# An Incentive-Compatible Multi-Armed Bandit Mechanism

Rica Gonen*
Yahoo! Research Labs
701 First Street
Sunnyvale, CA 94089
gonenr@yahoo-inc.com

Elan Pavlov
Media Lab
MIT
Cambridge MA, 02149
elan@mit.edu

## ABSTRACT

This paper presents a truthful sponsored search auction based on an incentive-compatible multi-armed bandit mechanism. The mechanism described combines several desirable traits. The mechanism gives advertisers the incentive to report their true bid, learns the click-through rate for advertisements, allows for slots with different quality, and loses the minimum welfare during the sampling process.

The underlying generalization of the multi-armed bandit mechanism addresses the interplay between exploration and exploitation in an online setting that is truthful in high probability while allowing for slots of different quality. As the mechanism progresses the algorithm more closely approximates the hidden variables (click-though rates) in order to allocate advertising slots to the best advertisements. The resulting mechanism obtains the optimal welfare apart from a tightly bounded loss of welfare caused by the bandit sampling process.

Of independent interest, in the field of economics it has long been recognized that preference elicitation is difficult to achieve, mainly as people are unaware of how much happiness a particular good will bring to them. In this paper we alleviate this problem somewhat by introducing a valuation-discovery process to the mechanism which results in a preference-elicitation mechanism for advertisers and search engines.

## 1. INTRODUCTION

The central goal of the field of mechanism design is to define allocations and payments that maximize the welfare of the participants. The central paradigm which aids the mechanism designer in achieving this goal is *preference elicitation* which essentially means finding incentives (via payment rules) that motivate the participants to honestly report their valuations for any possible allocation.

In the field of economics e.g., [4, 3] it has long been recognized that preference elicitation is difficult to achieve. This is not merely (or even largely) due to people's reluctance to disclose their valuations but to a large extent stems from people's ignorance of their own preferences. Although issues such as *loss aversion* [17, 18], contribute to the problem, the main difficulty is that people are unaware of how much happiness a particular good will bring them.

Our goal is to alleviate this problem somewhat by using a process of discovery to allow people to learn their value for a good while interacting with the mechanism. In this paper we focus on a particular online game, the one created by Overture/Yahoo! and modified by Google to auction keywords. Keyword auctions are Yahoo! and Google's method of allocating advertising slots to potential advertisers. Each advertiser can bid on a set of *keywords*. In the current method, for each keyword the advertisers are ranked by multiplying the valuation that each advertiser declared for the keyword times the expected *click-through rate (CTR)* of that advertiser.

An additional unique challenge of the keyword auction is the allocation of multiple copies of the good with different quality at each stage [10]. In a companion paper [15] we focus on the more general problem of preference elicitation for general repeated games with different entry and exit times (although with a single good allocated in each time period) as well as some results for the single-shot game.

In the keyword auction assuming that an advertiser gains no advantage if his ad is not clicked on[1], it is seen that the value to the advertiser is the *value per click* × CTR. The main problem is that the CTR is unknown to the advertiser (as well as initially to the search engine) and hence poses the problem of an allocation with unknown valuations.

In practice, Google and Yahoo! allocate new advertisers a number of impressions which suffice to determine (within an error bound) the CTR. Since in many cases such sampling is costly and can yield a large loss of welfare, Google and Yahoo! utilize some heuristics to estimate the click-through rate of a new advertiser. We are unaware of any analysis bounding the loss of welfare that can result from using these heuristics. Furthermore, it creates an incentive for users to repeatedly become "new" advertisers. Even if this can be prevented, e.g., by using payment information, the incentive to bid in a large number of auctions with very low value remains. In a companion paper [16] we bound the utility an advertiser can gain by bidding under multiple false identities for low value keywords.

A further peculiarity of the keyword auction is that in each round a number of slots are allocated to different advertisers. Since these slots are of different quality [10] the auction

*confidentially submitted for presentation in WWW 2007 May 8th.

[1] In practice some advertisers are interested in rasing visibility which poses a problem to current methods of charging for advertisements. This problem is beyond the scope of this paper.

*WWW 2007*, May 8–12, 2007, Banff, Alberta, Canada.
.

mechanism must take into account which slot is allocated to which user. To deal with this problem we slightly generalize the commonly accepted assumption that the quality of slots is independent of the advertiser assigned to the slot.

In the Google keyword auction, game prices are determined for the allocated advertisers in an attempt to create a truthful mechanism via e.g., a second-price auction. In practice (e.g., [10], and [14])'s attempt to create a truthful mechanism via second-price auction does not result in a truthful mechanism and the mechanism used for determining prices is *not* truthful. So in addition to the challenge of revealing the advertisers' valuations while comparing the quality of different slots, our algorithm faces the challenge of designing a truthful mechanism while charging differently for different slots. Our main tool for charging for different slots is a generalization of the ladder prices due to [2].

Models of imperfect and symmetric information for prices have been extensively studied recently, e.g., [7, 8]. We choose to use the classic *multi-armed bandit (MAB)* as our main technical tool to learn advertisers' valuations. The multi-armed bandit is a well studied problem (e.g., [21, 5]) which deals with the balancing of exploration and exploitation in online problems with multiple possible solutions. In the simplest version of the MAB problem a user must choose at each stage (the number of stages is known in advance) a single bandit/arm. This bandit will yield a reward which depends on some *hidden* distribution. The user must then choose whether to exploit the currently best known distribution or to attempt to gather more information on a distribution that currently appears suboptimal. The MAB is known to be solvable via the Gittins [13] index and there are solutions that approximate the optimal expected payoff. We choose to generalize the MAB solution in [11] due to its simplicity and optimal sampling complexity. Our solution retains the sample complexity of [11] and hence is sample complexity optimal. The MAB has been recently studied in a more general setting by [9] but using a weaker notion of truthfulness.

Although the MAB has been extensively studied it has generally been studied in the context of a single user choosing from non-strategic arms [19] even when studied in the context of slot auctions [20], however the important question of a truthful mechanism for strategic arms remains open. Furthermore, allowing different slots with varying quality adds several technical difficulties to our solution which do not exist in previous work. Obviously, in the context of an online auction for keywords the arms/advertisers will act as strategic utility-maximizing agents. Our goal is therefore to design a truthful mechanism for the strategic case. By defining the keyword problem as an instance of a truthful mechanism for MAB we can approximate the optimal payoff for the MAB and hence approximate the optimal welfare for the auction. As our **M***ulti-***A***rmed truth***F***ul band***I***t* **A***uction (MAFIA)* algorithm achieves optimal sampling complexity we are also able to bound the welfare loss we encounter from the sampling process. This bound is shown to be tight for any sampling keywords algorithm.

When looking at randomized algorithms for mechanism design we must be careful about which notion of truthfulness we use. Since we are sampling click through by users, for any finite time horizon $T$, there is a finite probability that the sampling is done incorrectly and hence will influence our truthfulness. We therefore use the notion of *truthfulness with high probability* due to [1] for finite time horizons. Furthermore, our algorithm is also truthful in expectation.

**Organization:** The rest of the paper is organized as follows: In section 2 we present our model, the bandit problem and define necessary assumptions. We give some intuition by looking at the case of a single slot in section 3. Section 4 presents the MAFIA algorithm which its properties of truthfulness, welfare maximization, sampling complexity and the bounded welfare lost by sampling are analyzed in section 5. We conclude and discuss future extensions in section 6.

## 2. THE MODEL

In our model $N$ risk neutral, utility maximizing advertisers bid for advertising slots based on a keyword. In this paper we focus on the bidding process for a single keyword, as multiple keywords are analogous. We therefore suppose w.l.o.g. that the keywords appears at every time $t$. Whenever that keyword appears in the search at, $K_t$[2] slots of advertisements appear in the results. Each advertiser $i$ has a private value for each click through which we denote by $v_i$. This value is independent of the slot the ad originally appeared in.

We assume that all of the advertisers are present in the system throughout the entire running of the algorithm and that there are no budget constraints. These assumptions are relaxed in [15]. The algorithm runs in time rounds starting at $t = 1$ and ending at $t = T$. By setting $t = T$ meaning a finite-time horizon we assume a harder setting than an infinite-time horizon as our algorithm is a sampling algorithm. Consequently any proof that applies for the finite-time horizon also applies for the infinite-time horizon. Our model studies a one-shot incomplete information game meaning that advertisers do not change their valuations in the different time periods of the algorithm and can not learn about each other's valuations.

We also assume that the "quality" of each slot $j$ (which is essentially the probability of a click though if an advertisement appears in slot $j$) is monotonically decreasing and is independent of the advertisers, i.e., the first slot has the highest probability to be clicked on regardless of the ad presented in it. The second slot has the second highest probability to be clicked on etc.

Since different slots are of different quality if (for example) advertiser $a$ is presented in the first slot and gets a click and advertiser $b$ is presented in the second slot and does not gets a click, we can not just simply update advertiser's $a$ click through rate with an extra click and reduce advertiser $b$'s click through rate as we don't know what clicks would have happened if advertiser $b$ was presented in the first slot. In order to be able to compare click through rates across slots we define normalization constants between slots $j-1$ and $j$ for all $K \geq j > 1$. Denote by $r_j$ a click in slot $j$ and $\neg r_j$ no click in slot $j$. There are four cases:

- $\beta_j^1$ - the probability that an advertisement would have been clicked in slot $j$ (if we had shown it in slot $j$) given that it was clicked in slot $j-1$, i.e., $\beta_j^1 = Pr[r_j | r_{j-1}]$
- $\beta_j^2$ - the probability that an advertisement would have

[2] We assume for the ease of exposition that $K_t = K_{t+1} = K$ for all time period $t$. We also assume without the loss of generality that $K \leq N$, since superfluous slots can remain blank.

been clicked in slot $j$ given that it was *not* clicked in slot $j-1$, i.e., $\beta_j^2 = Pr[r_j|\neg r_{j-1}]$.

- $\tilde{\beta}_j^1$ - the probability that an advertisement would have been clicked in slot $j-1$ given that it was clicked in slot $j$, i.e., $\tilde{\beta}_j^1 = Pr[r_{j-1}|r_j]$.
- $\tilde{\beta}_j^2$ - the probability that an advertisement would have been clicked in slot $j-1$ given that it was *not* clicked in slot $j$ i.e., $\tilde{\beta}_j^2 = Pr[r_{j-1}|\neg r_j]$.

The assumption that click through rate decays monotonically with lower slots by the same factors for each advertiser has been widely assumed in practice and in theory.

We generalize the common assumption of monotonicity and assume that there exists constants that allow us to calculate all of the conditional probabilities both when there is a click through and when there is not. Given the large data sets that search engines have we believe that this assumption is justified in practice.

Each advertiser $i$ has a *click through rate* $\alpha_i$ which is the probability of a click on the advertisement given that it appeared in the first slot[3]. This value is unknown to $i$ as well as to the mechanism. Since $\alpha_i$ is unknown to $i$ and the mechanism we estimate it at each time $t$ and denote the observed probability by $\alpha_i^t$.

Finally, we denote by $\bar{v}_i$ the bid for each click-through stated by advertiser $i$ to the mechanism (which might not be the true value). We also denote by $p_i^t \leq \bar{v}_i$ the price which advertiser $i$ is charged at time $t$ by the mechanism. We assume that advertisers have quasi-linear utility function and as such advertiser $i$ placed at slot $j$ at time $t$ obtains an expected utility of $\beta_2^1 \cdot ... \cdot \beta_j^1 \cdot \alpha_i^t \cdot (v_i - p_i^t)$ per impression at time $t$.

## 2.1 The Bandit Problem

The multi-armed bandit problem, originally described by Robbins [21], is a statistical decision model of an agent trying to optimize his decisions while improving his information at the same time. In the multi-arm bandit problem, the gambler has to decide which arm of $K$ different slot machines to play in a sequence of trials so as to maximize his reward.

The bandit problem is best formulated as an infinite horizon Markov decision problem in discrete time with time index $t = 0, 1, ...$ At each time $t$ the decision maker chooses amongst $N$ arms and we denote this choice by $a_t \in \{1, ..., N\}$. If $a_t = i$, a random payoff $x_t^i$ is realized and we denote the associated random variable by $X_t^i$. In our slot auction $x_t^i = \alpha_i^t \cdot v_i$ where the click through rate $\alpha_i^t$ is the random payoff element of the problem while the value $v_i$ is a constant, hence the total payoff for arm $i$ is $v_i \times \alpha_i^t$. The state variable of the Markovian decision problem is given by $s_t$ where in our slot auction a vector of all allocated advertisers click-through-rate at time $t$, $\alpha_i^t$ and 0 if $i$ is not allocated a slot in time $t$. The distribution of $x_t^i$ is $F^i(\cdot; s_t)$. The state transition function $\phi$ depends on the choice of the arm and the realized payoff: $s_{t+1} = \phi(x_t^i; s_t)$. Let $S_t$ denote the set of all possible states in period $t$. A feasible Markov policy $a = \{a_t\}_{t=0}^{\infty}$ selects an available alternative for each conceivable state $s_t$, i.e., $a_t : S_t \rightarrow \{1, ..., N\}$. Payoffs are evaluated according to the discounted expected payoff criterion where the discount factor $\delta$ satisfies $0 \leq \delta < 1$. The motivation for assuming a discount factor is that the seller of the slot auction prefers payment sooner rather than later. The payoff from each $i$ depends only on outcomes of periods with $a_t = i$. In other words, we can decompose the state variable $s_t$ into $N$ components $(s_t^1, ..., s_t^N)$ such that for all $i$: $s_{t+1}^i = s_t^i$ if $a_t \neq i$, $s_{t+1}^i = \phi(s_t^i, x_t)$ if $a_t = i$, and $F^i(\cdot, s_t) = F^i(\cdot; s_t^i)$.

# 3. ILLUSTRATION OF OUR PROTOCOL FOR THE SINGLE SLOT CASE

We illustrate the main idea behind our protocol for the simple case when there is a single slot available at any given time. Our algorithm starts with a set $S = N$ of all advertisers and no knowledge of their click through rate. At each time period $t$ and for each advertiser $i \in S$ we have an estimate of $i$'s click through rate $\alpha_t^i$ as well as an estimate of how accurate our estimation is, i.e., a (probabilistic) bound on $|\alpha_t^i - \alpha_i|$ which depends on the time period $t$ that the algorithm sampled. We will denote this bound by $\gamma^l$ where $l$ is the stage number (details below). Advertisers are removed from the set $S$ and are not considered for sampling once the algorithm learned that their estimated click through rate is less than the maximum click through rate (even when adjusting for sampling errors).

We divide our protocol into multiple stages. Each stage consists of a (variable) number of rounds. When a stage starts we consider all advertisers at the set $S$. Obviously, this set has an $i$ s.t. $v_i * \alpha_t^i$ is maximal. Suppose w.l.o.g. that the maximal element is the first element we consider in $S$. If we would merely choose to do an exploitation that is we could just allocate the slot to the first advertiser. However, there are other possible advertisers that are worthy of consideration. These are the advertisers $j$ s.t. $v_i(\alpha_t^i - \gamma^l) < v_j(\alpha_t^j - \gamma^l)$. In this case the inaccuracy of $i, j$ overlap. Therefore the algorithm allocates to this stage a sufficient number of rounds to sample all of these possible advertisers (for simplicity we assume w.l.o.g. that the time finishes, i.e., $t = T$, only when starting a new stage. If there is insufficient time to finish a stage we simply don't sample in that stage.).

This algorithm (due to [11]) works (in a PAC sense) if the players are non-strategic. Of course, if the players are strategic we have to motivate them to give the correct values $v_i$. Since all of the advertisers arrive and depart at the same time and we only allocate a single slot at any given time we can set the price for any stage to be defined as the critical value at that stage to be sampled (i.e. given sampled $j$ s.t. $v_j * \alpha_j^t$ is maximal from among the advertisers not sampled, the price for player $i$ is $\frac{v_j * \alpha_j^t}{\alpha_t^i}$.)

Our single slot protocol is in table 1.

We now proceed to formally define our algorithm for the general case.

# 4. MULTI-ARMED TRUTHFUL BANDIT AUCTION(MAFIA)

When turning to the general case we encounter several new problems. The first problem is that we now have multiple slots. In a simplified setting where the slots are of the same quality, [20] presented a multi armed bandit mechanism where values are known and no assumption of adver-

[3] The normalization constants enable us to use the first slot as a baseline

```
THE SINGLE SLOT ILLUSTRATIVE ALGORITHM

1. All advertisers $i$ report their value $\bar{v}_i$.
2. Set the time to be $t = 1$ and the set of advertisers $S = N$
3. Set initial click though rates for each advertiser - $i : x_1^i = 0$
4. Sample every advertiser $i \in S$ once.
5. if $i$ was clicked on charge advertiser $i$ $p_i^t = \frac{\alpha_j^t}{\alpha_i^t} v_j$ where $x_t^j$ is the second highest payoff.
6. If we are at a new stage $l$ define confidence parameters - Let $\gamma^l = \sqrt{\frac{\log(cnl^2/\delta)}{l}} \cdot$, let $x_t^{\max} = \max_{i \in S} x_t^i$.
7. Define the set of advertisers which we no longer need to sample: for every $i \in S$ such that $x_t^{\max} - x_t^i \geq 2\gamma^l$ set $S = S \backslash i$.
8. t=t+1
9. If $|S| > 1$ (there are still too many possibilities) then
   Go to 4
10. from $\tau = t$ to $T$ allocate advertiser $i$ to the slot
11. for every $t$ if $i$ was clicked on charge $i$ $p_i^t = \frac{\alpha_j^t}{\alpha_i^t} v_j$
```

**Table 1: The illustrative algorithm simple case with a single slot**

tisers strategic behavior is taken. However, when slots are of different quality we must take care that during the sampling procedure we allocate "better" slots to "better" advertisers. Of course, payments need to be taken for such preferential allocations to work.

Our main protocol is in table 2. The protocol samples each advertiser $i$ in turn until there is a sufficient gap between the observed payoffs of the $K$ highest advertisers and advertiser $i$ such that with sufficient probability the $i$'th advertiser is not one of the advertisers we want to retain. The algorithm removes all of the advertisers with a sufficiently large gap and continues to sample the remaining advertisers as long as there is not a large enough gap between the best advertisers and the rest of the advertisers to remove them using the procedure in table 4. The main protocol utilizes several sub procedures. Table 3 is used to normalize the click-through probabilities of different slots so that they can be compared to the same baseline slot. Prices are set in table 6 to motivate the advertisers to honestly report their bids to the mechanism. The prices are computed following the truthful ladder scheme of [2].

When the most desired $K$ advertisers remain, each advertiser needs to be allocated the proper slot; meaning the most desired advertiser in first slot, the second most desired in second slot etc. This is done in table 5 simply by ensuring that there is a sufficient gap between two consecutive advertisers' observed probabilities.

```
THE MAIN ALGORITHM

1. All advertisers $i$ report their value $\bar{v}_i$
2. Set time $t = 1$, $l = 1$ and the set of advertisers $S = N$
3. Set initial click though rates for each $i$, $x_1^i = 0$
4. Randomly sample every advertiser $i \in S$ once.
   (a) for every time $t$ in stage $l$:
   (b) We normalize click-through rates to allow us to compare them: normalize-click-through-rate(input:$S$,output:$\alpha_i^t$ for all $i \in S$).
   (c) t=t+1
5. l=l+1
6. Define confidence parameter: $\gamma^l = \sqrt{\frac{\log(cnl^2/\delta)}{l} \cdot \frac{1}{\max\{\beta_2^1 \cdot \ldots \cdot \beta_K^1, \tilde{\beta}_K^2 \cdot \tilde{\beta}_{K-1}^1 \cdot \ldots \cdot \tilde{\beta}_2^1\}} \cdot \frac{1}{K}}$ and $x_t^{\max_K}$ be the $K$th highest payoff $x_t^i$ of $i \in S$.
7. Discard obviously suboptimal advertisers: for each $i \in S$ such that $x_t^{\max_K} - x_t^i \geq 2\gamma^l$ set $S = S \backslash i$.
8. If $|S| > K$ (there are still too many possibilities) then allocate-slots-for-sampling(input:$S$, for all $i \in S$ $x_i^t$). Go to 5
9. Decide which slots are allocated to which advertisers: match-$K$-slots(input:$t$,$S$,output:for all $i \in S$ slot $j \in K$).
10. From $\tau = t$ to $T$ and for all $i \in S$ allocate advertiser $i$ to slot $j$
11. If $i$ got a click charge price of $p_i^t$:
    compute-ladder-price(input:$i$,$j x_t^{z_{j+1}}, \ldots, x_t^{z_K}$, output:$p_i^t$), where $z_j$ the advertiser that was allocated slot $j$.
```

**Table 2: The main algorithm used for the MAB sampling problem**

```
NORMALIZE-CLICK-THROUGH-RATE
(INPUT:$S$,OUTPUT:$\alpha_i^t$ $\forall$ $i \in S$)

1. For every $i \in S$ that was given slot $j$:
   if $i$ got a click normalize the click by $\tilde{\beta}_j^1 \cdot ... \cdot \tilde{\beta}_2^1$
   else normalize the click by $\tilde{\beta}_j^2 \cdot \tilde{\beta}_{j-1}^1 ... \cdot \tilde{\beta}_2^1$
   Update $\alpha_i^t$ (and $x_t^i$) accordingly
2. if $i$ got a click charge price of $p_i^t$:
   compute-ladder-price(input:$i$,$jx_t^{z_{j+1}}, ..., x_t^{z_K}$, output:$p_i^t$), where $z_j$ the advertiser that was allocated slot $j$.
```

**Table 3: Normalizing click-through-rate to a baseline using constants**

```
ALLOCATE-SLOTS-FOR-SAMPLING
(INPUT:$S$, $\forall$ $i \in S$ $x_i^t$)

1. Order the payoffs $x_i^t$ of $i \in S$ and denote the $d$'th high payoff by $x_{i_d}^t$
2. Sample every advertiser $i \in S$ for every time $t$ in stage $l$ in the following order:
   for every slot $j \in K$ chose an advertiser at random without repetition of $i_{(j-1)(|S|/K)+1}$ to $i_{j|S|/K}$.
   (a) normalize-click-through-rate(input:$S$,output:$\alpha_i^t$ for all $i \in S$)
   (b) t=t+1
```

**Table 4: Choosing which advertisers get which slots during sampling**

# 5. THE MAFIA ANALYSIS

This section analyzes the properties of the MAFIA algorithm. The properties we focus on are truthfulness, welfare maximization, welfare lost by sampling and sampling complexity.

## 5.1 Truthfulness

To show truthfulness we will prove that assuming that the algorithm correctly finds the best advertisers, every advertiser $i$ gains his maximum expected utility $\beta_2^1 \cdot ... \cdot \beta_j^1 \cdot \alpha_i^t \cdot (v_i - p_i^t)$ for every time $t$ when reporting his true value, i.e., $\bar{v}_i = v_i$ and getting the according allocation of slot $j$. If truthfulness holds for every time $t$ it follows that truthfulness holds over the slot auction as whole since no advertiser can gain by losing on utility in some time periods in order to gain utility in some other time period[4]. Since we will show that the auction has arbitrary high probability of finding the optimal allocation this will suffice.

```
MATCH-$K$-SLOTS
(INPUT:$t$,$S$,OUTPUT:$\forall$ $i \in S$ $slot$ $j \in K$)

For $z = 1$ to $K - 1$:
  1. Sample all advertisers $i \in S$
     (a) For every time $t$ in stage $l$:
     (b) normalize-click-through-rate(input:$S$,output:$\alpha_i^t$ for all $i \in S$)
     (c) If advertiser $i'$ that is allocated slot $K - z + 2$ got a click charge $p_{i'}^t$
         compute-ladder-price (input:$i'$,$K - z + 1$,$x_t^{z_{K-z+2}}, ..., x_t^{z_K}$,output:$p_i^t$).
     (d) t=t+1
  2. Use confidence parameter - $\gamma^l = \sqrt{\frac{\log(cnl^2/\delta)}{l} \cdot \frac{1}{\max\{\beta_2^1 \cdot ... \cdot \beta_K^1, \tilde{\beta}_K^2 \cdot \tilde{\beta}_{K-1}^1 \cdot ... \cdot \tilde{\beta}_2^1\}} \cdot \frac{1}{K-z}}$
  3. for every advertiser $i \in S$ such that $x_t^{\max_{K-z}} - x_t^i \geq 2\gamma^l$ set $S = S \backslash i$.
  4. allocate the removed $i$ in slot $K - z + 1$.
```

**Table 5: Matching advertisers with slots after sampling is done**

```
COMPUTE-LADDER-PRICE
(INPUT:$i$,$j$,$x_t^{z_{j+1}}, ..., x_t^{z_K}$,OUTPUT:$p_i^t$)

1. for $f = j + 1$ to $K$ $\alpha_{z_f}^t = x_t^{z_f} / v_{z_f}$
2. $p_i^t = \sum_{f=j}^{K} (\frac{\alpha_i^t \cdot \beta_2^1 \cdot ... \cdot \beta_f^1 - \alpha_i^t \cdot \beta_2^1 \cdot ... \cdot \beta_{f+1}^1}{\alpha_i^t \cdot \beta_2^1 \cdot ... \cdot \beta_j^1}) \frac{\alpha_{z_{f+1}}^t}{\alpha_i^t} v_{z_{f+1}}$
```

**Table 6: Calculate Prices**

Since we show that for finite time , the algorithm can only hope to succeed with some probability which is less than 1, we can not hope to show that the mechanism is *always* truthful. However, following [1] we show that the algorithm is truthful w.h.p. which depends on the parameters $\gamma^l$. Since $\gamma^l$ has a constant $c$ which we can use to fine-tune the tradeoff between expected success and expected loss of welfare we can ensure that the probability of success which we denote by $1 - \lambda$ is arbitrarily high. This will then ensure that the probability of an advertiser to gain by lying is bound by some arbitrary low constant which we denote by $\theta$.

To illustrate why the algorithm require truthfulness with high probability and in expectation consider the following scenario: given advertiser $i$ with true value $v_i$ per click and a real click-through-rate $\alpha_i^t$ at time $t$ and assume that at time $t$ the first slot should have been allocated to advertiser $i$ if $i$'s click-through-rate estimated correctly by the algorithm. Now assume that the algorithm estimated a much lower click-through-rate for $i$ at time $t$ $\bar{\alpha}_i^t < \alpha_i^t$ (which can happened with probability $\lambda$) such that it is placed in slot $K$ instead of the first slot. If advertiser $i$ lies such that $\bar{v}_i \cdot \bar{\alpha}_i^t = v_i \cdot \alpha_i^t$ then $i$'s lie actually results in a better utility

[4] It is important to recall that our model studies a one-shot incomplete information game. This means that advertisers do not change their valuations in the different time periods of the algorithm.

for $i$ then reporting his true value. We can insure that this kind of lie is not beneficial for the advertisers ($\theta$ is small) if $\lambda$ is taken to be small enough. The formal proof follows:

Let $\alpha_i^t$ and $\hat{\alpha}_i^t$ be the click-through-rate found by the algorithm at time $t$ with probability $1-\lambda$ and probability $\lambda$ respectively when advertiser $i$ reports $v_i$ and let $\bar{\alpha}_i^t$ and $\hat{\bar{\alpha}}_i^t$ be the click-through-rate found by the algorithm at time $t$ with probability $1-\lambda$ and probability $\lambda$ respectively when advertiser $i$ reports $\bar{v}_i$.

Lemma 1. *Given advertiser $i$ and time $t$, for all $\bar{v}_i \neq v_i$ reported by advertiser $i$ that results in price $\bar{p}_i^t$ it holds that, $(1-\lambda)(\beta_2^1 \cdot \ldots \cdot \beta_j^1 \cdot \alpha_i^t \cdot (v_i - p_i^t)) + \lambda(\beta_2^1 \cdot \ldots \cdot \beta_{j'}^1 \cdot \hat{\alpha}_i^t \cdot (v_i - p_i^t)) \geq (1-\lambda)(\beta_2^1 \cdot \ldots \cdot \beta_{j''}^1 \cdot \bar{\alpha}_i^t \cdot (v_i - \bar{p}_i^t)) + \lambda(\beta_2^1 \cdot \ldots \cdot \beta_{j'''}^1 \cdot \hat{\bar{\alpha}}_i^t \cdot (v_i - \bar{p}_i^t))$ where $j, j', j'', j'''$ are the slots allocated to $i$ at time $t$ with click-through-rates $\alpha_i^t, \hat{\alpha}_i^t, \bar{\alpha}_i^t, \hat{\bar{\alpha}}_i^t$.*

Proof. We divide the proof into several claims.

- The algorithm is truthful in high probability, furthermore for finite time $T$ the mechanism is truthful for sufficiently small $c$ (large $\gamma^l$).
- The mechanism is truthful in expectation.
- The first sampling of each advertiser is truthful since it does not depend on the declaration.
- The sampling until converging on the final set of advertisers is truthful.
- The allocation of different advertisers to different slots during the running of the algorithm is truthful.
- The final matching between the $K$ slots and the $K$ highest estimated advertisers is truthful.

Let $u_i^1 = (\beta_2^1 \cdot \ldots \cdot \beta_j^1 \cdot \alpha_i^t \cdot (v_i - p_i^t))$, $u_i^2 = (\beta_2^1 \cdot \ldots \cdot \beta_{j''}^1 \cdot \bar{\alpha}_i^t \cdot (v_i - \bar{p}_i^t))$, $u_i^3 = (\beta_2^1 \cdot \ldots \cdot \beta_{j'''}^1 \cdot \hat{\bar{\alpha}}_i^t \cdot (v_i - \bar{p}_i^t))$, and $u_1^4 = (\beta_2^1 \cdot \ldots \cdot \beta_{j'}^1 \cdot \hat{\alpha}_i^t \cdot (v_i - p_i^t))$

Claim 1. *If $\lambda \leq \min_{i \in N} \left\{ \frac{u_i^1 - u_i^2}{u_i^1 - u_i^2 + \max_{i \in N}\{u_i^3 - u_i^4\}} \right\}$ then for all advertisers their optimal strategy assumes that the optimal allocation is found,*

Proof. If we set $\lambda$ to be small enough meaning $\lambda \leq \min_{i \in N} \left\{ \frac{u_i^1 - u_i^2}{u_i^1 - u_i^2 + \max_{i \in N}\{u_i^3 - u_i^4\}} \right\}$ then the probability of the algorithm not finding the optimal welfare at some time period $\tau$ will result in an arbitrarily low probability of an advertiser being able to gain by lying and hence w.h.p. the advertiser will tell the truth. Formally, since $\theta \leq \lambda$ then as $\lambda_{c \to \infty} \to 0$ then $\theta \to 0$

□

It is now shown that the mechanism is truthful in expectation.

Lemma 2. *The mechanism is truthful in expectation.*

The proof is omitted and appears in the full paper.

Following claim 1 we assume in the next claims that the algorithm finds the optimal welfare. So fixing the advertiser $i$ and time $t = \tau$, there are two possible lies that $i$ can make. $i$ can either increase or decrease his value. We will show that both of these possible lies yield a negative change in $i$'s utility,

The initial stage is not impacted by advertiser $i$ changing his value in any direction as advertisers are placed randomly and all start with an initial payoff per click of $x_t^i = 0$.

Claim 2. *While sampling until discovering the final $K$ advertisers (lines 6-7 in the main algorithm) advertiser $i$ can not improve his utility by lying.*

Proof. There are two possibilities to consider. The first case is when $i$ decreases his value. By decreasing his value $\bar{x}_t^i$, $t < \tau$ maybe such that $\bar{x}_t^i \leq x_t^{\max_K} - 2\gamma^l$ and advertiser $i$ will be discarded before time $\tau$. It follows that in this case advertiser $i$'s utility is 0 for lying but may be better when telling the truth as $x_t^i$ for $t \leq \tau$ maybe greater than $x_t^{\max_K} - 2\gamma^l$ and advertiser $i$ would not be removed out of the auction in such an early stage.

The second case is when $i$ increases his value. In this case by increasing his value $\bar{x}_t^i$, for $t \leq \tau$ maybe such that $\bar{x}_t^i > x_t^{\max_K} - 2\gamma^l$ and advertiser $i$ will not be removed out of the auction before time $\tau$. If at any time $t \leq \tau$ $x_t^i \leq x_t^{\max_K} - 2\gamma^l$ then advertiser $i$ should have been removed out of the auction but was left in the auction because of his increased value report. To prove our truthfulness claim we need to show that in this case his utility at time $\tau$ is negative as his true report would have given 0 at that time. Assume that advertiser $i$'s payoff when reporting $\bar{v}_i$, $\bar{x}_i^t$ is ranked between the $(j-1)(|S|/K)+1$ and $j(|S|/k)$ payoffs in allocate-slots-for-sampling procedure. Advertiser $i$ is sampled at slot $j$ and $i$'s utility then is

$$\alpha_i^\tau \cdot \beta_2^1 \cdot \ldots \cdot \beta_j^1 (v_i - \textstyle\sum_{f=j}^K \frac{\alpha_i^\tau \cdot \beta_2^1 \cdot \ldots \beta_f^1 - \alpha_i^\tau \cdot \beta_2^1 \cdots \beta_{f+1}^1}{\alpha_i^\tau \cdot \beta_2^1 \cdots \beta_j^1} \cdot \frac{\alpha_{f+1}^\tau}{\alpha_i^\tau} v_{f+1}) =$$

$$\beta_2^1 \cdot \ldots \cdot \beta_j^1 \alpha_i^\tau v_i - \beta_2^1 \cdot \ldots \cdot \beta_j^1 \alpha_{i_{j+1}}^\tau v_{i_{j+1}} (1 + \beta_{j+1}^1) - \qquad (1)$$
$$\beta_2^1 \cdot \ldots \cdot \beta_j^1 \alpha_{i_{j+2}}^\tau v_{i_{j+2}} (\beta_{j+1}^1 + \beta_{j+1}^1 \beta_{j+2}^1) \ldots < 0$$

Inequality (1) is less than 0 as advertiser $i$'s true payoff $\alpha_i^t v_i$ is less than any other payoffs of advertisers that got allocations in the allocate-slots-for-sampling procedure. Therefore $i$ is better off reporting his true value in this case. Note that even if the algorithm found a different click-through-rate for $i$ when lying than when reporting the truth (as $i$ might be discarded after a single sampling) $i$'s utility is still less than 0. That is not surprising given that the algorithm leaves only the advertisers with bounded distance to the real click-through-rate. ∎

□

We now show that the next procedure also affords no opportunity for $i$ to gain by lying.

Claim 3. *For any advertiser $i$ the optimal utility which can be achieved in the allocate-slot-for-sampling procedure is achieved by a truthful declaration.*

Proof. Once again there are two conditions to consider. The first case is if $i$ decreases his value. In this case, if advertiser $i$ "survived" the removal stages his decreased value can still impact his rank of payoff in the procedure allocate-slot-for-sampling. As advertiser $i$ has decreased his value he can only be ranked lower in the payoff list and therefore be sampled in lower slots. Assume without loss of generality that advertiser $i$ was sampled at slot $j$ when telling the truth and sampled at slot $j+1$ when lowering his value. Due to our probability constants the bandit algorithm can normalize the click-through-rates such that it will find the same click-through-rate for $i$ when telling the truth and sampled at slot $j$ and when lying and sampled at slot $j+1$. So as long as advertiser $i$ is allocated a slot when lying he can not impact the algorithm finding of his click-through-rate but he can be charged a different price when allocated slot $j+1$ instead of slot $j$. All that is left to show is that the difference between $i$'s utility when allocated slot $j$ and $i$'s utility when allocated slot $j+1$ is positive, i.e., $\beta_2^1 \cdot ... \cdot \beta_j^1 \cdot \alpha_i^\tau(v_i - p_i^\tau) - \beta_2^1 \cdot ... \cdot \beta_{j+1}^1 \cdot \alpha_i^\tau(v_i - \bar{p}_i^\tau) \geq 0$ where $\bar{p}_i^\tau$ is the price $i$ is charged at slot $j+1$.

$\beta_2^1 \cdot ... \cdot \beta_j^1 \cdot \alpha_i^\tau(v_i - p_i^\tau) - \beta_2^1 \cdot ... \cdot \beta_{j+1}^1 \cdot \alpha_i^\tau(v_i - \bar{p}_i^\tau) = (\beta_2^1 \cdot ... \cdot \beta_j^1 \cdot \alpha_i^\tau - \beta_2^1 \cdot ... \cdot \beta_{j+1}^1 \cdot \alpha_i^\tau)(v_i - \frac{\alpha^\tau_{i_{(j+1)|S|/K}})}{\alpha_i^\tau} \cdot v_{i_{(j+1)|S|/K}})$.

The first term of the equation is positive as $\beta_2^1 \cdot ... \cdot \beta_j^1 \cdot \alpha_i^\tau - \beta_2^1 \cdot ... \cdot \beta_{j+1}^1 \cdot \alpha_i^\tau = \beta_2^1 \cdot ... \cdot \beta_j^1 \cdot \alpha_i^\tau \cdot (1 - \beta_{j+1}^1)$ and $\beta_{j+1}^1$ is a probability constant and therefore less than 1.

The second term of the equation is positive as $i$'s payoff $x_i^\tau = v_i \cdot \alpha_i^\tau$ is ranked higher then the $(j+1)|S|/K$ advertiser's payoff otherwise $i$ would have been matched to the $j$st slot when reporting $v_i$.

The reverse case of $i$ increasing his value follows. We consider advertiser $i$ who was sampled at slot $j$ when reporting the true value $v_i$ and assume with out loss of generality that $i$ was sampled at slot $j-1$ when increasing his reported value $\bar{v}_i$.

Due to our probability constants the bandit algorithm can normalize the click-through-rates such that it will find the same click-through-rate for $i$ when telling the truth and sampled at slot $j$ and when lying and sampled at slot $j-1$. So as long as advertiser $i$ is allocated a slot when lying he can not impact the algorithm finding of his click-through-rate but he can be charged a different price when allocated slot $j-1$ instead of slot $j$. All is left to show is that the difference between $i$'s utility when allocated slot $j$ and $i$'s utility when allocated slot $j-1$ is positive, i.e., $\beta_2^1 \cdot ... \cdot \beta_j^1 \cdot \alpha_i^\tau(v_i - p_i^\tau) - \beta_2^1 \cdot ... \cdot \beta_{j-1}^1 \cdot \alpha_i^\tau(v_i - \bar{p}_i^\tau) \geq 0$ where $\bar{p}_i^\tau$ is the price $i$ is charged at slot $j-1$.

$\beta_2^1 \cdot ... \cdot \beta_j^1 \cdot \alpha_i^\tau(v_i - p_i^\tau) - \beta_2^1 \cdot ... \cdot \beta_{j-1}^1 \cdot \alpha_i^\tau(v_i - \bar{p}_i^\tau) = (\beta_2^1 \cdot ... \cdot \beta_j^1 \cdot \alpha_i^\tau - \beta_2^1 \cdot ... \cdot \beta_{j-1}^1 \cdot \alpha_i^\tau)(v_i - \frac{\alpha^\tau_{i_{(j-1)|S|/K}})}{\alpha_i^\tau} \cdot v_{i_{(j-1)|S|/K}})$.

Both terms are negative which together proves a positive difference in utility for $i$ in reporting the truth. The first term of the equation is negative as $\beta_2^1 \cdot ... \cdot \beta_j^1 \cdot \alpha_i^\tau - \beta_2^1 \cdot ... \cdot \beta_{j-1}^1 \cdot \alpha_i^\tau = \beta_2^1 \cdot ... \cdot \beta_{j-1}^1 \cdot \alpha_i^\tau \cdot (\beta_j^1 - 1)$ and $\beta_{j-1}^1$ is a probability constant meaning less than 1.

The second term of the equation is negative as $i$'s payoff $x_i^\tau = v_i \cdot \alpha_i^\tau$ is ranked lower then the $(j-1)|S|/K$ advertiser's payoff otherwise $i$ would have been matched to the $j-1$st slot when reporting $v_i$. ∎

□

The final case to consider is when matching the advertisers to their final slots.

Claim 4. *In the match the $K$ slots stage, advertiser i can not improve his utility by lying.*

Proof. Once again we must look at the two possibilities. If advertiser $i$ decreases his value meaning $\bar{v}_i < v_i$ then advertiser $i$ will be removed in an earlier stage in match-$K$-slots procedure and be allocated a lower slot than he would have if he reported his true value. Similarly to the case of "Allocate slots for sampling" part of the algorithm he can not change the click-through-rate found by the algorithm but he can be charged a different price when allocated in a lower slot. Just like in the "Allocate slots for sampling" stage $i$'s utility difference between allocation in slot $j$ when telling the truth and allocating in slot $j+1$ in lying is positive meaning that $i$ is worse off reporting $\bar{v}_i < v_i$.

The second possibility is of increasing his value in this case $i$ will remain in a later stage in match-$K$-slots procedure and be allocated a higher slot than he would have if he reported his true value. Similarly to the case of "Allocate slots for sampling stage" he can not change the click-through-rate found by the algorithm but he can be charged a different price when allocated in a higher slot. Just like in the "Allocate slots for sampling" stage $i$'s utility difference between allocation in slot $j$ when telling the truth and allocating in slot $j-1$ in lying is positive meaning that $i$ is worse off reporting $\bar{v}_i > v_i$. ∎

□

So assuming that our MAFIA algorithm finds the optimal welfare [5] with probability $(1-\lambda)$ bounded as above, then the MAFIA algorithm is truthful with probability $(1-\theta)$. ∎

## 5.2 Welfare Maximization

In this section we will prove that our algorithm approximates the optimal welfare. Our proof will closely follow the proof of [11]. However, before we can utilize their proof we have to deal with the truthfulness properties. If advertisers incorrectly report their values then it is obviously impossible to maximize welfare. The main problem, is that our proof of truthfulness assumes approximation of the welfare. Since the proof of approximation requires truthfulness we are in somewhat of a bind.

In order to resolve this problem we show that we can in fact decouple the two proofs. This will follow from the fact that the $\theta - truthfulness$ property and the $\lambda - welfare$ property are positively correlated. The more truthful we are the better welfare we can achieve and vice-versa. In other words we can set $\lambda$ to be small such that truthfulness is reached with probability $1 - \theta \rightarrow 1$ , then using lemma 3 and assuming truthfulness with probability 1, we show that there exist time period where the algorithm maximizes welfare with probability $1 - \lambda$.

Lemma 3. *Given a MAFIA algorithm which is truthful with probability $(1-\theta)$ that maximizes welfare at some time period $\tau$ with probability $1-\lambda$ then if $\theta$ increases then $\lambda$ increases and if $\theta$ decreases then $\lambda$ decreases.*

Proof. We will show that if $\theta$ decreases then $\lambda$ decreases. The other case is similar. If $\theta$ decreases it follows that the algorithm is truthful with higher probability. In this case the advertisers will report their true value with higher probability. As every advertiser $i$'s observed payoff $x_i^t = \bar{v}_i \cdot \alpha_i^t$, if $\bar{v}_i = v_i$ with high probability then $x_t^{\max_K} - x_t^i = \bar{x}_t^{\max_K} - \bar{x}_t^i$ with higher probability and therefore $x_t^{\max_K} - x_t^i \geq 2\gamma^l \Rightarrow \bar{x}_t^{\max_K} - \bar{x}_t^i \geq 2\gamma^l$ with higher probability. Thus the rejection/acceptance of the desired/undesired advertiser is with lower probability $\lambda$ ∎

[5] from some time period $\tau$

□

Lemma 4. *$\exists\tau$ such that the MAFIA algorithm finds the optimal welfare $\sum_{i\in N}\sum_{t=\tau}^{T}\alpha_i^t\cdot v_i$ with probability $1-\lambda$.*

Proof. The main argument of the proof is that the observed payoff $x_i^t$ of advertiser $i$ at time $t$ that was not removed is within $\gamma^l$ of the true payoff $x_i$. As $\gamma^l$ goes to zero as $l$ increases (which is to say as $t$ increases) then after long enough time we are left with the advertiser with the best payoff, i.e. maximized social welfare. As our MAFIA algorithm allocates $K$ slots and not just one we also need to verify that we get the best payoff in every slot of the $K$ slots and therefore maximize social welfare for all slots.

We will start by showing the former:
Let $S_t$ be the set of advertisers left in the auction at time $t$. For any time $t$ and advertiser $i\in S_t$ we have that,

$$\begin{gathered}\Pr[|x_i^t-x_i|\geq \\ \gamma^l\cdot\max\{\beta_2^1\cdot\ldots\cdot\beta_K^1,\tilde{\beta}_K^2\cdot\tilde{\beta}_{K-1}^1\cdot\ldots\cdot\tilde{\beta}_2^1\}]\leq \\ e^{-(\gamma^l\cdot\max\{\beta_2^1\cdot\ldots\cdot\beta_K^1,\tilde{\beta}_K^2\cdot\tilde{\beta}_{K-1}^1\cdot\ldots\cdot\tilde{\beta}_2^1\})^2t}\leq\frac{\delta}{cnt^2K}\end{gathered}\tag{2}$$

The first inequality follows from the Chernoff bound and the second by substituting $\gamma^l$ in the bound[6]. By union bound over the $K$ slots and by union bound over all times from $t=1$ to $T$ it follows that with probability at least $1-\delta/n$ for any time $t$ and any advertiser $i\in S_t$, $|x_i^t-x_i|\leq\gamma^l\cdot\max\{\beta_2^1\cdot\ldots\cdot\beta_K^1,\tilde{\beta}_K^2\cdot\tilde{\beta}_{K-1}^1\cdot\ldots\cdot\tilde{\beta}_2^1\}$. Therefore with probability $1-\delta$, the $K$'th best advertisers are never eliminated.

Now we need to show that we get the best $j$'th advertiser for each slot $j$ of the $K$ slots. That will be showed in $K$ iterations. First we want to show that for the first time period $t$ where $|S_t|=K$ the lowest payoff in $S_t$ is greater by at least $2\gamma^l$ of the highest payoff in $N\backslash S_t$. Similarly to inequality (2) with probability $1-\delta$, the best $K$'th advertiser is never eliminated.

Second we want to show that for the first time period $t$ where $|S_t|=K-1$ the lowest payoff in $S_t$ is greater by at least $2\gamma^l$ of the highest payoff in $N\backslash S_t$. Similarly to inequality (2)

$$Pr[|x_i^t-x_i|\geq\gamma^l\cdot\max\{\beta_2^1\cdot\ldots\cdot\beta_K^1,\tilde{\beta}_K^2\cdot\tilde{\beta}_{K-1}^1\cdot\ldots\cdot\tilde{\beta}_2^1\}]\leq$$

$$e^{-(\gamma^l\cdot\max\{\beta_2^1\cdot\ldots\cdot\beta_K^1,\tilde{\beta}_K^2\cdot\tilde{\beta}_{K-1}^1\cdot\ldots\cdot\tilde{\beta}_2^1\})^2t}\leq\frac{\delta}{cnt^2(K-1)}$$

By union bound over all times from $t=1$ to $T$ it follows that with probability at least $1-\delta/n$ for any time $t$ and any advertiser $i\in S_t$, $|x_i^t-x_i|\leq\gamma^l\cdot\max\{\beta_2^1\cdot\ldots\cdot\beta_K^1,\tilde{\beta}_K^2\cdot\tilde{\beta}_{K-1}^1\cdot\ldots\cdot\tilde{\beta}_2^1\}$. Therefore with probability $1-\delta$, the best $(K-1)$'th advertiser is never eliminated.

It is easy to see that all remaining $j\in|S_t|=K-2$ best $j$'th advertisers are never eliminated. As the algorithm's sample complexity is bounded (see lemma 5) and since the best $K$'th advertisers are ordered in the optimal order with probability $1-\delta$, as shown above, there exist time period $\tau$ where the algorithm finds the optimal welfare $\sum_{i\in N}\sum_{t=\tau}^{T}\alpha_i^t\cdot v_i$ with probability $1-\delta$ ∎

□

## 5.3 Sampling Complexity

Let $\beta=\max\{\beta_2^1\cdot\ldots\cdot\beta_K^1,\tilde{\beta}_K^2\cdot\tilde{\beta}_{K-1}^1\cdot\ldots\cdot\tilde{\beta}_2^1\}$ and denote by $\widehat{x}_i$ the real payoff of advertiser $i$ and by $x_t^i$ the observed payoff of advertiser $i$ at time $t$. Let $\widehat{x}_{\max_K}$ be the advertiser with the $K$'th highest real payoff and let $\Delta_i=\widehat{x}_{\max_K}-\widehat{x}_i$.

Lemma 5. *The MAFIA algorithm sample complexity is bounded by $O\left(\sum_{i=K+1}^{n}\frac{\log(\frac{n}{\delta\cdot\Delta_i})}{\Delta_i^2\cdot\beta\cdot K}+\sum_{i=2}^{K}\frac{\log(\frac{n}{\delta\cdot\Delta_i})}{\Delta_i^2\cdot\beta\cdot(K+1-i)}\right)$*

Proof. To prove the algorithm's sample complexity we need to bound the number of time rounds it will take to remove an advertiser.

Consider advertiser $i$ which should be removed then $x_t^{\max_K}-x_t^i\geq2\gamma^l$.

The assumption that advertiser's real payoff and the observed one are different by at most $\gamma^l$ for every time $t$, i.e., $|x_t^i-\widehat{x}_i|\leq\gamma^l$ yields that $\Delta_i-2\gamma^l=(\widehat{x}_{\max_K}-\gamma^l)-(\widehat{x}_i+\gamma^l)\geq x_t^{\max_K}-x_t^i\geq2\gamma^l$. The last inequality follows from the fact that $i$ is considered for removal. Since $\Delta_i$ is a constant and $\gamma^l\to0$ there exists a $t$ s.t. $\Delta_i\geq4\cdot\gamma^l$. By substituting $t=O\left(\frac{\log(\frac{n}{\delta\cdot\Delta_i})}{\Delta_i^2\cdot\beta\cdot K}\right)$ it hold that $\Delta_i>4\cdot\sqrt{\frac{\log(cnt^2/\delta)}{t}\cdot\frac{1}{\beta}\cdot\frac{1}{K}}$. As the algorithm removes all but $K$ advertisers in the first stage and then places every advertiser in its slot $j$ by removing it from the above slot $j-1$, the sample complexity of the algorithm is then (approximately) $\sum_{i=2}^{n}t$.

□

## 5.4 Bounding the welfare lost by sampling

Lemma 6. *The total lost in welfare resulting of the sampling process is tightly bound by $O(\sum_{i=2}^{n}\Delta_i)$.*

Proof. The welfare loss per advertiser $i$ which is removed at some time period must be at least $\Delta_i$ since each advertiser is sampled at least once.

With arbitrary high probability the advertiser is sampled at most until stage $l'$ s.t. $\Delta_i\leq2\gamma^{l'}$. Therefore, $\Delta_i<O(\sqrt{\frac{\log l'^2}{l}})$. Or $\Delta_i^2=O(\frac{\log l'}{l'})$. It can be seen (by substitution) that the solution for $l'$ is $l'=O(\frac{1}{\Delta_i^2}\log\frac{1}{\Delta_i^2})$. Since the welfare lost is $\Delta_i*l'$ the loss is $O(\Delta_i*\frac{1}{\Delta_i^2}\log\frac{1}{\Delta_i^2})=O(\frac{1}{\Delta_i}\log\frac{1}{\Delta_i^2})=O(\frac{\log\Delta_i}{\Delta_i})\to_{\Delta_i\to\infty}0$ so the loss is bounded by the loss for small $\Delta_i$ and hence the welfare loss is optimal (up to constants).

□

# 6. CONCLUSIONS AND FUTURE WORK

In this paper we presented a truthful multi-armed-bandit mechanism for discovering the valuations of advertisers in a slot-auction game. Our mechanism allows for slots with different quality while learning the click-through-rates of the advertisers and motivates them to report their true valuations per click. The mechanism presented obtains an optimal welfare, apart from a tightly bounded loss of welfare on the bandit sampling process, and achieves a decreased sampling cost.

Our results assumed that there are no budget constraints as well as concurrent and simultaneous start and stop times for all the advertisers. These assumptions can be relaxed

[6] $c$ is a constant

and are explored in our current work [15] which however assumes that all slots are of equal quality.

## Acknowledgements

The authors would like to thank David Pennock for his helpful comments.

## 7. REFERENCES

[1] A. Archer, C. Papadimitriou, K. Talwar, and E. Tardos. An approximate truthful mechanism for combinatorial auctions with single parameter agents. *In Proc. of the 14th SODA, 2003.*

[2] G. Aggarwal, A. Goel and R. Motwani. Truthful Auctions for Pricing Search Keywords. *Proceding of EC'06*

[3] D. Ariely, G. Loewenstein, and D. Prelec. Tom Sawyer and the Myth of Fundamental Value. *Journal of Economic Behavior and Organization, forthcoming, 2005*

[4] D. Ariely, G. Loewenstein, and D. Prelec Coherent arbitrariness: Stable demand curves without stable. *Quarterly Journal of Economics, 2003*

[5] D. A. Berry and B. Fristedt. Bandit problems. Sequential allocation of experiments. *Chapman and Hall 1985*

[6] D. Bergemann and J.Valimaki. Bandit Problems. *Social Science Research Network Electronic Paper Collection.*

[7] Bergemann, Dirk and Valimaki, Juuso "Learning and Strategic Pricing," emph Econometrica, Econometric Society, vol. 64(5), pages 1125-49, September. 1996.

[8] Dynamic price competition Dirk Bergemann and Juuso Valimaki *Journal of Economic Theory, Volume 127, issue 1,pp. 232-263* 2006

[9] Bergemann and Valimaki Efficient Auctions, Available at: $http : //papers.ssrn.com/sol3/papers.cfm?abstract_id = 936633$

[10] B. Edelman, M. Ostrovsky and M. Schwarz. Internet Advertising and the Generalized Second Price Auction: Selling Billions of Dollars Worth of Keywords. *Working paper 2005*

[11] E. Even-Dar, S. Manor, and Y. Mansour. PAC Bounds for Multi-Armed Bandit and Markov Decision Processes. *The Fifthteenth Annual Conference on Computational Learning Theory 2002*

[12] D. Fudenberg and J. Tirole Game Theory *MIT Press. (1991).*

[13] J.C. Gittins. Multi-armed Bandit Allocation Indices. *Wiley, New York. Mathematical Reviews: MR90e:62113 (1989)*

[14] R. Gonen. Untruthful Behavior in Google Slot Auctions. *Unpublished manuscript 2004*

[15] R. Gonen and E. Pavlov. Discovering value in repeated games. *Working paper 2006.*

[16] R. Gonen and E. Pavlov. False bid prevention in sponsered search. *Working paper 2007.*

[17] D. Kahneman and A. Tversky. Prospect Theory: An Analysis of Decision under Risk. *Econometrica 47, 263-291. (1979)*

[18] D. Kahneman. Loss Aversion in Riskless Choice: A Reference Dependent Model. *Quarterly Journal of Economics 106, 1039-1061. (1991)*

[19] R. Kleinberg. Anytime Algorithms for Multi-Armed Bandit Problems. *Proceedings of the 17th ACM-SIAM Symposium on Discrete Algorithms (SODA 2006).*

[20] S. Pandey, and C. Olston. Handling Advertisements of Unknown Quality in Search Advertising. *NIPS 06*

[21] H. Robbins. Some Aspects of the Sequential Design of Experiments. *In Bulletin of the American Mathematical Society, volume 55, pages 527535, 1952.*