Packing LPs

- m fixed *resources* with capacity C_a
- *Items* i arrive online with *options* O_i, *values* v_{io}, *rsrc. use* s_{ioa}.
  - Choose $o \in O_i$, using up capacity s_{ioa} in *all* a.

Thm[FHKMS10,AWY]: W.h.p, the PD algorithm is a $(1-O(\epsilon))$-aprx, as long as items have low value ($v_{io} \leq \frac{\epsilon \mathrm{OPT}}{\log n}$) and small size ($s_{ioa} \leq \frac{\epsilon^3 C_a}{\log n}$).

Other Results and Extensions (random order model):

- Agrawal, Wang, Ye: Updating dual variables by periodic solution of the dual program: $C_a \leq \frac{m \log n}{\epsilon^2}$ or $s_{ioa} \leq \frac{\epsilon^2 C_a}{M}$

General Stochastic Packing LPs

- m fixed *resources* with capacity C_a
- *Items* i arrive online with *options* O_i, *values* v_{io}, *rsrc. use* s_{ioa}.
  - Choose $o \in O_i$, using up capacity s_{ioa} in *all* a.

Thm[FHKMS10,AWY]: W.h.p, the PD algorithm is a $(1-O(\epsilon))$-aprx, as long as items have low value ($v_{io} \leq \frac{\epsilon \mathrm{OPT}}{\log n}$) and small size ($s_{ioa} \leq \frac{\epsilon^3 C_a}{\log n}$).

Other Results and Extensions (random order model):

- Agrawal, Wang, Ye: Updating dual variables by periodic solution of the dual program: $C_a \leq \frac{m \log n}{\epsilon^2}$ or $s_{ioa} \leq \frac{\epsilon^2 C_a}{M}$
- Vee, Vassilvitskii , Shanmugasundaram 2010: extension to convex objective functions: Using KKT conditions.

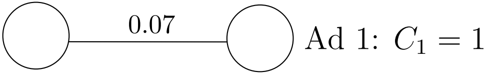
Ad Allocation: Problems and Models

	Online Matching: $v_{ia} = s_{ia} = 1$	Disp. Ads (DA): $s_{ia} = 1$	AdWords (AW): $s_{ia} = v_{ia}$
Worst Case	Greedy: $\frac{1}{2}$, [KVV]: $1-\frac{1}{e}$-aprx	Inapproximable ?	[MSVV,BJN]: $1-\frac{1}{e}$-aprx
Stochastic (i.i.d.)	[FMMM09]: 0.67-aprx i.i.d with known distribution	[FHKMS10,AWY]: $1-\epsilon$-aprx, if $\mathrm{OPT} \gg \max v_{ia}$ and $n \gg m$	[DH09]: $1-\epsilon$-aprx, if $\mathrm{OPT} \gg \max v_{ia}$

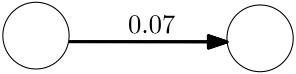
Ad Allocation: Problems and Models

	Online Matching: $v_{ia} = s_{ia} = 1$	Disp. Ads (DA): $s_{ia} = 1$	AdWords (AW): $s_{ia} = v_{ia}$
Worst Case	Greedy: $\frac{1}{2}$, [KVV]: $1 - \frac{1}{e}$-aprx	Inapproximable Free Disposal [FKMMP09]: $1 - \frac{1}{e}$-aprx	[MSVV,BJN]: $1 - \frac{1}{e}$-aprx
Stochastic (i.i.d.)	[FMMM09]: 0.67-aprx i.i.d with known distribution	[FHKMS10,AWY]: $1 - \epsilon$-aprx, if $\text{OPT} \gg \max v_{ia}$ and $n \gg m$	[DH09]: $1 - \epsilon$-aprx, if $\text{OPT} \gg \max v_{ia}$

DA: Free Disposal Model

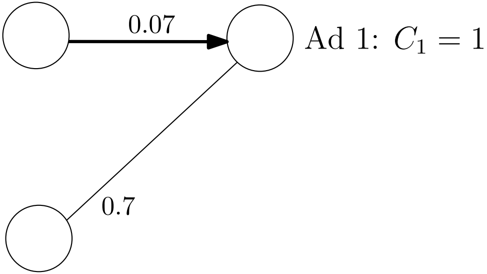


DA: Free Disposal Model

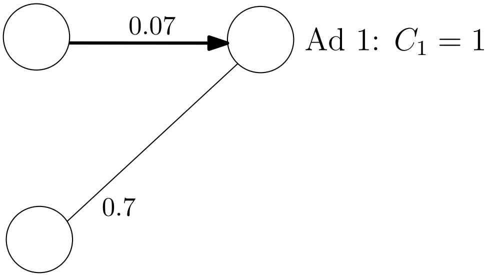


Ad 1: $C_1 = 1$

DA: Free Disposal Model

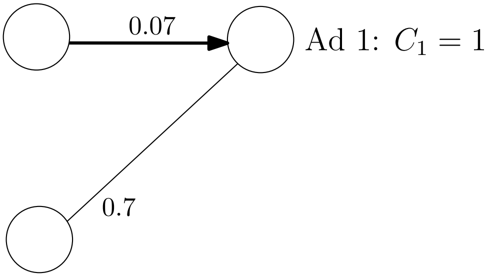


DA: Free Disposal Model



- Advertisers may not complain about extra impressions, but no bonus points for extra impressions, either.

DA: Free Disposal Model



- Advertisers may not complain about extra impressions, but no bonus points for extra impressions, either.
- Value of advertiser = sum of values of top C_a items she gets.

Greedy Algorithm

Assign impression to an advertiser
maximizing Marginal Gain = (imp. value - min. impression value).

Greedy Algorithm

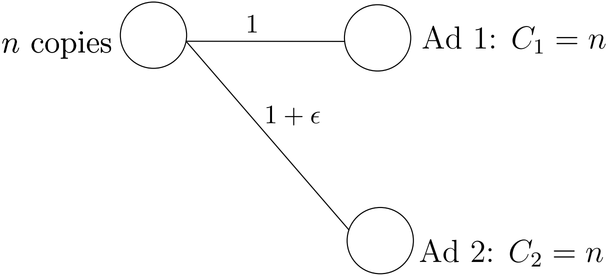
Assign impression to an advertiser
maximizing Marginal Gain = (imp. value - min. impression value).

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

Greedy Algorithm

Assign impression to an advertiser
maximizing Marginal Gain = (imp. value - min. impression value).

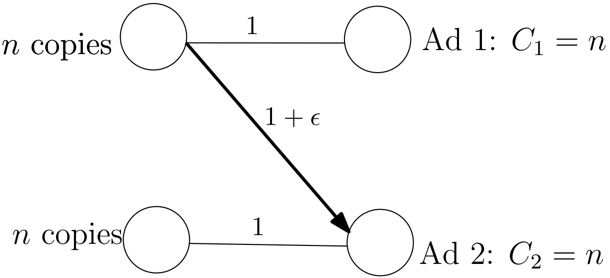
- Competitive Ratio: 1/2. [NWF78]
  - Follows from submodularity of the value function.



Greedy Algorithm

Assign impression to an advertiser
maximizing Marginal Gain = (imp. value - min. impression value).

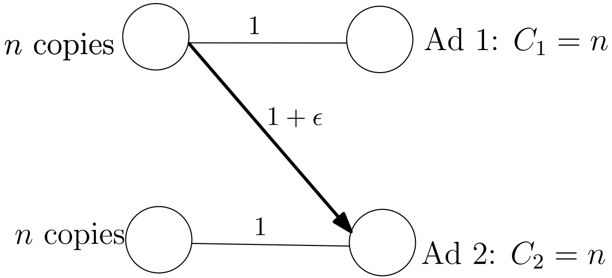
- Competitive Ratio: 1/2. [NWF78]
  - Follows from submodularity of the value function.



Greedy Algorithm

Assign impression to an advertiser
maximizing Marginal Gain = (imp. value - min. impression value).

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



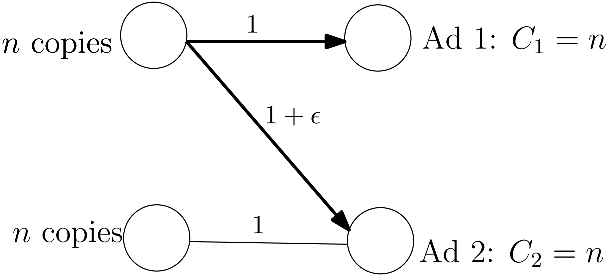
Evenly Split?

A better algorithm?

Assign impression to an advertiser a
maximizing (imp. value - β_a),
where $\beta_a =$ average value of top C_a impressions assigned to a.

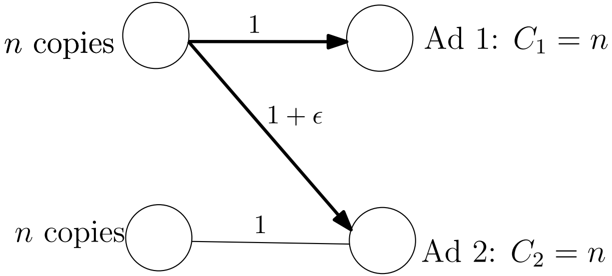
A better algorithm?

Assign impression to an advertiser a
maximizing (imp. value - β_a),
where β_a = average value of top C_a impressions assigned to a.



A better algorithm?

Assign impression to an advertiser a
maximizing (imp. value - β_a),
where $\beta_a =$ average value of top C_a impressions assigned to a.



- Competitive Ratio: $\frac{1}{2}$ if $C_a >> 1$. [FKMMP09]
  - Primal-Dual Approach.

An Optimal Algorithm

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.

An Optimal Algorithm

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: $v(1) \geq v(2) \ldots \geq v(C_a)$:

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

An Optimal Algorithm

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 $C_a > O(\frac{1}{\epsilon})$. [Feldman, Korula, M., Muthukrishnan, Pal 2009]

Online Generalized Assignment (with free disposal)

- Multiple Knapsack: Item i may have different value (v_{ia}) and different size s_{ia} for different ads a.
- DA: $s_{ia} = 1$, AW: $v_{ia} = s_{ia}$.

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

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

Online Generalized Assignment (with free disposal)

- Multiple Knapsack: Item i may have different value (v_{ia}) and different size s_{ia} for different ads a.
- DA: $s_{ia} = 1$, AW: $v_{ia} = s_{ia}$.

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

- Offline Optimization: $1 - \frac{1}{e} - \delta$-aprx[FGMS07,FV08].
- Thm[FKMMP09]: There exists a $1 - \frac{1}{e} - \epsilon$-approximation algorithm if $\frac{C_a}{\max s_{ia}} \geq \frac{1}{\epsilon}$.

Proof Idea: Primal-Dual Analysis [BJN]

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

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

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

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

Proof Idea: Primal-Dual Analysis [BJN]

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

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

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

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

- Proof:
  1. Start from feasible primal and dual ($x_{ia} = 0$, $\beta_a = 0$, and $z_i = 0$, i.e., Primal=Dual=0).
  2. After each assignment, update x, β, z variables and keep primal and dual solutions.
  3. Show $\Delta(\text{Dual}) \leq (1 - \frac{1}{e})\Delta(\text{Primal})$.

Ad Allocation: Problems and Models

	Online Matching: $v_{ia} = s_{ia} = 1$	Disp. Ads (DA): $s_{ia} = 1$	AdWords (AW): $s_{ia} = v_{ia}$
Worst Case	Greedy: $\frac{1}{2}$, [KVV]: $1 - \frac{1}{e}$-aprx	Inapproximable Free Disposal [FKMMP09]: $1 - \frac{1}{e}$-aprx	[MSVV,BJN]: $1 - \frac{1}{e}$-aprx
Stochastic (random arrival order)	[FMMM09]: 0.67-aprx	[FHKMS10,AWY]: $1 - \epsilon$-aprx, if $\text{OPT} \gg \max v_{ia}$ and $n \gg m$	[DH09]: $1 - \epsilon$-aprx, if $\text{OPT} \gg \max v_{ia}$

Outline: Online Allocation

- Online Stochastic Assignment Problems
  - Online (Stochastic) Matching
  - Online Generalized Assignment (with free disposal)
  - Online Stochastic Packing
  - Experimental Evaluation
- Online Learning and Allocation

Dual-based Algorithms in Practice

- Algorithm:
  - Assign each item i to ad a that maximizes $v_{ia} - \beta_a$.

Dual-based Algorithms in Practice

- Algorithm:
  - Assign each item i to ad a that maximizes $v_{ia} - \beta_a$.

- More practical compared to Primal Algorithms:
  - Just keep one number β_a per advertiser.
  - Suitable for Distributed Ad Serving Schemes.

Dual-based Algorithms in Practice

- Algorithm:
  - Assign each item i to ad a that maximizes $v_{ia} - \beta_a$.

- More practical compared to Primal Algorithms:
  - Just keep one number β_a per advertiser.
  - Suitable for Distributed Ad Serving Schemes.
- Training-based Algorithms
  - Compute β_a based on historical/sample data.

Dual-based Algorithms in Practice

- Algorithm:
  - Assign each item i to ad a that maximizes $v_{ia} - \beta_a$.

- More practical compared to Primal Algorithms:
  - Just keep one number β_a per advertiser.
  - Suitable for Distributed Ad Serving Schemes.
- Training-based Algorithms
  - Compute β_a based on historical/sample data.
- Hybrid approach (see also [MNS07]):
  - Start with trained β_a (past history), blend in online algorithm.

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
- Efficiency: free disposal model
- 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%
opt	100
greedy	69
pd-avg	77
pd-exp	82
dualbase	87
hybrid	89

- pd-exp & pd-avg outperform greedy by 9% and 14% (with more improvements in *tight* competition.)
- dualbase outperforms pure online algorithms by 6% to 12%.
- Hybrid has a mild improvement of 2% (up to 10%).
- pd-avg performs much better than the theoretical analysis.

Other Metrics: Fairness

- Qualititative definition: advertisers are "treated equally."

Other Metrics: Fairness

- Qualititative definition: advertisers are "treated equally."
- Quantitative definition that achieves this:
  - varied and often subjective.

Other Metrics: Fairness

- Qualititative definition: advertisers are “treated equally.”
- Quantitative definition that achieves this:
  - varied and often subjective.

- One suggestion[FHKMS10]: Compute "fair" solution x^*, measure ℓ_1 distance to x^*.

Other Metrics: Fairness

- Qualititative definition: advertisers are "treated equally."
- Quantitative definition that achieves this:
  - varied and often subjective.

- One suggestion[FHKMS10]: Compute "fair" solution x^*, measure ℓ_1 distance to x^*.
- Fair solution:
  - Each a chooses best C_a impressions (highest v_{ia})
  - Repeat:
    - Impressions shared among those who chose them.
    - If some a not receiving C_a imps, a chooses an additional imp.

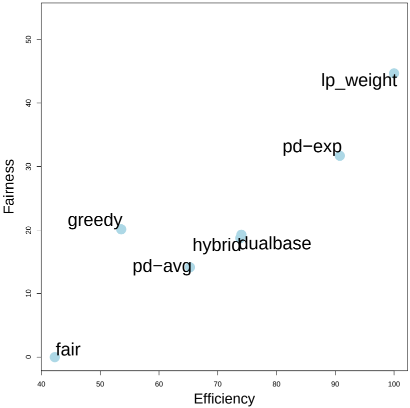
Other Metrics: Fairness

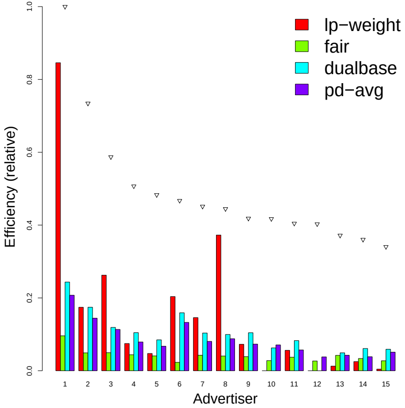
- Qualititative definition: advertisers are "treated equally."
- Quantitative definition that achieves this:
  - varied and often subjective.

- One suggestion[FHKMS10]: Compute "fair" solution x^*, measure ℓ_1 distance to x^*.
- Fair solution:
  - Each a chooses best C_a impressions (highest v_{ia})
  - Repeat:
    - Impressions shared among those who chose them.
    - If some a not receiving C_a imps, a chooses an additional imp.

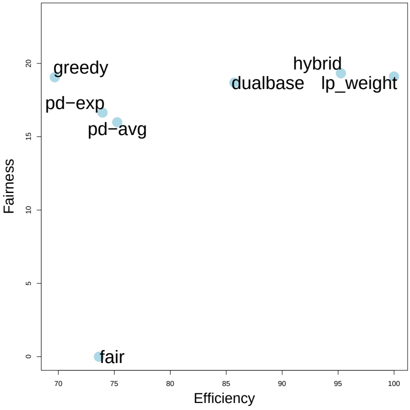
- Sharing policies:
  - Equal: all interested advertisers share equally
  - Proportional: share $\sim v_{ia}$.
  - Stable matching: highest v_{ia} gets all. [Thm: eff $\geq \text{OPT}/2$]

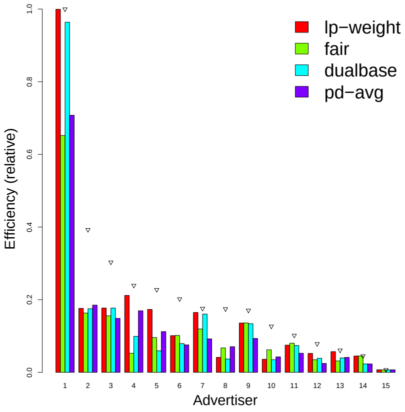
Experiments: highlights





Experiments: highlights





Online Ad Allocation: Interesting Problems

- Online Stochastic DA:
  - Simultaneous online worst-case & stochastic optimization.
  - Bicriteria fairness, efficiency analysis
  - Tradeoff between delivery penalty and efficiency
  - More complex stochastic modeling (drift, seasonality, etc.)
  - Practical utility of primal algorithms?

Online Ad Allocation: Interesting Problems

- Online Stochastic DA:
  - Simultaneous online worst-case & stochastic optimization.
  - Bicriteria fairness, efficiency analysis
  - Tradeoff between delivery penalty and efficiency
  - More complex stochastic modeling (drift, seasonality, etc.)
  - Practical utility of primal algorithms?
- Online matching:
  - Power of 3 choices?
  - Gap between lower and upper bound ($0.67 < 0.98$).
  - Apply "power of 2 choices" in stochastic optimization.

Results: Three Recent Papers

- Online Stochastic Matching: Beating $1 - \frac{1}{e}$, FOCS 2009.
  - online stochastic matching in iid model with known dist.
  - 0.67-approximation (idea: power of two choices)
  - Feldman, Mehta, M., Muthukrishnan

Results: Three Recent Papers

- Online Stochastic Matching: Beating $1 - \frac{1}{e}$, FOCS 2009.
  - online stochastic matching in iid model with known dist.
  - 0.67-approximation (idea: power of two choices)
  - Feldman, Mehta, M., Muthukrishnan

- Online Stochastic Packing applied to Display Ad Allocation, ESA 2010.
  - Online stoch. packing in random order model: online routing.
  - $1 - \epsilon$-approximation under assumptions (idea: learn dual variables.)
  - Feldman, Henzinger, Korula, M., Stein

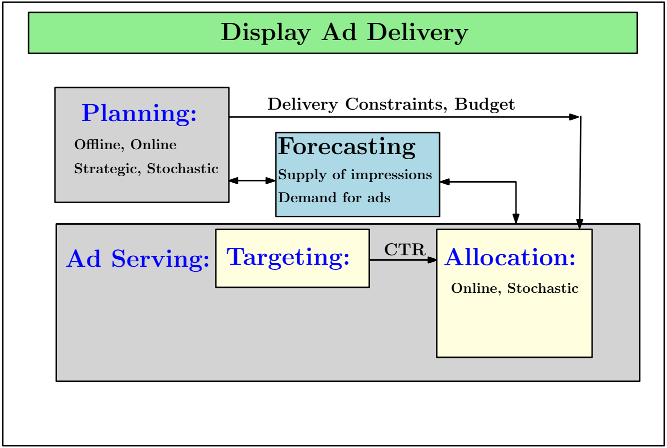
Results: Three Recent Papers

- Online Stochastic Matching: Beating $1 - \frac{1}{e}$, FOCS 2009.
  - online stochastic matching in iid model with known dist.
  - 0.67-approximation (idea: power of two choices)
  - Feldman, Mehta, M., Muthukrishnan
- Online Stochastic Packing applied to Display Ad Allocation, ESA 2010.
  - Online stoch. packing in random order model: online routing.
  - $1 - \epsilon$-approximation under assumptions (idea: learn dual variables.)
  - Feldman, Henzinger, Korula, M., Stein
- Online Ad Assignment with Free Disposal, WINE 2009.
  - online generalized assignment problems with free disposal.
  - $1 - \frac{1}{e}$-competitive algorithm (idea: primal-dual analysis.)
  - Feldman, Korula, M., Muthukrishnan, Pal

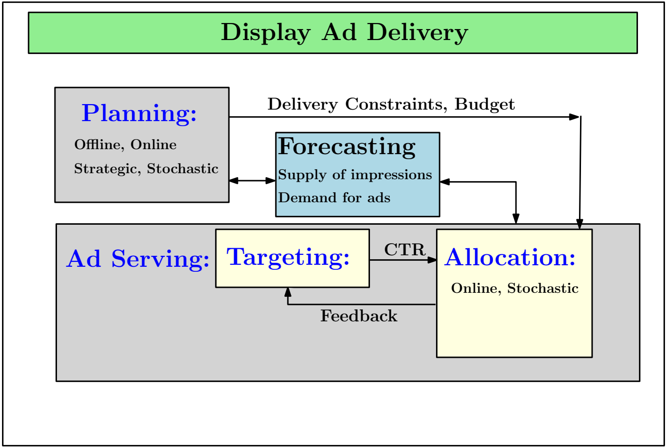
Outline: Online Allocation

- Online Stochastic Assignment Problems
  - Online (Stochastic) Matching
  - Online Generalized Assignment (with free disposal)
  - Online Stochastic Packing
  - Experimental Results
- Online Learning and Allocation

Display Ad Delivery



Display Ad Delivery



Online Learning & Allocation

- Value: Estimated Click-Through-Rate (CTR).

Online Learning & Allocation

- Value: Estimated Click-Through-Rate (CTR).
- Combined online capacity planning & learning?
  - Budgeted Active Learning
    - Madani, Lizotte, Greiner 2004, Active Model Selection.

Online Learning & Allocation

- Value: Estimated Click-Through-Rate (CTR).
- Combined online capacity planning & learning?
  - Budgeted Active Learning
    - Madani, Lizotte, Greiner 2004, Active Model Selection.
  - Bayesian Budgeted Multi-armed Bandits:
    - Guha, Munagala, Multi-armed Bandits with Metric Switching Costs.
    - Goel, Khanna, Null, The Ratio Index for Budgeted Learning, with Applications.
    - Guha, Munagala, Pal, Multi-armed Bandit with Delayed Feedback.

Online Learning & Allocation

- Value: Estimated Click-Through-Rate (CTR).
- Combined online capacity planning & learning?
  - Budgeted Active Learning
    - Madani, Lizotte, Greiner 2004, Active Model Selection.
  - Bayesian Budgeted Multi-armed Bandits:
    - Guha, Munagala, Multi-armed Bandits with Metric Switching Costs.
    - Goel, Khanna, Null, The Ratio Index for Budgeted Learning, with Applications.
    - Guha, Munagala, Pal, Multi-armed Bandit with Delayed Feedback.
  - Budgeted Unknown-CTR Multi-armed Bandit
    - Pandey, Olston 2007, Handling Advertisement of Unknown Quality.

Online CTR Learning: Mixed Explore/Exploit

- Pandey, Olston 2007, Handling Advertisement of Unknown Quality.

Online CTR Learning: Mixed Explore/Exploit

- Pandey, Olston 2007, Handling Advertisement of Unknown Quality.
- Algorithm: Revised Greedy
  - Upon arrival of query of type i, assign it to an ad a maximizing $P_{ia} = (\hat{c}_{ia} + \sqrt{\frac{2 \ln n_i}{n_{ia}}}) b_{ia}$
    where $\hat{c}_{ia}$ is the current estimate of CTR, n_{ia} is the number of times i has been assigned to a, n_i is the number of queries of type i so far.

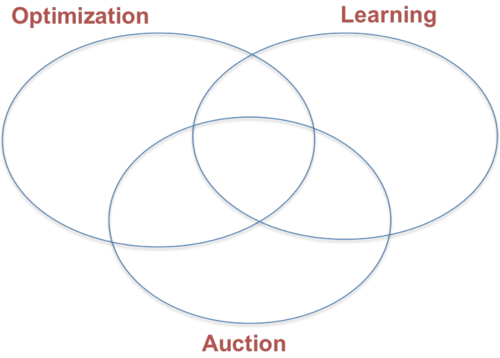
Online CTR Learning: Mixed Explore/Exploit

- Pandey, Olston 2007, Handling Advertisement of Unknown Quality.
- Algorithm: Revised Greedy
  - Upon arrival of query of type i, assign it to an ad a maximizing $P_{ia} = (\hat{c}_{ia} + \sqrt{\frac{2 \ln n_i}{n_{ia}}}) b_{ia}$ where $\hat{c}_{ia}$ is the current estimate of CTR, n_{ia} is the number of times i has been assigned to a, n_i is the number of queries of type i so far.
- Thm[PO07]: ALG $\geq \frac{\text{OPT}}{2} - O(\ln n)$ where n is the number of arrivals.

Outline of this talk

- Ad serving in repeated auction settings
  - General architecture.
  - Allocation for budget constrained advertisers.
- Ad delivery for contract based settings
  - Planning
  - Ad Serving
- Other interactions
  - Learning + allocation
  - Learning + auction
  - Auction + contracts

Three main theory/practice problems



Outline

Learning + Alloc

Hybrid ad serving

Online Learning & Allocation

- Value: Estimated Click-Through-Rate (CTR).

Online Learning & Allocation

- Value: Estimated Click-Through-Rate (CTR).
- Combined online capacity planning & learning?
  - Budgeted Active Learning
    - Madani, Lizotte, Greiner 2004, Active Model Selection.

Online Learning & Allocation

- Value: Estimated Click-Through-Rate (CTR).
- Combined online capacity planning & learning?
  - Budgeted Active Learning
    - Madani, Lizotte, Greiner 2004, Active Model Selection.
  - Bayesian Budgeted Multi-armed Bandits:
    - Guha, Munagala, Multi-armed Bandits with Metric Switching Costs.
    - Goel, Khanna, Null, The Ratio Index for Budgeted Learning, with Applications.
    - Guha, Munagala, Pal, Multi-armed Bandit with Delayed Feedback.

Online Learning & Allocation

- Value: Estimated Click-Through-Rate (CTR).
- Combined online capacity planning & learning?
  - Budgeted Active Learning
    - Madani, Lizotte, Greiner 2004, Active Model Selection.
  - Bayesian Budgeted Multi-armed Bandits:
    - Guha, Munagala, Multi-armed Bandits with Metric Switching Costs.
    - Goel, Khanna, Null, The Ratio Index for Budgeted Learning, with Applications.
    - Guha, Munagala, Pal, Multi-armed Bandit with Delayed Feedback.
  - Budgeted Unknown-CTR Multi-armed Bandit
    - Pandey, Olston 2007, Handling Advertisement of Unknown Quality.

Online CTR Learning: Mixed Explore/Exploit

- Pandey, Olston 2007, Handling Advertisement of Unknown Quality.

Online CTR Learning: Mixed Explore/Exploit

- Pandey, Olston 2007, Handling Advertisement of Unknown Quality.
- Algorithm: Revised Greedy
  - Upon arrival of query of type i, assign it to an ad a maximizing $P_{ia} = (\hat{c}_{ia} + \sqrt{\frac{2 \ln n_i}{n_{ia}}}) b_{ia}$
    where $\hat{c}_{ia}$ is the current estimate of CTR, n_{ia} is the number of times i has been assigned to a, n_i is the number of queries of type i so far.

Online CTR Learning: Mixed Explore/Exploit

- Pandey, Olston 2007, Handling Advertisement of Unknown Quality.
- Algorithm: Revised Greedy
  - Upon arrival of query of type i, assign it to an ad a maximizing $P_{ia} = (\hat{c}_{ia} + \sqrt{\frac{2 \ln n_i}{n_{ia}}}) b_{ia}$ where $\hat{c}_{ia}$ is the current estimate of CTR, n_{ia} is the number of times i has been assigned to a, n_i is the number of queries of type i so far.
- Thm[PO07]: ALG $\geq \frac{\text{OPT}}{2} - O(\ln n)$ where n is the number of arrivals.

Online Learning & Auction Incentives

[Devanur,Kakade'09, Babaioff,Sharma,Slivkins'09]

- Multi-Armed Bandit algorithms achieve an "implicit" exploration-exploitation tradeoff to get a regret of $O(\sqrt{T})$ (e.g., UCB).
- Can these be run in tandem with truthful auctions? (e.g., 2nd price for a single slot).

Online Learning & Auction Incentives

[Devanur,Kakade'09, Babaioff,Sharma,Slivkins'09]

- Multi-Armed Bandit algorithms achieve an "implicit" exploration-exploitation tradeoff to get a regret of $O(\sqrt{T})$ (e.g., UCB).
- Can these be run in tandem with truthful auctions? (e.g., 2nd price for a single slot).
- A naive explore-exploit method gets $O(T^{2/3})$ regret:
  - Explore ads for the first *phase*, giving them out for free.
  - Fix the CTRs thus learned in the first phase.
  - Run 2nd price auction for the 2nd phase.

Online Learning & Auction Incentives

[Devanur,Kakade'09, Babaioff,Sharma,Slivkins'09]

- Multi-Armed Bandit algorithms achieve an "implicit" exploration-exploitation tradeoff to get a regret of $O(\sqrt{T})$ (e.g., UCB).
- Can these be run in tandem with truthful auctions? (e.g., 2nd price for a single slot).
- A naive explore-exploit method gets $O(T^{2/3})$ regret:
  - Explore ads for the first *phase*, giving them out for free.
  - Fix the CTRs thus learned in the first phase.
  - Run 2nd price auction for the 2nd phase.
- Can you do better that this simpe decoupling?
- No!

Theorem

[DK09,BSS09] For every truthful auction (under certain assumptions), there exist bids, ctrs, s.t. regret $= \Omega(T^{2/3})$.

Outline

Learning + Alloc

Hybrid ad serving

Hybrid ad serving: Contracts + Spot Auctions

Given a page view, and two types of advertisers:

- Contract-based.
- Auction-based.

Hybrid ad serving: Contracts + Spot Auctions

Given a page view, and two types of advertisers:

- Contract-based.
- Auction-based.

- Decide who wins and how much do they pay.
- Requirements:
  - For each contract-advertiser, meet its demand.
  - Implement the scheme using proxy-bidding for contract-advertisers in the spot auction.

Hybrid ad serving: Contracts + Spot Auctions

- Naive solution: If a contract-adv is eligible and has not finished demand, then let it win the spot. Bid infinity for all auctions.

Hybrid ad serving: Contracts + Spot Auctions

- Naive solution: If a contract-adv is eligible and has not finished demand, then let it win the spot. Bid infinity for all auctions.

- Optimize for revenue: If the auction pressure (price) is low then let the contract-adv win. Bid a low bid for all auctions.

Hybrid ad serving: Contracts + Spot Auctions

- Naive solution: If a contract-adv is eligible and has not finished demand, then let it win the spot. Bid infinity for all auctions.

- Optimize for revenue: If the auction pressure (price) is low then let the contract-adv win. Bid a low bid for all auctions.
  - Unfair to contract-adv, since low auction-price $\Rightarrow$ it is a lower value impression.

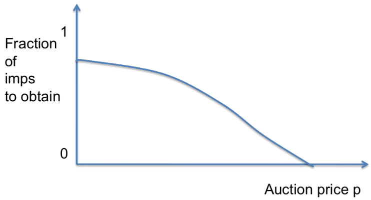
Hybrid ad serving: Contracts + Spot Auctions

- Naive solution: If a contract-adv is eligible and has not finished demand, then let it win the spot. Bid infinity for all auctions.
- Optimize for revenue: If the auction pressure (price) is low then let the contract-adv win. Bid a low bid for all auctions.
  - Unfair to contract-adv, since low auction-price $\Rightarrow$ it is a lower value impression.
- Ideally:
  - Provide contract-adv with a representative allocation, an equal slice of impressions from each price-point.
  - A price-oblivious scheme, i.e., bid without seeing the auction bids.
  - Revenue per auction: average auction-price of impressions given away to contract-advertisers is at most some target t.

Obtaining representative allocations

Two main ideas:

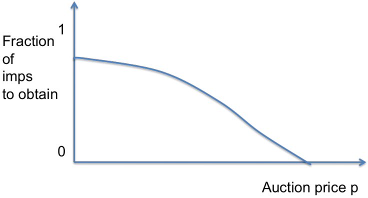
1. Can implement any decreasing function $a(p)$ for fraction of impressions of auction-price p.



Obtaining representative allocations

Two main ideas:

1. Can implement any decreasing function $a(p)$ for fraction of impressions of auction-price p.



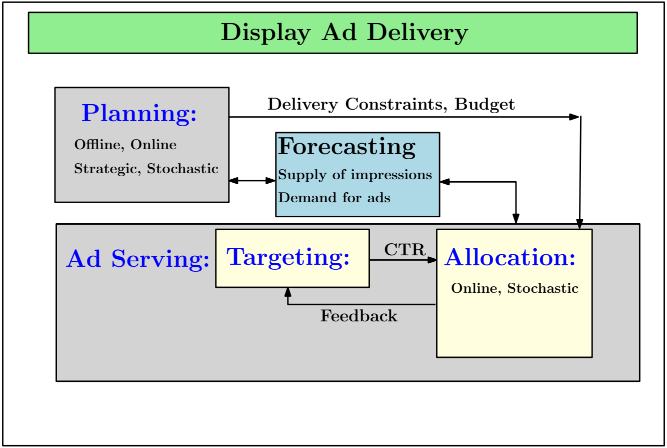
2. Solve the system for well chosen distance functions:

$$\text{Minimize dist(U, a)}$$

$$\text{s.t.:} \int_p a(p)f(p)dp = d$$

$$\int_p pa(p)f(p)dp \leq td$$

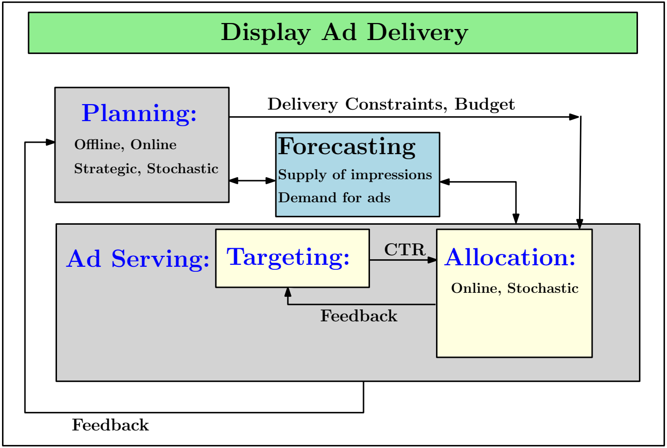
Display Ad Delivery



Open Problems:

- **Optimal combined online allocation & learning.**
- **Feature selection and correlation in learning CTR.**

Display Ad Delivery



Open Problems:

- **Optimal combined online allocation & learning.**
- **Feature selection and correlation in learning CTR.**
- **Optimal combined stochastic planning and serving?**

Thank You