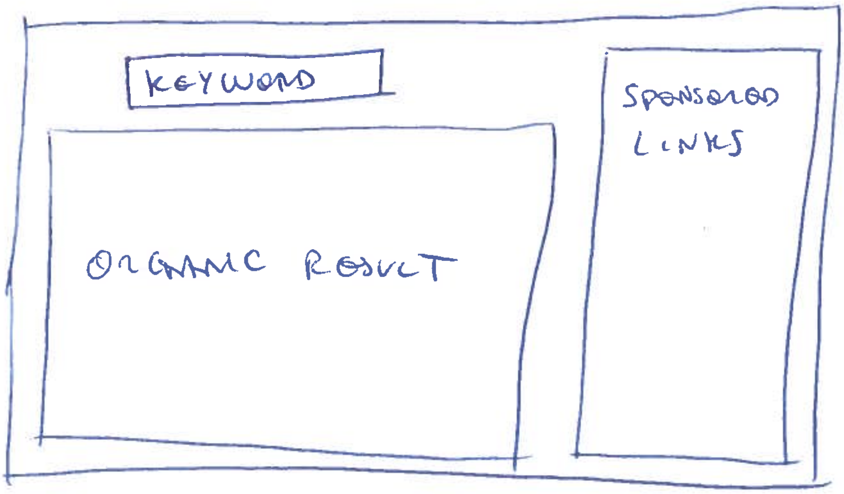
SPONSORED SEARCH AUCTIONS

BASIC MODEL

SCENARIO:



SOMETIMES : MULTIPLE SLOTS

INGREDIENTS:

A USER ⟶ MAKES A QUERY

ADVERTISERS ⟶ BUY SLOTS TO DISPLAY SPONSORED LINKS

SEARCH ENGINES ⟶ DISPLAY ADS

- FOR EACH KEYWORD, AN ADVERTISER DECIDES TO BID OR NOT AND HOW MUCH
- THE SEARCH ENGINE SELECTS THE BEST ADS
- PAY PER CLICK SCHEME

MODEL:

$a_i \in A$ ADVERTISERS

$s_j \in S$ SLOTS

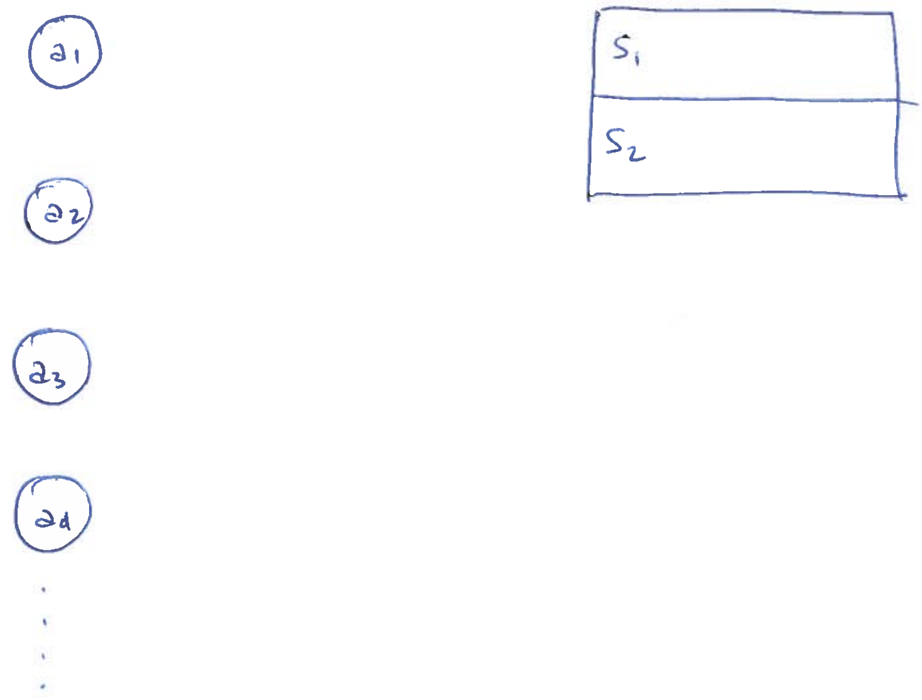
$v_i \in \mathbb{R}_{>0}$ VALUE OF ADVERTISER i (REPORTED TO THE SEARCH ENGINE)

$q_i \in [0,1)$ quality OF ADVERTISER i (ESTIMATED BY THE SEARCH ENGINE)

v = profile OF VALUES $(v_1, v_2 \dots v_n)$

v_{-i} = profile OF VALUES EXCLUDED v_i

XAMPLE:



ROBLEM: WHAT IS THE BEST ALLOCATION OF ADS TO SLOTS?

IT IS A MATCHING PROBLEM

EFINITION: SOCIAL WELFARE. THE UTILITY OF A SINGLE ADV IS $q_i v_i$ IF ALLOCATED AND 0 OTHERWISE

ORMULATION:

$$\max \sum_i q_i v_i x_i$$

$$\sum_i x_i = K$$

$$x_i \in \{0,1\} \quad \forall i$$

K IS THE NUMBER OF SLOTS

OLUTION: ANY ALLOCATION OF THE FIRST k ADS IN TERMS OF $q_i v_i$ IS OPTIMAL

OMPLEXITY: $\left(O(|A|)\right)$ $O(n)$ WHERE $n = |A|$

PAYMENTS: VCG-LIKE : p_i = [redacted]

$$= SW^*(\hat{v}_{-i}) - SW^*_{-i}(\hat{v})$$

PAY-PER-CLICK: p_i IS DIVIDED BY q_i GIVEN THAT AN ADV PAYS ONLY ONCE CLICKED

EXAMPLE:

a_1 $q_1 \cdot v_1 = 3$

a_2 $q_2 \cdot v_2 = 4$

a_3 $q_3 v_3 = 5$

a_4 $q_4 v_4 = 4.5$

$k=2 \Rightarrow$ OPTIMAL ALLOCATION

$\langle a_3, a_4 \rangle$

$p_3 = 8.5 - 4.5 = 4$

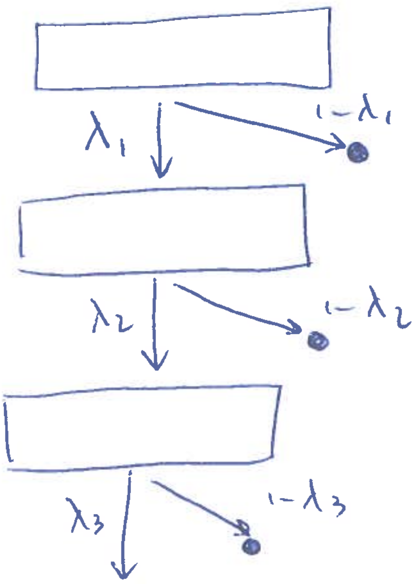
$p_4 = 9 - 5 = 4$

POSITION DEPENDENT EXTERNALITY

MODEL: THE USER BEHAVES IN MARKOVIAN FASHION WITH A GIVEN PROBABILITY TO OBSERVE THE NEXT SLOT AND THE REMAINING PROBABILITY TO STOP OBSERVING THE SLOT

$\lambda_i \in [0,1]$

$\Lambda_i \in [0,1]$



$$\Lambda_J = \prod_{I=1}^{J-1} \lambda_I$$

$\Lambda_1 = 1$

$\Lambda_2 = \lambda_1$

$\Lambda_3 = \lambda_1 \lambda_2$

$\Lambda_4 = \lambda_1 \lambda_2 \lambda_3$

$$\Lambda_1 \geq \Lambda_2 \geq \Lambda_3 \geq \dots \Lambda_k$$

AN AD IN POSITION j HAS A DISCOUNT OF Δ_j

XAMPLE:

a_1
a_2
a_3
a_4
a_5

$$q_1 v_1 + q_2 v_2 \Delta_2 + q_3 v_3 \Delta_3 + q_4 v_4 \Delta_4 + \dots$$

ORMULATION:

$$\text{MAX} \sum_i \sum_j q_i v_i x_{ij} \Delta_j$$

$$\sum_j x_{ij} \le 1 \qquad \forall i$$

$$\sum_i x_{ij} \le 1 \qquad \forall j$$

$$x_{ij} \in \{0,1\} \qquad \forall i, j$$

OLUTION: THE ALLOCATION IN WHICH THE h-th ad (ONCE RANKED IN INCREASING ORDER IN $q_i v_i$) IS IN THE h-th slot IS OPTIMAL

OMPLEXITY: $O(n \lg(k))$

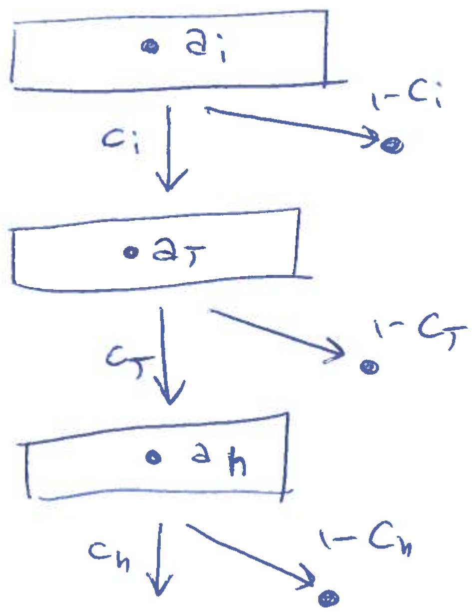
AYMENTS: VCG-LIKE $\quad p_i = SW^*(v_{-i}) - SW^*_{-i}(v)$

BSERVATION: IN THIS CASE, THE PAYMENT IS NOT THE SECOND-HIGHEST DUE TO Δ_i

OMMENTS: MANY SEARCH ENGINES ADOPT THE GSP THAT IS NOT INCENTIVE COMPATIBLE (TRUTHFUL)

• AD DEPENDENT EXTERNALITY

MODEL: THE USER BEHAVES IN MARKOVIAN FASHION AS IN THE CASE OF POSITION EXTERNALITIES, BUT THE PROBABILITIES DEPEND ON THE ALLOCATED ADS



$$C_z = \prod_{a_i \text{ precedes } a_z} c_i$$

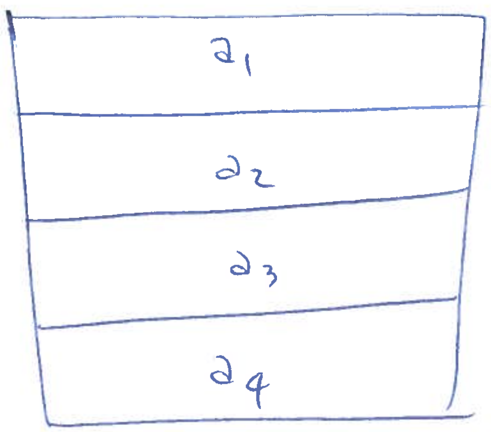
$$C_i = 1$$

$$C_j = c_i$$

$$C_h = c_i \cdot c_j$$

C_i discounts the utilities of the AGENTS

EXAMPLE:



$$q_1 v_1 + c_1 q_2 v_2 + c_1 c_2 q_3 v_3 + c_1 c_2 c_3 q_4 v_4$$

BSERVATION: THERE IS NOT A SIMPLE LINEAR (INTEGER) FORMULATION AS IN THE PREVIOUS CASE

UESTION: IS IT POSSIBLE TO FIND THE OPTIMAL ALLOCATION IN POLYNOMIAL TIME?

ANSWER: YES IT IS, BY USING DYNAMIC PROGRAMMING

ALGORITHM: IT CAN BE SHOWN THAT, GIVEN THE SET OF ADS IN THE OPTIMAL ALLOCATION, THEY ARE RANKED ACCORDING TO

$$\frac{q_i v_i}{1-c_i}$$

THAT IS, ONCE ORDERED THE ADS ACCORDING TO THEIR POSITION IN THE OPTIMAL ALLOCATION

$$\frac{q_1 v_1}{1-c_1} \geq \frac{q_2 v_2}{1-c_2} \geq \frac{q_3 v_3}{1-c_3} \geq \cdots$$

THE ALGORITHM RANKS ADS IN DECREASING ORDER IN $\frac{q_i v_i}{1-c_i}$ AND THEN IT APPLIES THE FOLLOWING PROCEDURE

[TABLE]

	S_1	S_2	S_3	S_4	...	S_k
a_1					...	
a_2					...	
a_3					...	
a_4					...	
a_5					...	
a_6					...	
a_7					...	
$\vdots$	...		...	...	...	...
a_n					...	

1) WE FILL THE TABLE FROM THE LAST ROW AND WE GO UP UNTIL THE FIRST ROW

EACH CELL OF THE TABLE CONTAINS THE OPTIMAL VALUE OF THE ALLOCATION FROM SUCH SLOT ON

1.1 THE LAST ROW IS A CONSTANT

	s_1	s_2	s_3	....	s_k
a_n	$v_n \cdot q_n$	$v_n \cdot q_n$	$v_n \cdot q_n$		$v_n \cdot q_n$

1.2

	s_1	s_2	s_3	....	s_k
a_i					
a_{i+1}					

$$\text{MAX}\left\{ q_i v_i + c_i \cdot T(a_{i+1}, s_2),\ T(a_{i+1}, s_1) \right\}$$

$$\text{MAX}\left\{ q_i v_i,\ T(a_{i+1}, s_k) \right\}$$

2) IN THE FIRST ELEMENT OF THE TABLE WE HAVE THE VALUE OF THE OPTIMAL ALLOCATION

THE COMPLEXITY IS $O(\underbrace{n \cdot k}_{\text{FINDING THE OPTIMAL ALLOCATION GIVEN THE ORDERING}} + \underbrace{n \lg(n)}_{\text{SORTING THE ADS}})$

EXAMPLE:

	$q_i r_i$	c_i	$\frac{q_i r_i}{1-c_i}$
a_1	5	0.2	6.25
a_2	3	0.8	15
a_3	2	0.9	20
a_4	2	0.7	$6.\bar{6}$

$\Rightarrow$ a_3, a_2, a_4, a_1

$K = 2$

	S_1	S_2
a_3	0	5
a_2	0	5
a_4	0	5
a_1	5	5

$\text{MAX}\{2 + 0.7 \cdot 5, 5\} = 5.5$

$\text{MAX}\{3 + 0.8 \cdot 5, 5.5\} = 7$

$\text{MAX}\{2 + 0.9 \cdot 5, 7\} = 7$

OPTIMAL ALLOCATION $= \langle a_2, a_1 \rangle$

$K = 3$

	S_1	S_2	S_3
a_3	7.67	7	5
a_2	7.4	7	5
a_4	5.5	5.5	5
a_1	5	5	5

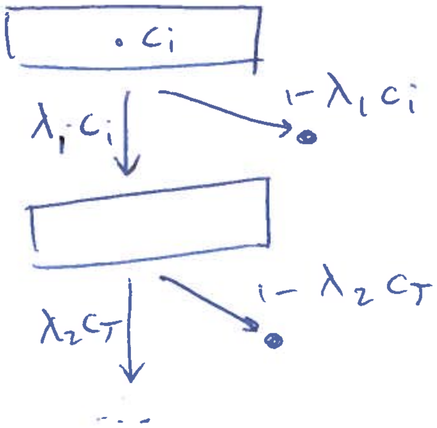
OPTIMAL ALLOCATION $= \langle a_3, a_2, a_1 \rangle$

OBVIOUSLY IF $k = 4$ THE OPTIMAL ALLOCATION IS:

$\langle a_3, a_2, a_4, a_1 \rangle$

• GENERAL EXTERNALITIES

MODEL:



THE EXTERNALITIES DEPEND ON BOTH POSITION AND AD

NJECTURE: IT IS COMMONLY BELIEVED THAT THE PROBLEM OF FINDING THE OPTIMAL ALLOCATION IS HARD (NP-HARD) AND THEREFORE THAT THERE IS NO POLYNOMIAL TIME ALGORITHM. (NO PROOF IS KNOWN)

BSERVATION: THE HARDNESS OF THE ALLOCATION PROBLEM DOES NOT ALLOW THE RESORT TO GROVES MECHANISMS

BSERVATION: IN ORDER TO HAVE AN ECONOMICALLY STABLE MECHANISM WE NEED AN ALLOCATION ALGORITHM THAT IS MONOTONIC

LGORITHM:
(1) SELECT AN ARBITRARY ORDERING OVER THE ADS
(2) FIND THE BEST ALLOCATION SUBJECT TO SUCH AN ORDERING BY USING DYNAMIC PROGRAMMING

ROPERTIES:
(1) THE ALGORITHM HAS A POLYNOMIAL-TIME COMPLEXITY $O(n \cdot k + n \cdot \log(n))$
(2) THE ALGORITHM IS MONOTONIC BECAUSE IT IS MAXIMAL-IN-ITS-RANGE (FINDING THE BEST ALLOCATION AMONG THOSE OF A FIXED RANGE)
(3) THE ALGORITHM PROVIDES AN APPROXIMATION OF $1/2$

SOME DETAILS ON POINT 3

- CALL OPT THE VALUE OF THE OPTIMAL ALLOCATION
- CALL APX THE VALUE RETURNED BY THE ALGORITHM

FOR EVERY POSSIBLE INSTANCE $\frac{APX}{OPT} \geq \frac{1}{2}$