Advertising Network Formation based on Stochastic Diffusion Search and Market Equilibria

Nikos Salamanos*
Athens University of Economics and Business
salaman@aueb.gr

Stavros Lopatatzidis
Athens University of Economics and Business
stauros.lopatatzidis@gmail.com

Michalis Vazirgiannis†
Athens University of Economics and Business & LIX, Ecole Polytechnique, France
mvazirg@aueb.gr

Antonis Thomas
Athens University of Economics and Business
antonis.thomas@gmail.com

ABSTRACT

The concept of social networks in conjunction with concepts from economics has attracted considerable attention in recent years. In this paper we propose the Stochastic Diffusion Market Search (SDMS), a novel contextual advertising method for mutual advertisement hosting among participating entities, where each owns a web site. In the scenario considered each participating agent/web-site buys or sells advertising links. In the proposed method the advertising market and network that formed into the system emerge from agents preferences and their social behavior into the network. SDMS consists of a variation of Stochastic Diffusion Search, a swarm intelligence metaheuristic, and an algorithm for market equilibria. We present an evaluation of the model and the experimental results show that the network potentially converges to a stable stage and the distribution of market prices adheres to power law properties.

Categories and Subject Descriptors

J.4 [**Social And Behavioral Sciences**]: Economics

General Terms

Economics, Swarm Intelligent, Market Equilibrium

Keywords

Online advertising, Market Mechanisms, Swarm Intelligent

*This research project is co-financed by E.U.-European Social Fund (75%) and the Greek Ministry of Development-GSRT (25%).

†Partially supported by the DIGITEO "LEVETONE" grant.

1. INTRODUCTION

In this paper we propose the *Stochastic Diffusion Market Search* (*SDMS*), an advertising method for mutual advertisement hosting among participating websites. Our motivation was to design an advertising method which encapsulates the characteristics of social networks and online markets. The design methodology of SDMS based on the Stochastic Diffusion Search (SDS) [2, 12], a population based *Swarm Intelligent metaheuristic*. Specifically, the SDMS is a social model of consumers' behavior when their preferences in the market are uncertain. The SDMS takes as input the set of advertisers and ad-publishers. Then based on the advertisers' indirect communication and preferences (advertisers' social role) the system produces the advertising network. The advertising network is a directed bipartite graph where each edge corresponds to a link that connects the ad-publisher (the web site that wants to host advertisements) to the advertiser.

The main features of our settings are that: a. there is no central authority (a search engine for instance) that defines and controls the advertisements matching b. the advertisers do not have direct access to ad-publishers' content (i.e. the content of the landing pages). Thus, a main issue arises: the advertisers are partially informed about the quality, the advertising similarity, of the participating web sites. Therefore, the preferences of advertisers/buyers to ad-publishers/goods have to be determined by a social procedure.

The SDMS formulates, in consecutive time periods, a *Fisher market*, a market consisting of buyers and divisible goods. We map each advertiser to a buyer and each ad-publisher (advertising slot) to a divisible good, the number of ad impressions that she is able to sell. Without harm to the generality of the design we make the assumption that the utility function of an advertiser for a given ad-publisher is linear. In order the method to be economically efficient, the *Cost Per Impression* (*CPI*) prices of the advertising slots have to be equal with the market equilibrium prices where the market clears (*market clearing* prices).

Thus, we devise an automatic pricing mechanism based on the polynomial time *tatonnement* algorithm (further called tatonnement) proposed by [7]. The tatonnement computes the equilibrium vector of prices for an exchange market. The

Fisher market is a special case of the exchange market. In our model the tatonnement is a subroutine which estimates the equilibrium CPI values.

We present an extensive evaluation of the method verifying that the network converges to a stable state (maximizing the *social welfare*) and that the distribution of market prices adheres to power law properties. Both results present a solution with attractive macroscopic properties that can be deployed in a real system.

The paper is organized as follows. Section 2 covers the related work. In section 3, we present briefly the SDS. In sections 4 and 5 we present the market formulation, the tatonnement algorithm and the SDMS. In section 6 we evaluate our method and finally section 7 concludes the paper.

2. RELATED WORK

Online advertising has attracted considerable attention in recent years. A case especially studied is the sponsored search advertising. Sponsored search advertising models involve auction models with the baseline usually being the *Generalised Second Price auctions (GSP)* [6].

In this paper we consider the case of contextual advertising. Our model based on the Stochastic Diffusion Search [2, 12] a Swarm Intelligent metaheuristic [3, 11] able to solve a variety of optimization problems. The time complexity and the convergence analysis of Stochastic Diffusion Search are presented in [14] and [13].

The behavioral model of SDS is very similar to the naive learning models from Game Theory [9, 17] especially with the *imitation dynamics* [8, 16]. Also, a part of our work is inspired by [4] where Ceyhan et al. present a theoretic analysis of the evolution of a market in the presents of social influence.

Recently, Saberi et al. [15] proposed an advertising exchange network model applicable on a network of blogs. Their system generates a more abstract model of economies, the exchange economies, were the agents/blogs could act both as ad-publishers and advertisers. They adopt an approach similar to ours and they apply a variation of tatonnement [7] algorithm in order the market to reach the equilibria. The main difference is that a market is defined on an existing social network, whereas in ours the mechanism generates the social network and the market.

A theoretic analysis of the advertising networks is presented in [10] where the authors analyze the topology of the advertising network for the case where web sites buy and sell advertising links.

In market equilibrium bibliography, Devanur et al. [5] present the DPSV algorithm, the first polynomial algorithm to compute the market equilibria for the linear case of Fisher model.

3. STOCHASTIC DIFFUSION SEARCH

Stochastic Diffusion Search (SDS) was proposed as a population based metaheuristic algorithm to solve pattern matching problems and is applicable to Swarm Intelligent systems. The SDS consists of a population of agents that synchronously search for the optimal solution of an optimization problem. The objective function of optimization problem has to be transformed to a summation of individual terms.

Initially each agent initializes at random the terms of the objective function defining an initial *hypothesis* about the optimal solution. In a repeated process the agents, using a *test* criterion, evaluate some of the terms of their hypothesis. Usually the test is a binary function and there is a mapping from the test score to the solution space of the optimization problem. The test score values divide the agents in two categories a. *inactive* (dissatisfied) for which the test score of their hypothesis is false and *active* (satisfied). In every time period each inactive agents chooses at random an agent for communication. If the selected agent is active then the selecting agent copies her hypothesis. Due to stochastic nature of the procedure the SDS approximates the optimal solution.

There are many variations of the SDS algorithmic schema [12]. The basic algorithmic pattern of SDS consists of the following steps (See [12](p26)):

1. For all agents do
2. **Initialize**: Agent picks a random hypothesis
3. **Test**: Agent partially evaluates her hypothesis
   - If $test\ criterion = TRUE$, $agent = Active$ (satisfied)
   - Else $agent = Inactive$ (dissatisfied)
4. **Diffuse**
   - Inactive agent meets a randomly chosen Active agent
   - Inactive agent update-change hypothesis
5. **Repeat** until Halting criterion.

4. MARKET FORMULATION

A set of agents/web sites, form an advertising exchange network in terms of buying and selling advertising links among them. The agents participate in an repeated advertising process where time is divided into advertising periods. Every agent could serve as ad-publisher *(AdP)*, or as advertiser *(Adv)*. For the rest of the paper we use lower case letters when we refer to advertisers and capital for ad-publishers. The agents have to connect first to an authority web site, the *Center*, that is of limited role and does not define the advertising network. The advertisers search for coherent ad-publishers through the *Center*. The *Center* executes the search queries initiated by the advertisers and collects the feedback provided by the advertisers. In consecutive time periods a Fisher market of advertising slots is formed into the system and in each period the *Center* runs the tatonnement aiming at an equilibrium state.

The linear case of the Fisher model is as follows. Consider a market with a set of n buyers and a set of N divisible goods. Each buyer i has an amount b_i of money to purchase goods and a utility function $u_i(J)$, the utility of i when he obtains a unit of the good J. We assume that the utility function of each buyer is linear with respect to the amount of goods that it consumes. The linear case of Fisher model results to a equilibrium vector of prices - the market clearing prices - and an optimal allocation of goods, where the market clears i.e. supply matches demand.

In our model we formulate a Fisher market as follows: Given n advertisers and N ad-publishers, we map each advertiser i to a buyer with budget b_i and each ad-publisher

Table 1: Notation used in SDMS.

N	Set of **ad-publishers**	AdP	**Ad-publisher**
$\{AdP\}^t$	Ad-publishers with non-zero demand	ν_J^t	J's visitors
CPI_J^t	J's Cost Per Impression price	f_J^t	J's impression for sell

n	Set of **advertisers**	Adv	**Advertiser**
$\{AdP\}_i^t$	i's preferences	$\{\overline{AdP}\}_i^t$	i's ad-publishers
$u_i^t(J)$	i's utility	$f_{J\to i}^t$	Impressions bought by i
$c_{J\to i}^t$	Clicks on i's ad	$CTR_i^t(J)$	i's CTR on ad-publisher J
F_i^t	The *friends* of i (contextually similar ads)	$\{\overline{CTR}_i^t\}$	i's avg-*Click Through Rate* (Eq. 2)

$\{\overline{CTR}^t\}$	**Market** avg-*Click Through Rate* (Eq. 3)	$AdNet^t$	**Advertising Network**

J to a divisible good. The quantity of J is the number of impression f_J^t that is able to sell in period t. We define that $f_J^t = \nu_J^{t-1}$, where ν_J^{t-1} are the visitors at J's web site in period $t-1$. In order to initialize the quantities $f_J^{t=1}$, we apply an period $t = 0$ where the *Center* computes the number of visitors $\nu_J^{t=0}$.

The preference of an advertiser i for an ad-publisher J is defined by the utility value $u_i^t(J)$. The value $u_i^t(J)$ denotes the utility per click obtained by i when her advertisement is published by J. In our model we assume that $u_i^t(J) = CTR_i^t(J)$. The CTR of ad i is the *Click Through Rate*, the number of clicks divided by the number of impressions.

The tatonnement algorithm computes the equilibrium market clearing prices of the goods. In our case, the equilibrium vector of prices corresponds to the *Cost per Impression* (CPI) values. Moreover, the equilibrium prices define for each advertiser i an optimal basket of goods $\{f_{J\to i}^t\}$, where $f_{J\to i}^t$ is the number of impressions that i will buy from J (the advertising matching).

Finally, we define that each ad-publisher J publishes the ads sequentially, one ad per impression with probability $\frac{f_{J\to i}^t}{f_J^t}$.

4.1 Market Equilibria

We apply to our model the recent algorithmic results of algorithmic market equilibria in order to devise an automatic pricing mechanism. Market equilibria occur when the supply is equal to demand. The existence of market equilibria has been proved by Arrow and Debreu [1]. In [7] the authors present a tatonnement algorithm, similar to Walrasian tatonnement. The tatonnement is an natural iterative process which converges to market equilibria for a very particular specifications of the excess demand function. Briefly, consider an exchange market with n traders and N goods. Each trader has an endowment of goods. At a given price p the trader sells his endowment and buys a bundle of goods. Iteratively, the traders announce prices and compute the demand, if the demand exceeds their supply then they reduce their prices else they increase the prices. The process continues until the demand is equal to supply, then the prices converge to market equilibrium prices. We must emphasize that transaction only take place when the process has converged to the equilibrium state.

The algorithm in [7] approximates the market equilibria for the general case of exchange markets. Computes a $(1+\epsilon)$-approximate market equilibrium, where the demand is at

Algorithm 1 Tatonnement

```
N Goods (AdPs), n Buyers (Advs), Budgets: {b_i}_{i∈n}
Initialize:
Set for each good J: quantity(J) = 1
Scale utilities: û_i(J) = u_i(J) · f_J^t
Set initial prices p_J = 1 for 1 ≤ J ≤ N
Define w ∈ ℜ^N_{++}
for T = (N/δ) log_{1+δ} N, where δ = ε/2 · (1 + ε) do
    Find α > 0 such that π = α · p ∈ Π, where:
    Π = {π ∈ ℜ^N_+ | π · w = 1}.
    Announce prices π = α · p
    Compute:
    for each Adv x_i ∈ argmax{u_i(x) | π · x ≤ b_i}
    find aggregate demand X = Σ_i x_i
    compute σ = 1/max_J X_J
    update prices: p_J ← p_J · (1 + δ · σ · X_J)
end for
Compute:
    Equilibrium Demand in the rth iteration:
    for each Adv i: x̄_i = (Σ_{r=1}^T σ(r)·x_i(r)) / (Σ_{r=1}^T σ(r)), x̄_i ∈ ℜ^N
    Equilibrium Prices in the rth iteration:
    π̄ = (Σ_{r=1}^T σ(r)·π_i(r)) / (Σ_{r=1}^T σ(r)), π̄ ∈ ℜ^N
Output:
    For each AdP J: CPI_J = π̄_J / f_J^t
    For each Adv i the bundle of goods:
    (f_{1→i}, ..., f_{N→i}) = x̄_i · (f_1, ..., f_N)
```

most $(1+\epsilon)$ times the supply. The number of iterations is $T = \frac{N}{\delta} log_{1+\delta} N$, where $\delta = \epsilon/(2 \cdot (1+\epsilon))$.

An exchange market consists of n traders and N goods. The traders has an endowment $w_i = (w_{i1}, ..., w_{iN})$ of goods and the total quantity of a good, in the market, is the sum of the traders endowments. The traders could act as sellers and as buyers. They sell their endowment and they buy goods. Algorithm 1 presents our variation of [7] for the special case of the advertising Fisher market. Actually, our tatonnement algorithm is the tatonnement presented in [7] using a reduction from the exchange market to the Fisher market ([7], section 3).

Fleischer et al. [7] proposed an new formulation of the market equilibrium problem as a convex optimization problem using *indirect utility functions*. In each iteration the algorithm restricts the prices to a convex set $\Pi \subseteq \Re^N$, where

N is the number of goods. A Fisher market consists of buyers and goods. A Fisher market is a special case of an exchange market if we consider that the endowments of the traders are proportional: $w_i = b_i w$, where $w \in \Re^N_{++}$. In this case we set $\Pi = \{\pi \in \Re^N_+ | \pi \cdot w = 1\}$. ([7], section 3).

Initially the algorithm defines a transformed market were the quantities of the goods is unit and scales appropriately the utilities (Alg. 1 step: 3,4). The initial prices $p = (p_1, ..., p_N)$ are set to one (Alg. 1 step: 5).

The next step is to define the appropriate convex set Π. In every iteration we restrict the prices p to the convex set $\Pi = \{\pi \in \Re^N_+ | \pi \cdot w = 1\}$, where $\pi = \alpha \cdot p$ (Alg. 1 step: 6,8). In other words, we define an vector $w \in \Re^N_{++}$ and in every iteration we compute the $\alpha > 0$ for which $\pi \cdot w = \alpha \cdot p \cdot w = 1$ (the inner product of π and w).

Finally, observe that the algorithm computes the equilibrium vector of prices for the transformed market where we consider the good as a unit. Thus, we have to compute for each good the price per unit and for each advertiser the optimal basket of goods (Alg. 1 step: 23-25).

5. THE SDMS METHOD

We propose the *Stochastic Diffusion Market Search* (*SDMS*) a social model of consumers behavior where their preferences in the market are uncertain. In this paper we study the special case of advertising market. The general algorithmic schema of SDMS is a variation of SDS presented in [12](p154). We assume that the consumer, the advertiser in our case, entering to an known market cannot predefine her preferences i.e. the ad-publishers on which it prefers to be published on.

In consecutive time periods $t = 1, 2, \ldots, \infty$, each advertiser searches for the most coherent ad-publishers. The objective function of the advertiser is to determine the set of ad-publisher where her ad will gather the max number of clicks, subject to the quality of the clicks (CTR values). She follows a stochastic behavior and her preferences (ad-publishers) emerge from a trial and error process (tests).

The advertiser's ***hypothesis*** consists of the set $\{AdP\}_i$ of the candidate ad-publishers and the vector of utility values $u_i(\cdot)$. The advertiser submits her hypothesis to the tatonnement. The tatonnement computes a. for each ad-publisher J the market equilibrium CPI^t_J prices and b. for each advertiser i the set $\{\overline{AdP}\}^t_i$ of ad-publishers and the optimal basket $\{f^t_{J\to i}\}_{J\in\{\overline{AdP}\}^t_i}$.

During the ***test phase*** the advertiser compares the realized output of her hypothesis, the average CTR of her ad (i.e the average utility), with the average CTR of the ads in the market (i.e market average utility). If the ***test score*** is *false* then updates her hypothesis in the ***diffusion phase***. In the following we present in more details these two phases.

5.1 Initialization Phase

In the first period each advertiser has to choose an initial hypothesis, one ad-publisher J that will submit to tatonnement. Each ad-publisher J corresponds to a divisible good, an advertising slot with $f^{t=1}_J$ impressions (quantity). Recall that the $f^{t=1}_J$ is the number of visitors in period $t = 0$. We assume that a given advertiser i is equally uncertain for the quality of ad-publishers, thus she selects at random one ad-publisher with probability proportional to the quantity of the goods (Eq. 1).

$$Pr_i(J) = \frac{f^t_J}{\sum\limits_{J\in\{AdP\}^t} f^t_J} \tag{1}$$

In terms of SDS, the tuple $(\{AdP\}^{t=1}_i, u^{t=1}_i(J))$, where $u^{t=1}_i(J) = 1$, corresponds to the advertiser's initial *hypothesis*$(i, t = 1)$.

5.2 Test Phase

The *Test Phase* (Alg. 2) consists of a. the subroutine *Tatonnement*, b. the *Active Network* and c. the *Test Criterion.*

The subroutine *Tatonnement* is an automatic pricing schema which takes as input a. the budgets b_i and the utilities $u_i(\cdot)$ of advertisers b. the set $\{AdP\}^t$, the ad-publishers with at least one potential buyer and c. the estimated f^t_J number of impressions (the quantities of the goods). The tatonnement outputs the CPI prices and the allocation of goods to buyers i.e. the advertising network $AdNet^t$. The $AdNet^t$ is defined by a. the equilibrium CPI prices b. the advertising matching *Adv-AdP* and c. the number of impressions $f^t_{J\to i}$ that each advertiser has bought.

The *Center* activates the $AdNet^t$ and at the end of each period t it returns a. the realized number $c^t_{J\to i}$ of clicks on i's ad and b. the average realized values, $\overline{CTR}^t_i$, and $\overline{CTR}^t$ where:

$$\overline{CTR}^t_i = \frac{Total\ clicks\ on\ Adv\, i}{Total\ impressions\ of\ Adv\, i} = \frac{\sum\limits_{J\in\{\overline{AdP}\}^t_i} c^t_{J\to i}}{\sum\limits_{J\in\{\overline{AdP}\}^t_i} f_{J\to i}} \tag{2}$$

$$\overline{CTR}^t = \frac{Total\ clicks\ on\ Advs}{Total\ impressions\ consumed} = \frac{\sum\limits_i \sum\limits_{J\in\{\overline{AdP}\}^t_i} c^t_{J\to i}}{\sum\limits_{J\in\{AdP\}^t} f_J} \tag{3}$$

The set $\{\overline{AdP}\}^t_i$ is the ad-publishers of i. The value $\overline{CTR}^t_i$ corresponds to the average utility (per click) of i with respect to her *hypothesis*(i, t). It is considered as the effectiveness (the quality) of $\{\overline{AdP}\}^t_i$ as ad-publishers.

The ***test criterion*** for a given advertiser i and for her *hypothesis*(i, t) is the distance between $\overline{CTR}^t_i$ and $\overline{CTR}^t$. The ***test score*** is *true* if $\overline{CTR}^t_i \geq \theta \cdot \overline{CTR}^t$, else is *false*. The parameter θ, $0 < \theta \leq 1$, is defined by i. If *true* then i is considered as *active* (satisfied) and in period $t + 1$ submits the same hypothesis to the tatonnement. Else, i is considered as *inactive* (dissatisfied) and she has to execute the diffusion phase.

Observe that the test score executed on the output of tatonnement and ad-network i.e. the equilibrium mapping of *Advs* to *AdPs*, the CPI prices and the realized similarity *Adv-AdP*.

5.3 Diffusion Phase

During the *Diffusion Phase* (Alg. 3) each *inactive* advertiser i has to update her *hypothesis*$(i, t-1)$ for the period

Algorithm 2 Test Phase

for $t = 1$ to ∞ **do**
  for all *AdP* J **do**
    set $f_J^t = \nu_J^{t-1}$
  end for
  for all *Adv* i **do**
    Collect $hypothesis(i,t) = (\{AdP\}_i^t, \{u_i^t(\cdot)\})$,
  end for
  Compute the *AdPs* with a potential buyer (active goods in the market):
  $\{AdP\}^t = \{J \mid \exists i, u_i^t(J) \neq 0\}$
  Run *Tatonnement*
  Return *Ad-Network*:
  $AdNet^t = \{(i, \{\overline{AdP}\}_i^t, \{f_{j\to i}^t\})\}, CPI_J^t$
  Activate *Ad-Network*
  Return the realized values:
  $c_{J\to i}^t, \overline{CTR}_i^t$
  Test Criterion:
  for all *Adv* i **do**
    if $\overline{CTR}_i^t < \theta \cdot \overline{CTR}^t$ **then**
      Inactive (Test Score=False)
      go to *Diffusion Phase*
    else
      Active (Test Score=True)
      (choose the same hypothesis updated by the realized values)
      $hypothesis(i, t+1) = (\{AdP\}_i^{t+1}, \{u_i^{t+1}(\cdot)\})$, where:
      $\{AdP\}_i^{t+1} = \{\overline{AdP}\}_i^t$ and $u_i^{t+1}(J) = CTR_i^t(J)$
    end if
  end for
end for

Algorithm 3 Diffusion Phase

1: **for** $t = 1$ to ∞ **do**
2:   **for** every **Inactive** *Adv* i **do**
3:     Choose new hypothesis:
4:     **Run *Find Friends***
5:     {
6:     Choose with prob. $\frac{1}{\#ads}$ one new ad g
7:     **if** $g = friend$ is TRUE (i.e g's ad similar to i's ad) **then**
8:       Set $F_i^t = F_i^{t-1} \cup \{g\}$
9:     **else**
10:       Set $F_i^t = F_i^{t-1}$
11:     **end if**
12:     }
13:     Return *friends* F_i^t
14:     Choose by the probabilistic rule Eq. 4 one friend k in F_i^t
15:     **if** k *Active* AND $\overline{CTR}_k^t > \overline{CTR}_i^t$ **then**
16:       copy k's preferences:
17:       $hypothesis(i, t+1) = (\{AdP\}_i^{t+1}, \{u_i^{t+1}(\cdot)\})$,
18:       where:
19:       $\{AdP\}_i^{t+1} = \{\overline{AdP}\}_i^t \cup \{\overline{AdP}\}_k^t$
20:       $u_i^{t+1}(J) = \begin{cases} CTR_i^t(J) & \text{if } J \in \{AdP\}_i^t \\ CTR_k^t(J) & \text{if } J \in \{AdP\}_k^t - \{AdP\}_i^t \end{cases}$
21:     **else**
22:       choose with uniform probability one new *AdP* J'
23:       $hypothesis(i, t+1) = (\{AdP\}_i^{t+1}, \{u_i^{t+1}(\cdot)\})$
24:       where:
25:       $\{AdP\}_i^{t+1} = \{\overline{AdP}\}_i^t \cup \{J\}$
26:       $u_i^{t+1}(J) = \begin{cases} CTR_i^t(J) & \text{if } J \in \{\overline{AdP}\}_i^t \\ 1 & \text{if } J' New \end{cases}$
27:     **end if**
28:   **end for**
29: **end for**

t. She chooses at random one advertiser and compares the $hypothesis(i, t-1)$ with hers.

SDS is a population based method where an implicit assumption is that the population of agents are homogeneous. Thus, during the diffusion phase agents' recommendations are valid. In our case a main issue arises, their is not a prior known population of advertisers that an given advertiser i could ask for recommendations. Thus, each advertiser has to define the subset of ads that are similar with hers.

We apply a method where in every period each advertiser explores the set of ads in the market, one ad per period, and characterizes with true/false if it is contextually similar to her ad or not. The set F_i^t of similar ads is called the *friends* of i. This will be used as input to the diffusion phase.

Specifically, the subroutine ***Find Friends*** encompasses the following steps. In each period, each advertiser chooses exactly one ad g with probability $\frac{1}{\#ads}$. Then, if the selected ad is similar with hers she updates her set of friends. Finally, she chooses an ad k by the probabilistic rule Eq.4.

$$Pr_i(k) = \frac{w_i(k)}{\sum_{k \in F_i^t} w_i(k)} \quad (4)$$

The value $w_i(k)$ represents i's ***social criterion*** for the advertiser k and corresponds to the influence that k has on the community of advertisers. We define:

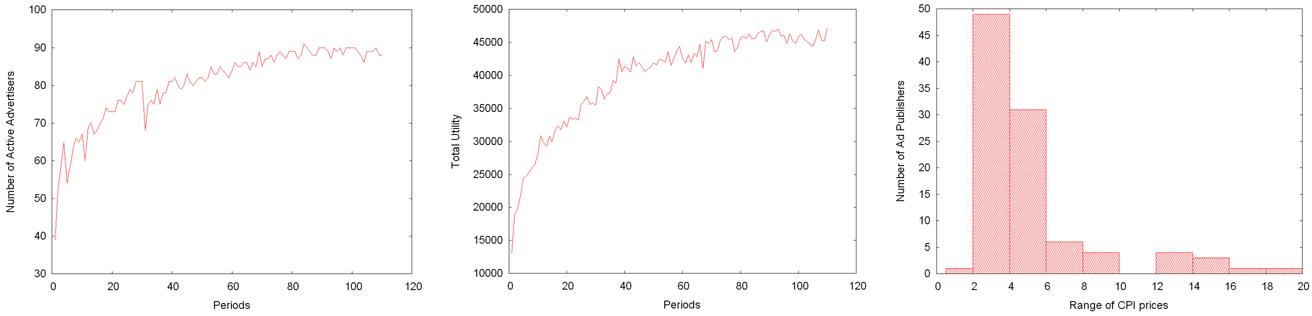
$$w_i(k) = \alpha\cdot(indegree(k)+1)+\beta\cdot(\#commonfriends+1) \quad (5)$$

where $\alpha + \beta = 1$. The parameters α and β are defined by i. The value one in the *in-degree* has the meaning that any advertiser has a link with herself and consequently positive probability to be collected even if she is isolated. Also, the value one in the *common friends* has the meaning that any advertiser is *friend* with herself. The *common friends* values give weight to the symmetric relations i.e. the case where i votes k as a friend and k votes i. The intuition is that the *in-degree* reflects advertiser's social influence and the *common friends* the convergence of advertisers' subjective beliefs with respect to ads similarity.

Briefly, the main diffusion phase conducts the following actions: i selects at random one friend k. If k is *active* and k's hypothesis is better than i's, then i copy k's preferences (Alg. 3, step 15-20). Else, she chooses at random one ad-publisher J' (Alg. 3, step 22-26). Observe that at step-19 agent i combines her hypothesis with k's by performing the union of the two relevant sets. The intuition is that the i cannot predict the externalities in the market. Thus, she submits to tatonnement her realized set of ad-publishers along with her subjective beliefs.

6. EVALUATION OF THE METHOD

In this section we present the experimental evaluation of the SDMS. The simulation and the tatonnement was imple-



(a) (b) (c)

Figure 2: a) Convergence State b) Social welfare c) Distribution of CPI prices in SDMS.

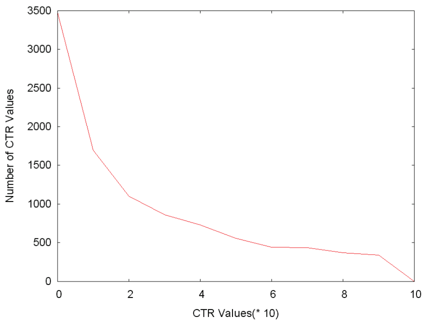


Figure 1: *Zipf* distribution of realized $\overline{CTR}$ values.

mented in MATLAB. Due to computational limitations, we simulated a market with 100 ad-publishers and 100 advertisers. During the simulation we set $\epsilon = 2$, thus the tatonnement computes a $(1+2)$-approximate market equilibrium. Also, we set α and β equal to 0.5 (Eq. 5) and $\theta = 0.9$ (Alg. 2 step 18). The experiments were performed on a Intel Core 2 Quad CPU Q8200 & 2.33GHz & 3.21GB RAM.

The experimental evaluation conducted on an artificially constructed data set. The number of visitors of each ad-publisher J is defined on the integer interval $[100, 1000]$ as follows: we pick with uniform probability an integer x from the interval $[0, 90]$, and we assign to J the $\nu_J = (x + 10) \cdot 10$ value. We consider that the number of visitors is constant in every period.

The advertisers' budget values range in the interval $[1000, 3000]$ as follows: we pick with uniform probability an integer x from the interval $[0, 2000]$ and we assign to advertiser i the budget $b_i = x + 1000$.

The $\overline{CTR}_i^t(J)$ values i.e the realized *Adv-AdP* similarity are approximated by a *Zipf* distribution with *Zipf* exponent 1.0. We generate for each advertiser i 100 *Zipf* values which correspond to the realized CTR values of the 100 ad-publishers. During the experiments the CTR values considered constant. In summary we generate $100 \cdot 100$ *Zipf* values. CTR distribution is presented in Fig. 1.

The *friends* of the advertisers are defined by a matrix 100×100 where each cell $[i, g]$ has the value 1 if i is friend with g and 0 otherwise. We constructed the matrix as follows: On a given advertiser i we assign with uniform probability an integer from the interval $[0, 9]$, this is the *color* of i. We repeat for all advertisers. Advertisers with the same *color* considered as similar (friends). We make the assumption that when i votes g as friend then g also considers i as friend. Thus, i and g, in other words, have the same subjective beliefs with respect to their ad similarity. This procedure creates 10 groups of friends. The previous process generates group of advertisers with the same *color* (subjective similarity) but probably with different CTR values on the same ad-publisher (realized similarity).

In the first experiment, we estimate the *convergence state* of SDMS, the state where a reasonable solution is discovered (see p.32 in [12]). The number of *active* advertisers is computed (Fig. 2(a)) for 110 consecutive periods. Recall that we consider as *active* the advertisers that are satisfied i.e. their average CTR (Eq. 2) is greater than the market average CTR (Eq. 3). Observe that the activity of advertisers is converging fast, as after the 20th period the SDMS process accelerates and after 60 periods more than 80% of the advertisers are active. Thus, the SDMS process potentially converges to an stable stage.

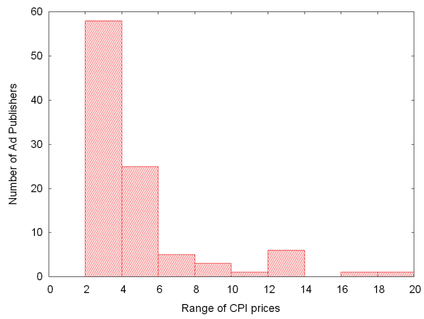


Figure 3: Distribution of CPI in the full information case.

Next, we deal with the social welfare of the network. In every period, we measure the sum of the total utility values $\sum_i \sum_J c_{J \to i}$, corresponding to the social welfare in the network. The results depict in Fig. 2(b). We observe that as the number of periods increased the social welfare also increased converging to stable level. Thus results verify our intuition that the SDMS is applicable to a social network with heterogeneous agents.

In Fig. 2(c) we present the distribution of CPI prices in SDMS after 110 periods and in Fig. 3 the distribution of

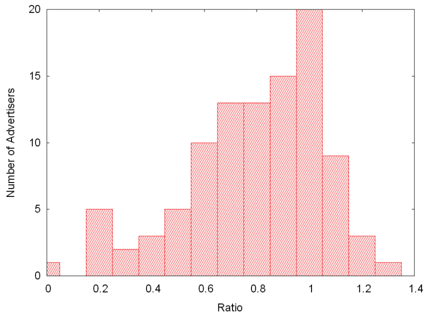


Figure 4: Total utility in SDMS vs full information case.

CPI in the case of the mature market. A mature market is a market where the preferences of advertisers is well defined, implying that the advertisers are fully informed about the realized CTR values. Thus, we run the tatonnement assuming that the advertisers are aware of the realized CTR values and these values are submitted to the tatonnement process. The experiments present that the distribution in SDMS is very similar with the full information case. It is also clear in both cases that a small number of *AdPs* sell in high prices. Thus, we claim that the stage of the market after the 110th period depicts power law properties.

Finally, we compare the SDMS total utility allocation after 110 periods to the total utility allocation in a mature market Fig. 4. We compute the ratio of total utility obtained in SDMS divided by the total utility obtained in the full information case. The results show that more than 50% of the population of advertisers have ratio greater than 0.6. This result verifies the effectiveness of our method since it approximates the full information case.

In summary, the experimental results verify that the network converges to a stable state and that the distribution of market prices adheres to power law properties. Both results present a solution with attractive macroscopic properties that can be deployed in a real system.

7. CONCLUSION

In this paper we proposed a novel method the SDMS (Stochastic Diffusion Market Search) for the formation of an advertising network consisting of a population of independent agents/web sites.

The innovation of SDMS lies in the fact that advertisement allocation emerges from indirect communication among the participants. The *Center*, the moderator of the system, is of limited role and does not define the advertising network. The *Center* executes the subroutines *tatonnement* and *ad-networks* but it does not define the preferences of the advertisers and their network of *friends*. All the parameters of the market, the *test* and the *diffusion* phase as well as the subjective beliefs of advertisers with respect to advertising similarity are defined only by the agents.

The SDMS integrates the algorithmic schema of *Stochastic Diffusion Search*, a Swarm Intelligent metaheuristic. Our method consists of a *Test Phase* where the advertisers evaluate their choices in the advertising market and a *Diffusion Phase* where update their preferences. The experimental results show that the network converges to a stable stage and the distribution of market prices adheres to power law properties.

8. REFERENCES

[1] K. J. Arrow and G. Debreu. Existence of an equilibrium for a competitive economy. *Econometrica*, 22(3):265–290, July 1954.

[2] J. M. Bishop. Stochastic searching networks. In *Artificial Neural Networks, 1989. First IEE International Conference on (Conf. Publ. No. 313)*, pages 329–331, October 1989.

[3] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems.* Oxford University Press, New York, Oxford, 1999.

[4] S. Ceyhan, M. Mousavi, and A. Saberi. Social influence and evolution of market share. In *In the NetEcon*, 2009.

[5] N. R. Devanur, C. H. Papadimitriou, and V. V. Vazirani. Market equilibrium via a primal–dual algorithm for a convex program. *Journal of the ACM (JACM)*, 55(5):1–18, October 2008.

[6] B. Edelman, M. Ostrovsky, and M. Schwarz. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. *American Economic Review*, 97(1):242–259, March 2007.

[7] L. Fleischer, R. Garg, S. Kapoor, R. Khandekar, and A. Saberi. A fast and simple algorithm for computing market equilibria. In *WINE '08: Proceedings of the 4th International Workshop on Internet and Network Economics*, pages 19–30, December 2008.

[8] D. Fudenberg and L. A. Imhof. Monotone imitation dynamics in large populations. *Journal of Economic Theory*, 140(1):229–245, May 2008.

[9] D. Fudenberg and D. K. Levine. *The Theory of Learning in Games.* The MIT Press, Cambridge, MA, 1998.

[10] Z. Katona and M. Sarvary. Network formation and the structure of the commercial world wide web. *Marketing Science*, 27(5):764–778, September - October 2008.

[11] J. Kennedy, R. C. Eberhart, and Y. Shi. *Swarm intelligence.* Morgan Kaufmann Publishers, San Francisco, CA, 2001.

[12] K. D. Meyer. *Foundations of Stochastic Diffusion Search.* PhD thesis, University of Reading, 2003.

[13] S. J. Nasuto and J. M. Bishop. Convergence analysis of stochastic diffusion search. *Journal of Parallel Algorithms and Applications*, 14(2):89–107, July 1999.

[14] S. J. Nasuto, J. M. Bishop, and S. Lauria. Time complexity analysis of the stochastic diffusion search. In *Proceedings of the International ICSC / IFAC Symposium on Neural Computation (NC 1998)*, pages 260–266, September 1998.

[15] A. Saberi and A. Shkolnik. Advertising space exchange in a network using market equilibrium algorithms. In *Fifth Workshop on Ad Auctions (adauctions2009).*

[16] K. H. Schlag. Why imitate, and if so, how? : A boundedly rational approach to multi-armed bandits. *Journal of Economic Theory*, 78(1):130–156, January 1998.

[17] H. P. Young. *Individual Strategy and Social Structure: An Evolutionary Theory of Institutions.* Princeton University Press, Princeton, NJ, 1998.