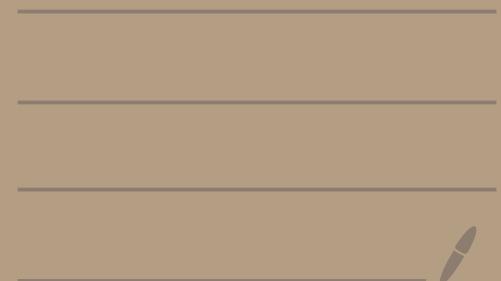


Game Theory (CS 4187)

Lecture 38 (28/11/2024)



> I have recorded the following YouTube video for this lecture notes because I couldn't complete all the topics of this lecture in class:

<https://youtu.be/y2JmkBuHBac>

> Very important from exam perspective.

Sponsored Search Auctions

apartments in Kompally ← Search query.

Sponsored
hivisionheights.com
<https://www.hivisionheights.com>
Hivision Residency in Kompally - Spacious 2 & 3 BHK...
Claim Free Interiors Worth ₹5 Lakhs With Your New Home Purchase at Hivision Heights!*
Apartment building · Hyderabad · Open · Closes 6 pm
Contact US · About Hivision Heights · Location Benefits · Project Highlights
[Call us](#)

Sponsored
rubrick.in
[https://www.rubbrick.in](https://www.rubrick.in)
Luxury flats sale in Kompally - Managed by TATA Projects.
Rubrick Sriveni Tripura: 5.24 Acres, 3 Towers, 60+ Amenities for a Unique Living Experience
Survey No. 494, 495 & 497, Gowdavelli Village, Medchal Mandal, Medchal Malkajgiri District...
[Call us](#)

Sponsored
omsree.com
<https://www.omsree.com>
Om Sree Galaxy at Kompally | 2 & 3 BHK Flats at Kompally
Explore Galaxy's premium 2 & 3 BHKs in **Kompally**, starting at ₹90L. Book your visit.
Download Brochure · Project Gallery · Site Layout · Project Specifications · Amenities

99 99 Acres
<https://www.99acres.com> › ... › Flats in Hyderabad
180+ Apartments / Flats for Sale in Kompally, Hyderabad

Om Sree Sky Park, one of the known housing societies in Hyderabad, brings classic yet modern 3,4 BHK **apartments in Kompally**. Here units have different sizes,



MagicBricks
<https://www.magicbricks.com> › flats-in-kompally-hydera...
Flats for sale in Kompally Hyderabad
Find over 848+ **Flats in Kompally** Hyderabad at MagicBricks. Search & Buy 1/2/3/4/5 BHK Apartments / Flats for sale in Kompally in different price range.
₹58,20,000.00 to ₹5,00,00,000.00
2 BHK Flats in Kompally · Flats for Sale in Kompally... · 3 BHK Flats in Kompally



NoBroker
<https://www.nobroker.in> › flats-for-sale-in-kompally_h...
Flats for Sale in Kompally Hyderabad
315 - Flats for Sale in Kompally Hyderabad | **Apartments in Kompally** for Sale Without Brokerage · 3 BHK Flat In Cascade Greens For Sale In Kompally · 2 BHK Flat ...



← Sponsored Ads.

Three Ad.

Positions.

Sponsored Search Auctions

Setup:

- > An user searches a query in a search engine.
- > This search query gets linked to a set of advertisers,
 $N = \{1, 2, \dots, n\}$
- > There are m slots to advertise. Each advertiser can get **at most one slot**.
- > The search engine uses **pay per click** model, i.e. an advertiser has to pay only if its advertisement gets clicked; NOT for just showing the advertisement.

Sponsored Search Auctions

Setup:

- > d_{ij} is the probability that a user will click on the i^{th} advertiser advertisement if it is in the j^{th} position ($j=1$ is the top-most position). For all $j \in N$, d_{ij} is non-increasing in j .
- > When an user click the advertisement of i^{th} advertiser, it derives some value $\theta_i \in R^+$ from it. θ_i may be the average money an user will spend on the advertiser's website. θ_i 's are private information.

Sponsored Search Auctions

Setup:

- > $\theta = (\theta_1, \theta_2, \dots, \theta_n)$. We have $\theta \in \Theta$. Also,

$$P(\Theta) = \prod_{i=1}^n P(\theta_i)$$

} Not needed for
DSIC mechanisms.
Why?

Assumptions:

- i) All clicks are valued equally no matter which position the advertisement is displayed, i.e. θ_j does not depend on advertisement position j . Valid?

Sponsored Search Auctions

Setup:

- > $\theta = (\theta_1, \theta_2, \dots, \theta_n)$. We have $\theta \in \Theta$. Also,

$$P(\Theta) = \prod_{i=1}^n P(\theta_i)$$

} Not needed for
DSIC mechanisms.
Why?

Assumptions:

ii) d_{ij} are knowns to search engine. Valid?

iii) d_{ij} does not depend on which other advertiser has been allocated to what other position.

VCG Mechanism For Sponsored Search Auction

Formulation:

- 1) Players: Advertisers. Set of advertisers $N = \{1, 2, \dots, n\}$.
- 2) Type: θ_i which is the value to advertiser i if the advertisement is clicked.
The search engine does NOT know θ_i . Advertiser i bids b_i . If DSIC, then $b_i = \theta_i$.

VCG Mechanism For Sponsored Search Auction

Formulation:

3) Outcome: Let $y_{i,j} \in \{0, 1\}$ denote whether the j^{th} advertisement slot is allocated to the i^{th} user. Let Y be a matrix whose $(i,j)^{\text{th}}$ component is y_{ij} .

Let $p = (p_1, p_2, \dots, p_n)$ where p_i is the payment of advertiser if an user clicks its advertisement.

$$X = \left\{ (Y, p) : y_{i,j} \in \{0, 1\}, \forall i, j ; \sum_{i=1}^n y_{ij} \leq 1, \forall j ; \sum_{j=1}^m y_{ij} \leq 1, \forall i ; p_i \in \mathbb{R}, \forall i \right\}$$

VCG Mechanism For Sponsored Search Auction

Formulation:

4) Utility Function:

$$U_i((\gamma, p), \theta_i) = \sum_{j=1}^m \left(y_{ij} \alpha_{ij} (\theta_i - p_j) \right)$$

Note: The utility function is NOT in quasi-linear form, i.e.

$v_i(a, \theta_i) - p_i$! So, in order to use VCG mechanism, we have to first convert the utility function in quasi-linear form. That's what we do next.

VCG Mechanism For Sponsored Search Auction

Formulation:

4) Utility Function:

$$U_i((\gamma, p), \theta_i) = \sum_{j=1}^m \left(y_{ij} \alpha_{ij} (\theta_i - p_j) \right)$$

$$= \sum_{j=1}^m \left(y_{ij} \alpha_{ij} \theta_i \right) - \sum_{j=1}^m \left(y_{ij} \alpha_{ij} p_j \right)$$

$$= \theta_i \sum_{j=1}^m \left(y_{ij} \alpha_{ij} \right) - p_i \sum_{j=1}^m \left(y_{ij} \alpha_{ij} \right)$$

VCG Mechanism For Sponsored Search Auction

Formulation:

4) Utility Function:

$$U_i((Y, P), \theta_i) = \theta_i \sum_{j=1}^m (y_{ij} \alpha_{ij}) - p_i \sum_{j=1}^m (y_{ij} \alpha_{ij})$$

$$= V_i(Y, \theta_i) - t_i \quad \text{where}$$

$$V_i(Y, \theta_i) = \theta_i \sum_{j=1}^m (y_{ij} \alpha_{ij}),$$

$$t_i = p_i \sum_{j=1}^m (y_{ij} \alpha_{ij})$$

VCG Mechanism For Sponsored Search Auction

Formulation:

4) Utility Function:

$$U_i((Y, P), \theta_i) = V_i(Y, \theta_i) - t_i$$

The above utility function is in quasi-linear form. According to the above utility function, the outcome is Y and $T = (t_1, t_2, \dots, t_n)$ and NOT Y and P . But wait! We finally need P_i 's because P_i is what i^{th} advertiser has to pay. How do we get P_i ?

VCG Mechanism For Sponsored Search Auction

Formulation:

4) Utility Function:

$$U_i((\gamma, p), \theta_i) = V_i(\gamma, \theta_i) - t_i$$

How do we get p_i ?

We first compute γ and t_i . Then use the following to compute p_i :

$$t_i = p_i \sum_{j=1}^m (y_{ij} \alpha_{ij})$$

VCG Mechanism For Sponsored Search Auction

Allocation Rule:

$$a^*(\theta) = \operatorname{argmax}_{a \in A} \sum_{i \in N} v_i(a, \theta_i)$$

For sponsored search auction, the above formula becomes,

$$y^*(\theta) = \operatorname{argmax}_y \sum_{i=1}^n v_i(y, \theta_i)$$

$$= \operatorname{argmax}_y \sum_{i=1}^n \left(\theta_i \sum_{j=1}^m (y_{ij} \alpha_{ij}) \right)$$

$$= \operatorname{argmax}_y \sum_{i=1}^n \sum_{j=1}^m (\theta_i \alpha_{ij}) y_{ij}$$

VCG Mechanism For Sponsored Search Auction

Allocation Rule:

$$y^*(\theta) = \arg \max_y \sum_{i=1}^n \sum_{j=1}^m (\theta_i \alpha_{ij}) y_{ij}$$

1	2	m
2				
.				
.				
n				

Pseudocode:

- 1) Let Z be a matrix whose (i,j) th element $z_{i,j} = \theta_i \alpha_{ij}$.
Set $y_{i,j} = 0, \forall i,j$.
- 2) For $j = 1, 2, \dots, m$.
- 3) Find the largest element of Z . Let it be the element (i^*, j^*) . Then set, $y_{i^*, j^*} = 1$.
- 4) Remove row i^* and column j^* from Z .

VCG Mechanism For Sponsored Search Auction

Allocation Rule:

$$y^*(\theta) = \arg \max_{Y} \sum_{i=1}^n \sum_{j=1}^m (\theta_i; \alpha_{ij}) y_{ij}$$

$\sum_j y_{ij}$

	1	2	m
1					
2					
..					
i					
..					
n					

Remember: For all $i \in N$, $\alpha_{i,1} \geq \alpha_{i,2} \geq \dots \geq \alpha_{i,m}$.

Pseudocode (better):

- 1) Set $S = N$. Set $y_{i,j} = 0$, $\forall i, j$.
- 2) For $j = 1, 2, \dots, m$.
- 3) $i^* = \arg \max_{i \in S} \theta_i \alpha_{ij}$
- 4) Set $y_{i^*,j} = 1$. Also, $S = S \setminus \{i^*\}$.

VCG Mechanism For Sponsored Search Auction

Allocation Rule:

$$y^*(\theta) = \arg \max_y \sum_{i=1}^n \sum_{j=1}^m (\theta_i; \alpha_{ij}) y_{ij}$$

> Special case: If $\alpha_{1,j} = \alpha_{2,j} = \dots = \alpha_{n,j} = \alpha_j$, i.e. click probability just depend on the position and NOT on the advertiser.

Pseudocode (special case):

- 1) Sort θ_i 's in descending order. Let $P(j)$ denote the advertiser in the j^{th} position of the sorted list.
- 2) Allocate j^{th} slot to $P(j)$.

VCG Mechanism For Sponsored Search Auction

Allocation Rule:

$$y^*(\theta) = \arg \max_y \sum_{i=1}^n \sum_{j=1}^m (\theta_i \alpha_{ij}) y_{ij}$$

> Wait! We don't know θ_i 's. Then how will any of these allocation rule work?

Answer: We use bids b_i 's instead of θ_i 's. Used θ_i because we are using direct mechanism.

VC G₂ Mechanism For Sponsored Search Auction

Pricing Rule:

$$P_i^*(\theta) = \left(\max_{a \in A} \sum_{j \neq i} V_j(a, \theta_j) \right) - \sum_{j \neq i} V_j(a^*(\theta), \theta_j)$$

For sponsored search auction,

$$t_i^*(\theta) = \left(\max_Y \sum_{K \neq i} V_K(Y, \theta_K) \right) - \sum_{K \neq i} V_K(Y^*(\theta), \theta_K)$$

> Used K because j is used as ad position index.

VCG Mechanism For Sponsored Search Auction

Pricing Rule:

$$t_i^*(\theta) = \left(\max_y \sum_{k \neq i} V_k(y, \theta_k) \right) - \sum_{k \neq i} V_k(y^*(\theta), \theta_k)$$

Now, there can be two cases for player i :

- 1) Case-1: Player i is NOT allocated an ad slot.
- 2) Case-2: Player i is allocated an ad slot.

Let's deal with case-1 First.

VCG Mechanism For Sponsored Search Auction

Pricing Rule:

$$t_i^*(\theta) = \left(\max_y \sum_{k \neq i} V_k(y, \theta_k) \right) - \sum_{k \neq i} V_k(y^*(\theta), \theta_k)$$

$\xleftarrow{\text{Term-1}}$ $\xrightarrow{\text{Term-2}}$

Case-1: Player i is NOT allocated an ad slot.

- We prove in the next page that for case-1, term 1 and term 2 are equal.
- Hence, $t_i^*(\theta) = 0$. But, what about $p_i^*(\theta)$? According to,

$$t_i = p_i \sum_{j=1}^m (y_{ij} \alpha_{ij})$$

Any $p_i^*(\theta)$ works. Logically $p_i^*(\theta) = 0$.

$$\sum_{K \neq i} V_k(y^*(\theta), \theta_K) = \sum_{K \neq i} \sum_{j=1}^m (\theta_K \alpha_{kj}) y_{kj}^*$$

For case-i, this term is ZERO.

\longleftrightarrow

Term - 2

$$= \sum_{K \neq i} \sum_{j=1}^m (\theta_K \alpha_{kj}) y_{kj}^* + \underbrace{\sum_{j=1}^m (\theta_i \alpha_{ij}) y_{ij}^*}_{}$$

$$= \sum_{K=1}^n V_k(y^*(\theta), \theta_K)$$

This is \max_y by definition

$$\sum_{K=1}^n V_k(y, \theta_K)$$

OR $y^*(\theta)$.

$$= \max_y \sum_{K=1}^n V_k(y, \theta_K)$$

$$= \max_y \sum_{K \neq i} V_k(y, \theta_K)$$

Since the above term is ZERO.

\longleftrightarrow

Term - 1

VCG Mechanism For Sponsored Search Auction

Pricing Rule:

$$t_i^*(\theta) = \left(\max_y \sum_{k \neq i} V_k(y, \theta_k) \right) - \sum_{k \neq i} V_k(y^*(\theta), \theta_k)$$

$\xleftarrow{\text{Term-1}}$ $\xrightarrow{\text{Term-2}}$

Case-2: Player i is allocated an ad slot.

- > When we are actually implementing a mechanism, we have to worry about its computational complexity.
- > Computational complexity of the pricing rule is NOT high.
 - Once $y^*(\theta)$ is computed, term-2 can be computed in $O(n)$ time.
 - To compute term-1, remove player i and solve the allocation problem again which has $O(nm)$ time complexity.

VCG Mechanism For Sponsored Search Auction

Pricing Rule:

$$t_i^*(\theta) = \left(\max_y \sum_{k \neq i} V_k(y, \theta_k) \right) - \sum_{k \neq i} V_k(y^*(\theta), \theta_k)$$

$\xleftarrow{\text{Term-1}}$ $\xrightarrow{\text{Term-2}}$

Case-2: Player i is allocated an ad slot.

- Here, and in exams we are interested in Simplifying the Payment rule (also allocation rule) as much as possible.
- Such simplification is only possible for the special case,

$$d_{1,j} = d_{2,j} = \dots = d_{n,j} = d_j,$$

- Recall, $p(j)$ denote the advertiser in the j th position of the sorted list.

VCG Mechanism For Sponsored Search Auction

Pricing Rule:

$$t_i^*(\theta) = \left(\max_y \sum_{k \neq i} V_k(y, \theta_k) \right) - \sum_{k \neq i} V_k(y^*(\theta), \theta_k)$$

← Term-1 → ← Term-2 →

Case-2: Player i is allocated an ad slot.

> Then, for case-2, $P(\tilde{j}) = i$ for some \tilde{j} .

$$\sum_{k \neq i} V_k(y^*(\theta), \theta_k) = \sum_{j=1}^{\tilde{j}-1} a_j \cdot \theta_{P(j)} + \sum_{j=\tilde{j}+1}^m a_j \cdot \theta_{P(j)}$$

Only valid when $m < n$.
Think what changes when $m \geq n$.

$$\max_y \sum_{k \neq i} V_k(y, \theta_k) = \sum_{j=1}^{\tilde{j}-1} a_j \cdot \theta_{P(j)} + \sum_{j=\tilde{j}}^m a_j \cdot \theta_{P(j+1)}$$

VCG Mechanism For Sponsored Search Auction

Pricing Rule:

$$t_i^*(\theta) = \left(\max_y \sum_{k \neq i} V_k(y, \theta_k) \right) - \sum_{k \neq i} V_k(y^*(\theta), \theta_k)$$

← Term-1 → ← Term-2 →

Case-2: Player i is allocated an ad slot.

> Then, for case-2, $P(\tilde{j}) = i$ for some \tilde{j} .

$$\sum_{k \neq i} V_k(y^*(\theta), \theta_k) = \sum_{j=1}^{\tilde{j}-1} a_j \cdot \theta_{P(j)} + \sum_{j=\tilde{j}+1}^m a_j \cdot \theta_{P(j)}$$

$$\max_y \sum_{k \neq i} V_k(y, \theta_k) = \sum_{j=1}^{\tilde{j}-1} a_j \cdot \theta_{P(j)} + \sum_{j=\tilde{j}}^m a_j \cdot \theta_{P(j+1)}$$

$P(\cdot)$ is for the original problem-
For $j \geq \tilde{j}$, bidders who were
getting ad position $j+1$ will get
ad position j when advertiser i is
NOT there.

VCG Mechanism For Sponsored Search Auction

Pricing Rule:

$$t_i^*(\theta) = \left(\max_y \sum_{k \neq i} V_k(y, \theta_k) \right) - \sum_{k \neq i} V_k(y^*(\theta), \theta_k)$$

← Term-1 → ← Term-2 →

Case-2: Player i is allocated an ad slot.

> Then, for case-2, $P(\tilde{j}) = i$ for some \tilde{j} .

$$t_i^*(\theta) = d_{\tilde{j}} \cdot \theta_{P(\tilde{j}+1)} + \sum_{j=\tilde{j}+1}^m d_j \cdot (\theta_{P(j)} - \theta_{P(j+1)})$$

Positive quantity. Why?

$$P_i^*(\theta) = \frac{t_i^*(\theta)}{d_{\tilde{j}}} \quad \text{Why? Because } t_i = P_i \sum_{j=1}^m (y_{ij} d_j).$$