Collective Influence Maximization for Multiple Competing Products with an Awareness-to-Influence Model

Dimitris Tsaras
Department of CSE, HKUST
dtsaras@connect.ust.hk

George Trimponias
Amazon.com
trimpog@amazon.lu

Lefteris Ntaflos
Department of CSE, HKUST
entaflos@connect.ust.hk

Dimitris Papadias
Department of CSE, HKUST
dimitris@cs.ust.hk

ABSTRACT

Influence maximization (IM) is a fundamental task in social network analysis. Typically, IM aims at selecting a set of seeds for the network that influences the maximum number of individuals. Motivated by practical applications, in this paper we focus on an IM variant, where the *owner* of multiple *competing products* wishes to select seeds for each product so that the *collective influence* across all products is maximized. To capture the competing diffusion processes, we introduce an *Awareness-to-Influence* (AtI) model. In the first phase, awareness about each product propagates in the social graph unhindered by other competing products. In the second phase, a user adopts the most preferred product among those encountered in the awareness phase. To compute the seed sets, we propose GCW, a game-theoretic framework that views the various products as agents, which compete for influence in the social graph and selfishly select their individual strategy. We show that AtI exhibits monotonicity and submodularity; importantly, GCW is a *monotone utility game*. This allows us to develop an efficient best-response algorithm, with quality guarantees on the collective utility. Our experimental results suggest that our methods are effective, efficient, and scale well to large social networks.

PVLDB Reference Format:
Dimitris Tsaras, George Trimponias, Lefteris Ntaflos, and Dimitris Papadias. Collective Influence Maximization for Multiple Competing Products with an Awareness-to-Influence Model. PVLDB, 14(7): 1124 - 1136, 2021.
doi:10.14778/3450980.3450981

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/dtsaras/GCW.

1 INTRODUCTION

Given a graph and an integer k, the objective of Influence Maximization (IM) is to discover a set of k seed nodes that influence the largest number of nodes in the graph. Finding the seed set with the maximum influence is NP-Hard according to most common models of influence spread. However, polynomial algorithms can obtain approximate solutions of high quality for models that exhibit *monotonicity* and *submodularity* [26]. Traditionally, IM is concerned with the diffusion of a single product in the social graph (the term "product" is generic and may also refer to ideas, innovations, technologies, etc.). On the other hand, in several emerging applications the *owner* of multiple *competing products* may be interested in their joint diffusion in the underlying social network. For example, an e-shop promoting various laptop models may wish to find seed sets of influencers for each model so that the total influence is maximized. As an alternative example, an agency may be in charge of advertising several social events on a given date, with the goal of maximizing the total attendance. The term "competing" essentially refers to the fact that products compete to get adopted by users. We consider that the owner has a budget for each product, which refers to the maximum number of seeds for that product. Users may have different *weights* for different products, which signify their importance. For instance, users whose profile or demographic characteristics match the advertised product have high weights.

Inspired by the above, we introduce the problem of *collective weighted influence maximization* CWim, where the owner of multiple competing products wishes to maximize their total weighted influence in a social graph. To capture the diffusion of the competing products we adopt an *awareness-to-influence* (AtI) model, which separates awareness and influence. AtI assumes that the influence of each product diffuses independently in the underlying social graph during the *awareness* phase. A node may thus become aware of several products, but in the end it adopts only one of them in the subsequent *influence* phase. For instance, although a user may find out about several events, he will choose the one most similar to his preferences (e.g., closest to his location).

We show that AtI possesses monotonicity and submodularity. Furthermore, we cast it as a game where products compete for influence in the social graph and show that it falls under the important class of *monotone utility games*. This enables a single-round best-response algorithm with an approximation bound on the collective utility. In summary, AtI is expressive and realistic enough to capture interesting dynamics for the diffusion of multiple competing products, while allowing for the efficient computation of solutions with good guarantees on the collective utility. Our contributions are summarized as follows:

- We formulate the collective influence maximization (CWim) problem for an owner with multiple competing products.
- We introduce an awareness-to-influence model (AtI) that captures the behavior of social network users, who process

Proceedings of the VLDB Endowment, Vol. 14, No. 7 ISSN 2150-8097.
doi:10.14778/3450980.3450981

and transmit the information they receive, before making a decision. We prove that AtI preserves submodularity and monotonicity.
- We compute the seed sets for the various products with a game-theoretic framework, termed GCW, where we treat the products as agents that compete for influence in the underlying social graph. We show that GCW falls under the class of *monotone utility games*, which leads to an efficient best-response algorithm with good quality guarantees. Furthermore, we are able to derive price of anarchy bounds.
- We evaluate our approach for different settings and real data sets, and demonstrate its high efficiency, solution quality, and scalability.

The remainder of the paper is organized as follows. Section 2 surveys influence diffusion models, and related work on competitive influence maximization. Section 3 formally defines the CWim problem, and presents the AtI diffusion model. Section 4 introduces a utility game with special properties that enable the fast game-theoretic algorithm of Section 5. Section 6 experimentally evaluates the proposed scheme, whereas Section 7 concludes our work.

2 BACKGROUND AND RELATED WORK

Section 2.1 overviews background on IM, and Section 2.2 related work on competitive IM.

2.1 Influence Maximization Background

Let $G = (V, E, P)$ be a directed social graph[1] , where V is the set of nodes, E is the set of edges, and P is the set of edge probabilities. Given a set S of *seed* nodes for a single product, there are several models that capture the spread of influence of S in G. Among the most popular ones is the *independent cascade* (IC) model [26]. In IC, all nodes except for the seeds are initially considered *inactive*. At time $t = 1$, each node v in S influences every neighbor u with a probability $p(v, u)$, i.e., the probability of edge $(v, u) \in E$. Influenced nodes remain *active* throughout the spread process. Every node activated at time t has a single chance, at step $t + 1$, to influence its inactive neighbors. The *influence set* $I_{IC}(S)$ of S under the IC model contains all active nodes at the end of the diffusion process.

Another popular diffusion model is *linear threshold* (LT) [26]. In LT, each node v_i has a threshold θ_i, chosen uniformly at random[2] in the range [0,1]. In addition, the influence probabilities of the incoming edges of v_i are subject to the constraint that $\sum_{(v,v_i)\in E} p(v, v_i) \leq 1$. According to the LT model, a node v_i is activated when the total probability of incoming edges from active neighbors reaches its threshold θ_i, i.e.,

$$\sum_{(v,v_i)\in E \wedge v:active} p(v, v_i) \geq \theta_i.$$

The *live edge* model (LE) [26] provides an equivalent view of the influence spread for IC and LT, which is order independent and facilitates proofs. Let X be a possible world, i.e., a sample outcome of the spread process. For IC, X is generated by selecting each edge $(v, u) \in E$ in advance with probability $p(v, u)$. Selected edges are considered *live*. For LT, assuming that thresholds are uniformly distributed in $[0, 1]$, each node v_i picks at most one of its incoming edges (v, v_i) with probability $p(v, v_i)$, and no edge with probability $1 - \sum_{(v,v_i)\in E} p(v, v_i)$.

In LE, node v is active in possible world X, if there is a path from a node in the seed set S to v, consisting only of live edges. Accordingly, the influence set $I^X_{\mathcal{M}}(S)$ of S in X under diffusion model $\mathcal{M}$ ($\mathcal{M}$ can be IC, LT or some other model that can be represented by LE) contains nodes reachable from S through live edges. Let $\sigma^X_{\mathcal{M}}(S) = |I^X_{\mathcal{M}}(S)|$ be the cardinality of $I^X_{\mathcal{M}}(S)$. By repeating the same process in all possible worlds, we obtain the influence (or expected spread) of S under $\mathcal{M}$: $\sigma_{\mathcal{M}}(S) = \sum_{\forall \text{world } X} Pr(X) \cdot \sigma^X_{\mathcal{M}}(S) = \mathbb{E}[|I^X_{\mathcal{M}}(S)|]$, where the expectation operator $\mathbb{E}[\cdot]$ is taken over all possible worlds X. Formally the influence maximization (IM) problem is defined as: *Given a graph G, a diffusion model $\mathcal{M}$, and an integer k, find the seed set S of cardinality k that maximizes the influence function $\sigma_{\mathcal{M}}(S) = \mathbb{E}[|I^X_{\mathcal{M}}(S)|]$ under $\mathcal{M}$*[3].

The IM problem is NP-Hard for IC and LT. Kempe et al [26] proposed an approximate greedy method, hereafter referred to as Gim. Gim starts with an empty seed set S, and at each step, it adds the node v that maximizes the *marginal gain* $\sigma(S \cup \{v\}) - \sigma(S)$; i.e., the addition of v to S yields the largest increase in the influence function. In *weighted* IM, each node has a weight proportional to its importance. In this case, the objective is to maximize the expected sum of weights of influenced users. Gim is still applicable; the only difference is that the marginal gain refers to the weight increase due to the addition of node v.

For IC and LT, and in general models where the objective function $\sigma(\cdot)$ exhibits *monotonicity* and *submodularity*, Gim yields a seed set whose influence set is within a factor $(1 - 1/e)$ of the optimal influence (for both the weighted and un-weighted versions), where e is the base of the natural logarithm. Let S_1 and S_2 be any two sets of nodes such that $S_1 \subseteq S_2 \subseteq V$. Monotonicity implies that $\sigma(S_1) \leq \sigma(S_2)$, i.e., adding more elements to the seed set cannot decrease the influence function $\sigma(\cdot)$. Submodularity implies that $\sigma(S_1 \cup \{v\}) - \sigma(S_1) \geq \sigma(S_2 \cup \{v\}) - \sigma(S_2)$, where $v \in V - S_2$, i.e., the marginal gain of a new node decreases as the seed set grows. Submodularity captures the property of diminishing returns.

Exactly evaluating $\sigma(S)$ is #P-complete [39]. In order to find the node with the largest marginal gain, Gim applies Monte Carlo sampling to repeatedly simulate the random choices during the diffusion process. Due to Monte Carlo sampling, the quality guarantee of Gim becomes $(1 - 1/e - \varepsilon)$, where the value of $\varepsilon > 0$ depends on the number of samples. The high computational cost of this step led to several algorithms that can be classified broadly into two types: (i) those that retain the approximation guarantee of Gim, but improve the running time in practice [19, 27], and (ii) those that ensure faster performance, but provide weaker (or no) approximation guarantees [12, 20, 39]. The Reverse Influence Sampling (RIS) method [5] attains the benefits of both categories by returning a $(1-1/e-\varepsilon)$-approximate solution with at least $1-|V|^{-l}$ probability, in $O(kl^2(|V| + |E|)log^2|V|/\varepsilon^3)$ time, where l is a tunable parameter that adjusts the trade-off between quality and running time.

[1] We use the terms graph/network and node/user interchangeably.

[2] The random thresholds capture the lack of knowledge about their values. In some approaches [3, 33] all thresholds are set to some fixed value, e.g., 0.5.

[3] For simplicity, in the rest of the paper we omit the diffusion model subscript and write $\sigma(S)$.

RIS is based on the concept of Reverse Reachable (RR) sets. An RR set for a node v represents the nodes that can influence v in some possible world X. In order to create an RR set for v, RIS first samples a possible world X using the live edge model. It then computes the nodes in X that can reach v by a depth-first-search from v, after reversing the direction of the edges (since RIS searches for nodes that can influence v, as opposed to those influenced by v). The nodes encountered constitute the RR set of v in X. The process is repeated multiple times with randomly sampled nodes, until a sufficient number of RRs has been constructed. The intuition is that if a seed set S is highly influential, members of S appear in the RR sets of numerous nodes. Motivated by this, RIS computes the set of k nodes that cover the maximum number of RR sets. The corresponding maximum coverage problem is solved using the standard approximate greedy method that iteratively adds nodes with the highest marginal gain on the number of covered RRs. As shown in [5], the fraction $F_R(S)$ of RR sets covered by a seed set S is an unbiased estimator of the influence; hence, RIS returns the seed set S^* as the solution of size k with the highest coverage $F_R(S^*)$ on the sampled RR sets.

Li et al. [29] apply RIS for weighted IM. Instead of creating RR sets of nodes chosen uniformly at random, they select each node v_i with probability $\frac{w_i}{W}$, where w_i is the weight of v_i and W is the sum of all weights. Intuitively, "heavy" nodes are sampled with high probability because they have large contribution to the total influence. After creating a sufficient number of RR sets, they solve the maximum coverage problem and return the seed set S^* with the highest coverage. They prove that $F_R(S) \cdot W$ is an unbiased estimator of $\mathbb{E}\left[\sum_{v_i \in I^X(S)} w_i\right]$. Thus, S^* also maximizes the weighted influence. Tang et al. [37, 38] utilize RIS with novel heuristics and statistics to reduce the samples required in order to maintain the $(1 - 1/e - \varepsilon)$ approximation. The D-SSA algorithm [34] combines RIS with the sampling method of [13], and uses a stop-and-stare strategy that terminates sampling when some quality guarantees are achieved. Huang et al. [24] improve on the work of [34], and perform an experimental analysis of IM algorithms. Another thorough experimental evaluation of IM methods can be found in [2].

2.2 Related Work on Competitive IM

Competitive IM typically assumes multiple competing agents that wish to maximize their local utility, and the objective is to reach a Nash equilibrium, i.e., a stable state where no agent has an incentive to deviate given the strategies of the rest. On the other hand, CWIM aims at maximizing the total influence of competing products. Monotone k-submodular function maximization [25, 35] also targets the total influence over multiple topics/products, which, however, do not compete for influence. Most models assume diffusion processes, where a node gets influenced by the first neighbor that activates it [1, 4, 7–9, 16, 23, 39]. This is usually *progressive*: once a node adopts the product, the decision is irreversible. Non-progressive processes have also been proposed [36].

Prior work on viral marketing [30] investigates the competitive setting from the perspective of a host, e.g., a social network hosting different companies. Each company specifies its budget, and the host aims at selecting seed sets that maximize the collective expected spread, while ensuring fairness. The model, termed K-LT, is inspired by the Weighted-Proportional competitive model [6]. K-LT assumes that an activated node chooses a product according to a proportional scheme, which depends on the products adopted by its neighbors in the previous time step. On the other hand, in the proposed ATI model a user is influenced by the product with the maximum similarity encountered during the awareness phase, without considering fairness. Due to the special properties of ATI, we are able to develop a game-theoretic algorithm with quality guarantees, whereas K-LT relies on heuristics. Furthermore, K-LT is limited to the linear threshold, whereas ATI can also employ independent cascade.

A 2-phase competitive model proposed by Goyal and Kearns [21] decomposes the influence dynamics into two parts. A switching phase, which dictates when a user can change from non-adoption to adoption, and a selection phase that specifies which product influences the user, conditional on switching. By considering a time-expanded graph, He and Kempe [22] prove that the Goyal-Kearns model is an instance of a general threshold model. They also show that the social welfare function is submodular; thus, the game is a utility game, which allows them to establish a price of anarchy of 2. The goal of the switching-selection model is to capture how user adoption depends on network effects, whereas ATI does not consider network effects for product adoption: a node can become aware of various products through its peers but whether it finally adopts the product does not depend on the decisions of its peers.

Most similar to ATI is the OR model [6] for competitive IM. OR is an extension of the conventional threshold model, in which awareness about each product diffuses independently. The actual influence/adoption occurs after the end of the diffusion process. Compared to OR, ATI is not limited to the threshold model. Moreover, while OR assumes general decision functions for the influence phase, in ATI a user is influenced by the product with the maximum similarity. This leads to a monotone utility game, a unique property of ATI that allows the development of a fast best-response algorithm with good quality guarantees.

3 PROBLEM AND MODEL DEFINITION

We consider an owner that wishes to advertise a set of competing products C to the users of a directed social graph $G = (V, E, P)$, under a competitive diffusion model $\mathcal{M}$. Each product $c_j \in C$ has a *budget* $k_j \in \mathbb{N}^+$ $(1 \leq k_j \leq |V|)$, which represents the number of users in its *seed set* S_j. Every user $v_i \in V$ has a weight $w_{i,j}$ $(0 \leq w_{i,j} \leq 1)$, for each $c_j \in C$, that corresponds to the *similarity* between v_i and c_j. For instance, if a user's v_i profile or demographic characteristics match the advertised product c_j, $w_{i,j}$ has a high value. The owner's target is to maximize the expected total similarity of the influenced users across all its products.

Collective Weighted Influence Maximization (CWIM) problem: given a graph G, a competitive diffusion model $\mathcal{M}$, a set of products C, a weight $w_{i,j}$ between each user v_i and product c_j, and $|C|$ integers $k_1, \ldots, k_{|C|}$, find the set of seed sets $\mathbb{S} = \{S_1, \ldots, S_j, \ldots, S_{|C|}\}$, with $|S_j| = k_j, 1 \leq j \leq |C|$, that maximize

Table 1: Abbreviations and Symbols

Symbol	Meaning
IC, LT, LE	Independent Cascade, Linear Threshold, Live Edge
RR set	Reverse Reachable set
NE	Pure Nash Equilibrium
C	Set of competing products $\{c_1, \ldots, c_j, \ldots, c_{\|C\|}\}$
S_j	Seed set of c_j
$k_j = \|S_j\|$	Budget of c_j
$w_{i,j}$	Similarity between user v_i and product c_j
$\mathbb{S}$	Set of seed sets $\{S_1, \ldots, S_j, \ldots, S_{\|C\|}\}$
$I_j^X(\mathbb{S})$	Set of users influenced by c_j in possible world X
$A_j^X(S_j)$	Set of users aware of c_j in possible world X
$\sigma(\mathbb{S})$	Total similarity (weight) of users in all influence sets
$\sigma_j(\mathbb{S})$	Individual utility of c_j
$\beta_j(\mathbb{S})$	Benefit to $\sigma(\mathbb{S})$ due to participation of c_j
$\delta_j(\mathbb{S})$	Divergence of c_j ($\sigma_j(\mathbb{S}) - \beta_j(\mathbb{S})$)
$AP(v, v_i, c_j)$	Awareness probability that user v informs v_i about c_j

the following objective function under $\mathcal{M}$:

$$\sigma(\mathbb{S}) = \mathbb{E}\left[\sum_{j=1}^{|C|} \sum_{v_i \in I_j^X(\mathbb{S})} w_{i,j}\right] \quad (1)$$

where $I_j^X(\mathbb{S})$ is the set of users influenced by c_j in possible world X.

We refer to the objective function $\sigma(\mathbb{S})$ as the *total similarity*, i.e., the expected total weight of users in all influence sets. For the simple case of a single product and assuming that all weights are equal to 1, CWIM reduces to the traditional IM problem, which is NP-hard [26]. Therefore, CWIM is NP-hard as well. In addition, since calculating the influence spread of a seed set S is #P-hard [10], computing $\sigma(\mathbb{S})$ is also #P-hard.

We cannot use directly IC or LT, which refer to a single product, as the underlying competitive diffusion model $\mathcal{M}$. Moreover, straightforward extensions of IC and LT to the competitive setting may not preserve monotonicity or submodularity. Consider, for instance, a social graph with two users v_1, v_2 connected by edge (v_1, v_2) of probability 1. Assume two products c_1, c_2, and weights $w_{1,1} = 1, w_{1,2} = 0.1, w_{2,1} = 1, w_{2,2} = 0.1$; e.g., both users are much more similar to c_1 than c_2. For this simple graph, assume a competitive diffusion process $\mathcal{M}^*$ defined as follows for node v_1 (similarly for v_2): (i) if v_1 is selected as a seed by product c_j, then v_1 adopts c_j (for simplicity, assume that a node cannot act as a seed for both products); (ii) if v_1 is not selected as a seed by any product but v_2 is a seed for c_j, then v_1 adopts c_j; (iii) else, if neither v_1 nor v_2 are seeds for any product, then they do not adopt any product. Under $\mathcal{M}^*$, if we only include v_1 in the seed set of c_1 ($S_1 = \{v_1\}, S_2 = \emptyset$), then v_1 will influence v_2, yielding $I(S_1) = \{v_1, v_2\}, I(S_2) = \emptyset$ and $\sigma(\{S_1, S_2\}) = w_{1,1} + w_{2,1} = 2$. On the other hand, if we set $S_1' = \{v_1\}, S_2' = \{v_2\}$, then based on the aforementioned diffusion process $\mathcal{M}^*$ we have $\sigma(\{S_1', S_2'\}) = w_{1,1} + w_{2,2} = 1.1$. Although $S_1 \subseteq S_1'$ and $S_2 \subseteq S_2'$, we have that $\sigma(\{S_1, S_2\}) > \sigma(\{S_1', S_2'\})$, which violates monotonicity. In an analogous way, we can construct examples of submodularity violations under $\mathcal{M}^*$. Next, we introduce a diffusion model for CWIM that preserves both properties. Table 1 contains the most frequent abbreviations and symbols.

3.1 Awareness-to-Influence Model

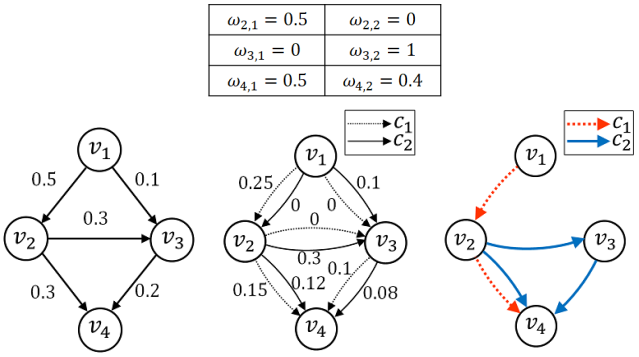
In many real-world applications users do not blindly follow the first influence; instead, they collect information, reproduce it and finally decide. Consequently, the proposed *Awareness-to-Influence* (ATI) is a 2-phase model consisting of an awareness and a subsequent influence phase. Specifically, initially for each product $c_j \in C$ a set S_j ($|S_j| = k_j$) of users are selected as its seeds and are informed about c_j. A user can act as seed of multiple products simultaneously; i.e., it is possible that $S_j \cap S_{j'} \neq \emptyset$ for pairs of products $c_j \neq c_{j'}$.

Awareness phase: In the terminology of ATI, when a node v is informed about a product c_j, it becomes *c_j-aware* (instead of *active* as in IC and LT). Then, it propagates information about c_j to each of its neighbors v_i with *awareness probability* $AP(v, v_i, c_j) = p(v, v_i) \cdot w_{i,j}$, so that a user v is more likely to notify about c_j friends connected through high-probability edges, and who are similar to c_j. ATI adopts the diffusion processes of IC and LT to model awareness. Specifically, in *Awareness-to-Influence Independent Cascade* (ATIIC), when a node becomes c_j-aware at time t, it informs every neighbor v_i about c_j, with awareness probability $AP(v, v_i, c_j)$, at time $t+1$. In *Awareness-to-Influence Linear Threshold* (ATILT), each node v_i has a threshold $\theta_{i,j}$ for every product c_j, chosen uniformly at random in the range [0,1]. Similar to [29], which applies LT to single-product weighted IM, we normalize the probabilities of the incoming edges of every user so that their sum equals 1. Node v_i becomes c_j-aware, when the awareness probabilities of its incoming edges about c_j exceed $\theta_{i,j}$:

$$\sum_{(v,v_i)\in E \wedge\ v:aware\ of\ c_j} p(v, v_i) \cdot w_{i,j} \geq \theta_{i,j}.$$

ATILE extends the live edge model to provide equivalent order-independent views of ATIIC and ATILT. We first create an augmented graph G' from G by replacing every edge (v, v_i) with $|C|$ edges from v to v_i. Each edge (v, v_i, c_j) in G' denotes that v may inform v_i about $c_j \in C$. For ATIIC, we generate a possible world X by tossing a biased coin, for every such edge, and keeping it with probability $AP(v, v_i, c_j)$. For ATILT, we keep at most one of the incoming c_j-edges to v_i with probability $AP(v, v_i, c_j)$ and no edge with probability $1 - \sum_{(v,v_i)\in E} AP(v, v_i, c_j)$. In any case, a kept edge (v, v_i, c_j) is marked as *c_j-live*. A *c_j-path* from a user v to another v_i contains only c_j-live edges, and it means that v_i becomes c_j-aware from v in X. The nodes that become c_j-aware are those reachable through a c_j-path from some node in S_j.

Figure 1 illustrates an example of ATIIC, considering two products c_1 and c_2. The weights of users v_2, v_3, v_4 for the two products are shown in the top table, e.g., v_2 has similarity 0.5 with c_1 and 0 with c_2. Weights for node v_1 are excluded because it has no incoming edges, and therefore it cannot be made aware of any product. Figure 1a shows the original graph G. The numbers next to the edges of G indicate their probabilities, e.g., $p(v_1, v_2) = 0.5$. The augmented graph G' in Figure 1b contains two edges for every edge of G (dotted lines correspond to c_1). Next to each edge is its awareness probability: e.g., $AP(v_1, v_2, c_1) = 0.25$ and $AP(v_1, v_2, c_2) = 0$. After tossing the coins, assume that the five edges of Figure 1c are marked as live, yielding three c_1-paths: $v_1 \rightarrow v_2$, $v_1 \rightarrow v_2 \rightarrow v_4$, $v_2 \rightarrow v_4$, and four c_2-paths: $v_2 \rightarrow v_4$, $v_2 \rightarrow v_3$, $v_2 \rightarrow v_3 \rightarrow v_4$, $v_3 \rightarrow v_4$. This



(a) Initial G **(b)** Augmented G' **(c)** *Live* edges

Figure 1: Augmented *Live Edge* model - AtIic

means that in this possible world, v_1 can make v_2 and v_4 aware of c_1, and no node aware of c_2.

In both AtIic and AtIlt the awareness propagation of c_j does not affect any other product $c_{j'} \neq c_j$. Accordingly, in AtIle there are $|C|$ independent diffusion processes, one for each product. Based on this observation, the equivalence between AtIle and AtIic (AtIlt) can be established using the equivalence of LE and IC (LT) ([26],[29]).

Influence phase: After the termination of the awareness phase, each node, having all the information available, makes its final decision. Obviously, users unaware of any product are not influenced. If a node is aware of multiple products, it is influenced by the one with the largest similarity, which corresponds to a *max rule*. If there are two or more products with the maximum similarity, it is influenced by the one that it first became aware of. In any case, the influence sets are disjoint, i.e., for any pair of products $c_j, c_{j'}$, it holds that $I_j^X(\mathbb{S}) \cap I_{j'}^X(\mathbb{S}) = \emptyset$. Observe that S_j is not necessarily a subset of $I_j^X(\mathbb{S})$, i.e., a user acting as a seed for some product may be influenced by another.

In an alternative influence scheme, a product could influence with a probability proportional to its similarity to the user. However, this *proportional rule* violates the monotonicity property. Assume for example a simple graph with a single user v and two products c_1 and c_2 with similarities to v equal to 0.9 and 0.1, respectively. If $S_1 = \{v\}$ and $S_2 = \emptyset$, then v is influenced by c_1 for a total similarity of 0.9. But if $S_1 = \{v\}$ and $S_2 = \{v\}$, then v is influenced by c_1 with probability 90% and by c_2 with probability 10% for a total similarity of $90\% \cdot 0.9 + 10\% \cdot 0.1 < 0.9$, which obviously violates monotonicity. In fact, this example demonstrates that in order to guarantee monotonicity, any product with similarity less than the maximum (among the products that a user is aware of) must be selected with zero probability.

In AtIle, among all the c_j-paths from v to v_i the one with the maximum similarity $w_{i,j}$ is called a *max live* path (there can be multiple max live paths). Such a path means that v_i is influenced by c_j in possible world X. Continuing the example of Figure 1 for AtIic, among the seven paths, only the c_2-path $v_2 \rightarrow v_4$ is not max live (Figure 1c). Assuming that $S_1 = \{v_1\}$ and $S_2 = \{v_2\}$, v_4 becomes aware of both c_1 and c_2, but it is influenced by c_1, with which it has higher similarity ($w_{4,1} = 0.5 > w_{4,2} = 0.4$).

Let $I_j^X(\mathbb{S})$ be the set of users influenced by some seed of S_j in possible world X. We define as $\sigma_j^X(\mathbb{S})$ the sum of weights of users in $I_j^X(\mathbb{S})$: $\sigma_j^X(\mathbb{S}) = \sum_{v_i \in I_j^X(\mathbb{S})} w_{i,j}$. The *total similarity* in X is $\sigma^X(\mathbb{S}) = \sum_{c_j \in C} \sigma_j^X(\mathbb{S})$. By summing over all possible worlds, we obtain: $\sigma(\mathbb{S}) = \sum_{\forall \text{world } X} Pr(X) \cdot \sigma^X(\mathbb{S}) = \sum_{\forall \text{world } X} Pr(X) \cdot \sum_{c_j \in C} \sum_{v_i \in I_j^X(\mathbb{S})} w_{i,j}$. Our aim is to maximize this expected value (see also Equation (1)).

3.2 Properties

In both AtIic and AtIlt, a user v_i is influenced by a product c_j in a possible world X, if and only if there is a *max live c_j-path* in X from a user in the seed set S_j to v_i. Observe that adding a new node v in S_j can only influence v_i towards c_j, but not towards other products. For instance, if before the addition v_i was influenced by $c_{j'}$ ($c_{j'} \neq c_j$), v_i may remain at $c_{j'}$ or may switch to c_j, but it cannot switch to another product c_k ($c_k \neq c_{j'}, c_k \neq c_j$) because the inclusion of v in S_j only generates new c_j-*paths* that do not alter the influence spread of other products. Lemma 1 proves that $\sigma(\cdot)$ is monotone, while Lemma 2 shows submobularity.

Lemma 1. *(Monotonicity) Let $\mathbb{S} = \{S_1, \ldots, S_j, \ldots, S_{|C|}\}$ and $\mathbb{S}' = \{S_1, \ldots, S_j \cup \{v\}, \ldots, S_{|C|}\}$ be two seed sets. Then, $\sigma(\mathbb{S}') \geq \sigma(\mathbb{S})$.*

Proof. The additional seed v in S_j can only influence users towards c_j, but not other products. Let V_1 be the set of users that are influenced by v and were not influenced by $\mathbb{S}$ in possible world X (i.e., $\forall v_i \in V_1, \forall c_j \in C : v_i \notin I_j^X(\mathbb{S})$). These users increase $\sigma_j^X(\mathbb{S}')$, and consequently $\sigma^X(\mathbb{S}')$ by $\Delta_1 = \sum_{v_i \in V_1} w_{i,j} \geq 0$, with respect to $\sigma^X(\mathbb{S})$. In addition, let V_2 be the set of users that were influenced by some product $c_{j'}$ in $\mathbb{S}$ ($c_{j'} \neq c_j$), and the addition of v changes their influence to c_j. A user v_i switches to c_j from its current product $c_{j'}$, if and only if $w_{i,j} \geq w_{i,j'}$. Accordingly, users in V_2 increase $\sigma^X(\mathbb{S}')$ by $\Delta_2 = \sum_{v_i \in V_2} (w_{i,j} - w_{i,j'}) \geq 0$. Since no other products are affected, V_1 and V_2 cover all cases of influence updates in $\mathbb{S}'$. Therefore: $\sigma^X(\mathbb{S}') = \sigma^X(\mathbb{S}) + \Delta_1 + \Delta_2 \geq \sigma^X(\mathbb{S})$. If we take the weighted sum over all possible worlds (we weigh each world X by its probability $Pr(X)$), it holds that $\sigma(\mathbb{S}') \geq \sigma(\mathbb{S})$, i.e., the total similarity cannot decrease by adding a user in a seed set. □

Lemma 2. *(Submodularity) Let $\mathbb{S} = \{S_1, \ldots, S_j, \ldots, S_{|C|}\}$ and $\mathbb{S}' = \{S'_1, \ldots, S'_j, \ldots, S'_{|C|}\}$ be two seed sets such that $S_j \subseteq S'_j, \forall\, 1 \leq j \leq |C|$. The marginal gain of adding a user v to a seed set S'_j in $\mathbb{S}'$ is at most as large as adding the same user to the corresponding seed set S_j in $\mathbb{S}$; i.e $\sigma(S'_1, \ldots, S'_j \cup \{v\}, \ldots, S'_{|C|}) - \sigma(\mathbb{S}') \leq \sigma(S_1, \ldots, S_j \cup \{v\}, \ldots, S_{|C|}) - \sigma(\mathbb{S})$, where $v \in V - S'_j$.*

Proof. A positive marginal gain occurs when the additional seed v causes some user to switch to c_j. We will show that any such switch in $\mathbb{S}'$ must also occur in $\mathbb{S}$. Assume, by contradiction, that adding v to S'_j in a possible world X causes a user v_i to switch to c_j, but adding v to S_j does not. This can only happen if v generates a *max live c_j-path* from a user v' in the seed set S'_j to v_i. There are two cases: (i) v is the initial user in the c_j-*path* ($v \rightarrow v' \rightarrow v_i$), or (ii) v is an intermediate user ($v' \rightarrow v \rightarrow v_i$). In

the first case, v' would influence v_i before the inclusion of v in S'_j, and v has no effect and zero gain. Case (ii) implies that there is *max live* c_j*-path* $v \rightarrow v_i$; accordingly, adding v to S_j would result in the same *max live* c_j*-path* $v \rightarrow v_i$ and thus affect v_i, generating the same marginal gain as in S'_j. Both cases contradict the assumptions, implying that $\sigma^X(S'_1, \ldots, S'_j \cup \{v\}, \ldots, S'_{|C|}) - \sigma^X(\mathbb{S}') \leq \sigma^X(S_1, \ldots, S_j \cup \{v\}, \ldots, S_{|C|}) - \sigma^X(\mathbb{S})$. By taking a weighted sum over all possible worlds as in Lemma 1, we conclude that the function $\sigma(\cdot)$ is submodular. □

3.3 Baseline Algorithm

Given the independent awareness processes, we can design a naïve algorithm for the CWIM problem as follows. Each product c_j selects the top-k_j seeds, where k_j is its budget, without taking into account the other awareness and influence processes. Essentially, each product seed set is chosen, as if there were no other products. Even though this baseline is reasonable, it does not enjoy quality guarantees. To see why, consider the simple example of a graph consisting of k disconnected (isolated) nodes, and k products of unit budget such that $w_{i,j} = w_i, \forall j$, i.e., all products have the same weight w_i for any given node v_i. Furthermore, assume without loss of generality that $w_1 \geq \cdots \geq w_k$. The optimal solution occurs when each product has a distinct node as its seed, with total similarity $\sum w_i$. On the other hand, the baseline algorithm will assign node v_1 as the seed for all k products, with total similarity w_1. Depending on the values of $w_1, \ldots, w_k$, the ratio between the optimal solution and the naïve one can thus become arbitrarily close to k. In the following, we propose a game-theoretic algorithm, which utilizes monotonicity and submodularity to achieve a fixed approximation ratio that is independent of the number of competitors.

4 GAME-THEORETIC FRAMEWORK

We apply a game-theoretic approach to CWIM because, as discussed in Section 4.2, a single round of best-response dynamics reaches a solution with high quality. A *game* in strategic form is the ordered triple $\mathcal{G} = (C, (\mathcal{S}_j)_{j\in\{1,\ldots,|C|\}}, (\sigma_j)_{j\in\{1,\ldots,|C|\}})$, where:

- $C = \{c_1, \ldots, c_{|C|}\}$ is the finite set of players.
- $\mathcal{S}_j$ is a finite non-empty strategy set available to player c_j. We denote the strategic space, i.e., the set of all vectors of pure strategies as $\mathcal{S} = \times_{j\in\{1,\ldots,|C|\}}\mathcal{S}_j = \mathcal{S}_1 \times \cdots \times \mathcal{S}_{|C|}$. A strategy vector $\mathbb{S} \in \mathcal{S}$ has the form $\mathbb{S} = (S_1, \ldots, S_{|C|})$, where $S_j \in \mathcal{S}_j \; \forall j \in \{1, \ldots, |C|\}$.
- $\sigma_j : \mathcal{S} \rightarrow \mathbb{R}$ is the individual utility of c_j, i.e., a function associating each vector of strategies $S \in \mathcal{S}$ with a utility for player c_j.

In *best response dynamics*, each player selects the strategy that maximizes its individual utility, given the current strategies of the other players. A *round* of best responses consists of all players choosing strategies sequentially exactly once. A game converges to a *pure Nash Equilibrium* (NE), when no player can increase its utility by playing deterministic best response[4]. The *social welfare*[5] $\sigma : \mathcal{S} \rightarrow \mathbb{R}$ represents the total utility for any strategy vector. The *social optimum* is the solution with the highest social welfare. The *price of anarchy* is the ratio between the social welfare of the social optimum and that of the worst possible NE; i.e., it is an upper bound on the number of times that the discovered solution (when best response dynamics converge) is worse than the optimal one.

Given $\mathbb{S} = \{S_1, \ldots, S_{|C|}\}$, $\mathbb{S} \oplus S'_j$ is the vector obtained if player c_j switches strategy from S_j to S'_j, i.e., $\mathbb{S} \oplus S'_j = \{S_1, \ldots, S_{j-1}, S'_j, S_{j+1}, \ldots, S_{|C|}\}$. The union of $\mathbb{S}$ and $\mathbb{S}'$ is defined as $\mathbb{S} \cup \mathbb{S}' = \{S_1 \cup S'_1, \ldots, S_{|C|} \cup S'_{|C|}\}$, i.e., the strategy vector where the strategy of each player is the union of its strategies in $\mathbb{S}$ and $\mathbb{S}'$. $\mathbb{S} \cup T_j$ represents the strategy vector where only c_j changes strategy from S_j to $S_j \cup T_j$. We denote by $\emptyset_j$ the null strategy for player c_j; $\mathbb{S} \oplus \emptyset_j$ implies that c_j drops out of the game. It trivially holds that $(\mathbb{S} \oplus \emptyset_j) \cup S_j = \mathbb{S}$. The quantity $\beta_j(\mathbb{S}) = \sigma(\mathbb{S}) - \sigma(\mathbb{S} \oplus \emptyset_j)$ denotes the *benefit* to the social welfare due to the participation of player c_j in the strategy vector $\mathbb{S}$. An interesting class concerns *utility* games [31, 41]:

Definition 1. *(Utility game) A game $\mathcal{G}$ with social welfare function σ is a utility game if the following three conditions hold:*

1. *The social welfare function $\sigma(\cdot)$ is submodular.*
2. *The sum of individual utility functions does not exceed the social welfare, i.e, $\sum_{j\in\{1,\ldots,|C|\}} \sigma_j(\mathbb{S}) \leq \sigma(\mathbb{S})$.*
3. *The individual utility of each player c_j is at least its benefit to the social welfare, i.e., $\sigma_j(\mathbb{S}) \geq \beta_j(\mathbb{S}), \forall j \in \{1, \ldots, |C|\}$.*

Basic games constitute a sub-class of utility games.

Definition 2. *(Basic utility game) A utility game is basic, if it additionally satisfies the two conditions:*

1. *The social welfare function $\sigma(\cdot)$ is monotone.*
2. *Condition 3 in Definition 1 is satisfied with equality, i.e., $\sigma_j(\mathbb{S}) = \beta_j(\mathbb{S}), \forall j \in \{1, \ldots, |C|\}$. We refer to this condition as the basicness property.*

Monotone utility games do not always accept a NE, but when they do the *price of anarchy* is at most 2 [41]; i.e., the social welfare of the equilibrium is at least half of the optimal. On the other hand, in basic utility games, a NE always exists and best response dynamics is guaranteed to converge to a NE in a finite number of rounds. Moreover, starting from empty strategies, a single round of best responses yields a solution with social welfare at least half the social optimum [17, 31], i.e., it matches the theoretical bound of 2 for the price of anarchy. The fact that we have quality guarantees after a round, without reaching a NE (which may take numerous rounds), renders basic utility games very attractive in computational terms, especially when each best response is costly. Next we elaborate on the connection between CWIM and (basic) utility games.

4.1 Utility Game for CWIM

We introduce the game GCW for CWIM and apply best response dynamics, where each product $c_j \in C$ constitutes a player. A seed set S_j ($|S_j| = k_j$) corresponds to a *strategy* of c_j. The individual utility of S_j for c_j is $\sigma_j(\mathbb{S}) = \mathbb{E}\Big[\sum_{v_i \in I_j^X(\mathbb{S})} w_{i,j}\Big]$. The goal of every player is to maximize its individual utility, i.e., the expected total

[4]It is possible to define Nash equilibria in mixed (probabilistic) strategies but we focus exclusively on pure Nash equilibria, as is common in the literature of competitive diffusion [1, 4, 15, 22, 39].

[5]The terms *social welfare*, *collective influence* and *total influence* refer to the same concept.

weight of the users it influences, given the seed sets of the products. The *social welfare* $\sigma(\mathbb{S})$ is the total weight from all influenced users $\sigma(\mathbb{S}) = \sum_{j=1}^{|C|} \sigma_j(\mathbb{S})$, i.e., the objective function of CWIM.

THEOREM 1. *GCW is a monotone utility game.*

PROOF. According to Lemma 1 the social welfare function $\sigma(\cdot)$ is monotone. Regarding the three conditions of Definition 1:

1. By Lemma 2, the social welfare function $\sigma(\cdot)$ is submodular.

2. The social welfare in GCW is defined as $\sigma(\mathbb{S}) = \sum_{j=1}^{|C|} \sigma_j(\mathbb{S})$. Thus, Condition 2 of Definition 1 holds as equality.

3. In any possible world X, a new product c_j may influence two types of users: (i) V_1 is the set of users that are not aware of (and, thus, they were not influenced by) any other product, and (ii) V_2 is the set of users that were influenced by some product $c_{j'}$ with weight $w_{i,j'} < w_{i,j}$. Users in V_1 increase the individual utility of c_j and the social welfare by the same value $\sum_{v_i \in V_1} w_{i,j}$. Users in V_2, however, add $\sum_{v_i \in V_2} w_{i,j}$ to $\sigma_j^X(\mathbb{S})$ and only $\sum_{v_i \in V_2} (w_{i,j} - w_{i,j'})$ to $\sigma^X(\mathbb{S})$. Thus, $\sigma_j^X(\mathbb{S}) = \sum_{v_i \in V_1} w_{i,j} + \sum_{v_i \in V_2} w_{i,j} \geq \sum_{v_i \in V_1} w_{i,j} + \sum_{v_i \in V_2} (w_{i,j} - w_{i,j'}) = \beta_j^X(\mathbb{S})$. Condition 3 of Definition 1 then holds by taking the weighted sum over all possible worlds. □

We define as $\mu_{T_j}(\mathbb{S}) = \sigma(\mathbb{S} \cup T_j) - \sigma(\mathbb{S})$ the *marginal gain* to the social welfare when the set of nodes T_j is added to the seed set S_j of c_j, or equivalently, when c_j changes strategy from S_j to $S_j \cup T_j$. The *benefit* constitutes a special case[6] of the *marginal gain* where S_j is empty: $\beta_j(\mathbb{S}) = \mu_{S_j}(\mathbb{S} \oplus \emptyset_j)$. We refer to the *divergence* $\delta_j(\mathbb{S})$ as the difference between the individual utility of c_j and the benefit of c_j to the social welfare: $\delta_j(\mathbb{S}) = \sigma_j(\mathbb{S}) - \beta_j(\mathbb{S}) = \sigma_j(\mathbb{S}) - \mu_{S_j}(\mathbb{S} \oplus \emptyset_j)$. As shown in the proof of Theorem 1, in any possible world X, $\delta_j(\mathbb{S}) = \sum_{v_i \in V_2} w_{i,j'}$, where V_2 is the set of nodes influenced by other products $c_{j'}$ in possible world X before the participation of c_j in the game. Due to the existence of such nodes, $\delta_j(\mathbb{S}) \geq 0$ and *GCW is not a basic game.*

Next, we argue that the best response of player c_j corresponds to a weighted influence maximization problem. Recall that during its best response, the goal of c_j is to find the seed set S_j that maximizes the expected value of the total weight of influenced nodes. Due to the interactions though, we must consider the effect of the other players. Concretely, c_j influences node v_i if two conditions hold: (i) v_i becomes c_j-aware, and (ii) v_i is not already aware of another product $c_{j'}$ with higher or equal similarity. Condition (i) is similar to traditional IM. To account for condition (ii), we modify the weight $w_{i,j}$ by multiplying it by the probability that v_i is unaware of some other product $c_{j'}$ with $j' < j$ and $w_{i,j'} \geq w_{i,j}$. Its weight thus becomes $w_{i,j} \cdot \overline{AP}(v_i)$, where $\overline{AP}(v_i)$ denotes the probability of condition (ii). At the end, player c_j aims at maximizing the quantity:

$$\mathbb{E}\left[\sum_{v_i \in A_j^X(S_j)} w_{i,j} \cdot \overline{AP}(v_i)\right]. \tag{2}$$

[6]In the rest of the paper, we express the benefit in terms of the more general concept of marginal gain.

The terms $\overline{AP}(v_i)$ are scaling constants that do not depend on S_j. (2) is a standard single-product IM formulation where the awareness probabilities from v to v_i are $AP(v, v_i, c_j)$ while the weights $w_{i,j}$ are changed to $w_{i,j} \cdot \overline{AP}(v_i)$. This can be solved using GIM, RIS or D-SSA, so that each best response is within $(1 - \frac{1}{e})$ of the optimum[7]. If we reach a stable state, that state corresponds to a $(1 - \frac{1}{e})$-NE.

Our next result establishes that the price of anarchy at any such $(1 - \frac{1}{e})$-Nash equilibrium is only slightly higher than 2.5. But first, we introduce Lemma 3 which describes the property of diminishing marginal gains.

LEMMA 3. *For the social welfare function σ of the GCW game, we have:*

$$\mu_{T_j}(\mathbb{S}) \geq \mu_{T_j}(\mathbb{S}'),$$

where $\mathbb{S} = \{S_1, \ldots, S_{|C|}\}, \mathbb{S}' = \{S'_1, \ldots, S'_{|C|}\}$ are two strategy vectors with $S_j \subseteq S'_j \; \forall j \in \{1, \ldots, |C|\}$, and $T_j \subseteq V$ is any set of nodes.

PROOF. Assume T_j consists of a a single node v. We distinguish between three cases. (i) If $v \in V - S'_j$ (hence, $v \in V - S_j$), then the inequality in the Lemma follows directly by the submodularity property. (ii) If $v \in S_j$ (hence, $v \in S'_j$), then the inequality holds trivially as $\mu_{T_j}(\mathbb{S}) = \mu_{T_j}(\mathbb{S}') = 0$. Finally, (iii) if $v \notin S_j$ but $v \in S'_j$, then $\mu_{T_j}(\mathbb{S}) \geq 0$ by monotonicity, whereas $\mu_{T_j}(\mathbb{S}') = 0$. Thus, $\mu_{T_j}(\mathbb{S}) \geq \mu_{T_j}(\mathbb{S}')$. If T_j contains multiple nodes, we add them sequentially, and apply the above argument in each step. □

THEOREM 2. *In GCW, the price of anarchy of a pure $(1 - \frac{1}{e})$-Nash equilibrium is upper bounded by $1 + \frac{1}{1 - \frac{1}{e}} \approx 2.582$.*

PROOF. Let $\mathbb{S}^*$ be a pure $(1 - \frac{1}{e})$-Nash equilibrium, and $\mathbb{O} = \{O_1, \ldots, O_{|C|}\}$ the set of strategies at the social optimum. Due to monotonicity, it holds that $\sigma(\mathbb{O}) \leq \sigma(\mathbb{S}^* \cup \mathbb{O})$. Let $\mathbb{O}^j$ denote the strategies selected by the first j players in the social optimum $\mathbb{O}$, with players $c_{j+1}, \ldots, c_{|C|}$ playing the empty strategy. We then have $\mathbb{O}^0 = \{\emptyset_1, \ldots, \emptyset_{|C|}\}, \mathbb{O}^1 = \{O_1, \emptyset_2, \ldots, \emptyset_{|C|}\}, \ldots, \mathbb{O}^{|C|} = \{O_1, \ldots, O_{|C|}\}$.
We next write:

$$\sigma(\mathbb{O}) - \sigma(\mathbb{S}^*) \leq \sigma(\mathbb{S}^* \cup \mathbb{O}) - \sigma(\mathbb{S}^*) =$$
$$= \sum_{j=1}^{|C|} [\sigma(\mathbb{S}^* \cup \mathbb{O}^j) - \sigma(\mathbb{S}^* \cup \mathbb{O}^{j-1})] = \sum_{j=1}^{|C|} \mu_{O_j}(\mathbb{S}^* \cup \mathbb{O}^{j-1})$$

By Lemma 3 we have:

$$\mu_{O_j}(\mathbb{S}^* \cup \mathbb{O}^{j-1}) \leq \mu_{O_j}(\mathbb{S}^* \oplus \emptyset_j),$$

for any player c_j. By Property 3 of Definition 1 for utility games we have:

$$\mu_{O_j}(\mathbb{S}^* \oplus \emptyset_j) \leq \sigma_j(\mathbb{S}^* \oplus O_j).$$

Furthermore, given $\mathbb{S}^*$ is an $(1 - \frac{1}{e})$-Nash equilibrium we have that:

$$(1 - \frac{1}{e}) \cdot \sigma_j(\mathbb{S}^* \oplus O_j) \leq \sigma_j(\mathbb{S}^*).$$

[7]In reality, the algorithms return a $(1 - \frac{1}{e} - \varepsilon)$-approximate solution, but for simplicity we ignore ε in the bounds by setting it to 0.

Combining the above:

$$\sigma(\mathbb{O}) - \sigma(\mathbb{S}^*) \leq \frac{1}{1-\frac{1}{e}} \cdot \sum_{j=1}^{|C|} \sigma_j(\mathbb{S}^*).$$

Given that $\sum_{j=1}^{|C|} \sigma_j(\mathbb{S}^*) = \sigma(\mathbb{S}^*)$, we finally obtain:

$$\sigma(\mathbb{O}) \leq \left(1 + \frac{1}{1-\frac{1}{e}}\right) \cdot \sigma(\mathbb{S}^*) \approx 2.582 \cdot \sigma(\mathbb{S}^*).$$

□

The above bound suggests that if we reach an approximate NE, then we can be sure that this point is close to the social optimum. Given GCW is not a basic game, the existence of a NE is not guaranteed and we may not converge to a NE. In general, under standard competitive diffusion models a NE only exists for graphs with specific structure [1, 4, 15, 22, 39]. In fact, even deciding whether a NE exists is an NP-hard problem in many competitive models [14]. Nevertheless, as we show next, due to the game structure, a single round of best response strategies can result in a strategy vector with a social welfare that is close to the social optimum. Furthermore, since our objective is the social welfare, it is not important for our work whether we reach a NE or not. Interestingly, our model exhibits both submodularity and monotonicity, which is not true for general competitive diffusion models [6].

4.2 Single-Round Error Bound

Our methodology is inspired by a fundamental result in basic utility games, which states that a single full round of exact best responses can produce a solution with provably high social welfare [17, 31]. Although GCW is not a basic game, and players can only play approximate best response strategies, we will show that a single round of best responses can produce a strategy vector whose deviation from the optimal solution is bounded.

Theorem 3. *In GCW, assuming the players start with empty strategies, the social value at the end of a full round of best responses is at least* $\frac{e-1}{2e-1} \approx 0.387$ *of the optimal social value, minus a term that depends on the extent to which the game violates the basicness property.*

Proof. Without loss of generality, we assume the players play best response strategies in the order $c_1, \ldots, c_{|C|}$. Let $\mathbb{O} = \{O_1, \ldots, O_{|C|}\}$ the set of strategies at the social optimum. Let $\emptyset = \{\emptyset_1, \ldots, \emptyset_{|C|}\}$ and $\mathbb{S} = \{S_1, \ldots, S_{|C|}\}$ be the initial and final states in the full round, respectively. Furthermore, we denote by $\mathbb{S}^j$ the set of strategies for the $|C|$ products after the first j best responses, i.e., $\mathbb{S}^j = \{S_1, \ldots, S_j, \emptyset_j, \ldots, \emptyset_{|C|}\}$. Obviously, $\mathbb{S}^0 = \emptyset$ and $\mathbb{S}^{|C|} = S$. We also define $\mathbb{O}^j$ as in Theorem 2. Each player selects its seed set using $(1-\frac{1}{e})$-approximate best-response with respect to the seed sets of previous players. Repeating over all players we have:

$$\sum_{j=1}^{|C|} \sigma_j(\mathbb{S}^j) \geq (1-\frac{1}{e}) \cdot \sum_{j=1}^{|C|} \sigma_j(\mathbb{S}^j \oplus O_j)$$
$$= (1-\frac{1}{e}) \cdot \sum_{j=1}^{|C|} \sigma_j(\mathbb{S}^{j-1} \oplus O_j). \quad (3)$$

The divergence $\delta_j(\mathbb{S}^{j-1} \oplus O_j) \geq 0$ corresponds to the extent to which the best response of player c_j violates the basicness property at strategy vector $\mathbb{S}^{j-1} \oplus O_j$, i.e., $\delta_j(\mathbb{S}^{j-1} \oplus O_j) = \sigma_j(\mathbb{S}^{j-1} \oplus O_j) - \mu_{O_j}(\mathbb{S}^{j-1} \oplus \emptyset_j)$. By summing over all players:

$$\sum_{j=1}^{|C|} \sigma_j(\mathbb{S}^{j-1} \oplus O_j) = \sum_{j=1}^{|C|} [\mu_{O_j}(\mathbb{S}^{j-1} \oplus \emptyset_j) + \delta_j(\mathbb{S}^{j-1} \oplus O_j)]. \quad (4)$$

By combining (3) and (4) we obtain:

$$\sum_{j=1}^{|C|} \sigma_j(\mathbb{S}^j) \geq (1-\frac{1}{e}) \cdot \sum_{j=1}^{|C|} [\mu_{O_j}(\mathbb{S}^{j-1} \oplus \emptyset_j) + \delta_j(\mathbb{S}^{j-1} \oplus O_j)]$$
$$= (1-\frac{1}{e}) \cdot \sum_{j=1}^{|C|} [\mu_{O_j}(\mathbb{S}^{j-1}) + \delta_j(\mathbb{S}^{j-1} \oplus O_j)]. \quad (5)$$

By Lemma 3 we have:

$$(1-\frac{1}{e}) \cdot \sum_{j=1}^{|C|} [\mu_{O_j}(\mathbb{S}^{j-1}) + \delta_j(\mathbb{S}^{j-1} \oplus O_j)] \geq$$
$$(1-\frac{1}{e}) \cdot \Big(\sum_{j=1}^{|C|} \mu_{O_j}(\mathbb{S} \cup \mathbb{O}^{j-1}) + \delta_j(\mathbb{S}^{j-1} \oplus O_j) \Big). \quad (6)$$

We define the non-negative two-variate function $\Delta(\mathbb{S}, \mathbb{O}) \geq 0$ as the total divergence of all players:

$$\Delta(\mathbb{S}, \mathbb{O}) = \sum_{j=1}^{|C|} \delta_j(\mathbb{S}^{j-1} \oplus O_j). \quad (7)$$

Using (6) and (7), and substituting the terms $\mu_{O_j}(\mathbb{S} \cup \mathbb{O}^{j-1})$ in (6) by the definition of the marginal gain, we can rewrite (5) as:

$$\sum_{j=1}^{|C|} \sigma_j(\mathbb{S}^j) \geq (1-\frac{1}{e}) \cdot (\sigma(\mathbb{S} \cup \mathbb{O}) - \sigma(\mathbb{S}) + \Delta(\mathbb{S}, \mathbb{O})). \quad (8)$$

Furthermore, similar to above we can rewrite the term $\sum_{j=1}^{|C|} \sigma_j(\mathbb{S}^j)$ as follows (the term $\Delta(\mathbb{S}, \mathbb{S})$ in (9) is defined as in Equation (7) if we replace the strategy vector $\mathbb{O}$ by $\mathbb{S}$):

$$\sum_{j=1}^{|C|} \sigma_j(\mathbb{S}^j) = \sum_{j=1}^{|C|} [\mu_{S_j}(\mathbb{S}^{j-1} \oplus \emptyset_j) + \delta_j(\mathbb{S}^j)] = \sigma(\mathbb{S}) + \Delta(\mathbb{S}, \mathbb{S}). \quad (9)$$

By combining (8) and (9):

$$\sigma(\mathbb{S}) = \sum_{j=1}^{|C|} \sigma_j(\mathbb{S}^j) - \Delta(\mathbb{S}, \mathbb{S})$$
$$\geq (1-\frac{1}{e}) \cdot (\sigma(\mathbb{O}) - \sigma(\mathbb{S}) + \Delta(\mathbb{S}, \mathbb{O})) - \Delta(\mathbb{S}, \mathbb{S}),$$

or equivalently:

$$\sigma(\mathbb{S}) \geq \frac{e-1}{2e-1} \cdot \sigma(\mathbb{O}) - \frac{e \cdot \Delta(\mathbb{S}, \mathbb{S}) - (e-1) \cdot \Delta(\mathbb{S}, \mathbb{O})}{2e-1}$$
$$\geq \frac{e-1}{2e-1} \cdot \sigma(\mathbb{O}) - \frac{e}{2e-1} \cdot \Delta(\mathbb{S}, \mathbb{S}).$$

Hence, the social value at the end of a full round of best responses $\sigma(\mathbb{S})$ is at least $\frac{e-1}{2e-1} \approx 0.387$ of the optimal social value $\sigma(\mathbb{O})$, minus

a term containing $\Delta(\mathbb{S}, \mathbb{S})$ that depends on the extent to which the game violates the basicness property. □

The term $\frac{e}{2e-1} \cdot \Delta(\mathbb{S}, \mathbb{S})$ impacts the quality of the bound: the lower the term, the better the bound. In this direction, we define the following ratio (assuming $\sigma(\mathbb{S}) \neq 0$):

$$\rho(\mathbb{S}) = \frac{\Delta(\mathbb{S}, \mathbb{S})}{\sigma(\mathbb{S})} \geq 0.$$

Ideally, the term $\rho(\mathbb{S})$ is 0, when the game is basic.

Corollary 1. *The bound of Theorem 3 can be rewritten equivalently as:*

$$\sigma(\mathbb{S}) \geq \frac{e-1}{2e-1+e \cdot \rho(\mathbb{S})} \cdot \sigma(\mathbb{O}).$$

Proof. The proof follows directly from Theorem 3 and the definition of $\rho(\mathbb{S})$. □

We refer to $b = \frac{e-1}{2e-1+e \cdot \rho(\mathbb{S})}$ as the approximation *lower bound* of GCW. If basicness holds, the best-response framework achieves a constant-ratio lower bound of $b^* = (e-1)/(2e-1) \approx 0.387$. On the other hand, lacking basicness implies that $\rho(\mathbb{S}) > 0$, which decreases the quality of the lower bound. Our extensive experimental evaluation on various data sets in Section 6.3 shows that b is in practice at least 0.2 and can even reach values close to b^*. The lower bound ignores the positive contribution from the term $\frac{e-1}{2e-1} \cdot \Delta(\mathbb{S}, \mathbb{O})$; so the actual approximation quality can be higher. Although approximation bounds for basic utility games have been investigated before [17, 31], this is the first work to derive a lower bound on the approximation quality in the absence of basicness.

5 GAME-THEORETIC ALGORITHM

Based on the previous discussion, GCW involves a single round of best responses, where each competitor c_j selects the seed set S_j that maximizes its individual utility, using some polynomial algorithm. The complication lies in the fact that S_j must take into account the current seed sets of the other products. Algorithm 1 shows a concrete instantiation of GCW based on Reverse Reachable (RR) sets. Initially, each player/product c_j is assigned an empty seed set (Line 2). GCW proceeds in one full round of best responses assuming that players play in some random order $c_1, \ldots, c_{|C|}$. Recall from Theorem 3 that $\mathbb{S}^j = \{S_1, \ldots, S_j, \emptyset_{j+1}, \ldots, \emptyset_{|C|}\}$, $0 \leq j \leq |C|$, is the strategy vector after the first j best responses. Competitor c_j selects a strategy S_j to maximize its individual utility, given $\mathbb{S}^{j-1}$ (Lines 3-6). For this purpose, given the current seed sets $\{S_1, \ldots, S_{j-1}\}$, we create a number of RR sets for c_j (Line 4). Next, by applying an approximate algorithm [40] for the *maximum coverage problem*, it finds the k_j users that cover the largest number of RR sets (Line 5).

Recall from Section 2.1 that for single-product weighted IM, $F_R(S) \cdot W$ is an unbiased estimator of the influence $\mathbb{E}\Big[\sum_{v_i \in I^X(S)} w_i\Big]$, where $F_R(S)$ is the fraction of RR sets covered by S, and W is the sum of user weights [29]. For CWIM, the existence of products necessitates an unbiased estimator for the influence $\mathbb{E}\Big[\sum_{v_i \in I_j^X(\mathbb{S}^j)} w_{i,j}\Big]$ for c_j that takes into account the current seed sets of the previous products $c_1, \ldots, c_{j-1}$. Consider product $c_{j'}$, $j' < j$, with $w_{i,j'} \geq w_{i,j}$. If user v_i becomes c_j-aware while it is already aware of $c_{j'}$, v_i will

Algorithm 1 : GCW (graph G, budget array $k(1\text{x}|C|)$, weight array $w(|V|\text{x}|C|)$)

1: **for** each product c_j
2: assign an empty seed set S_j
3: **for** $c_j := c_1$ **to** $c_{|C|}$
4: create RR sets for c_j given the current seed sets $\{S_1, \ldots S_{j-1}\}$ of products $c_1, \ldots, c_{j-1}$
5: Find the seed set S_j^* of k_j users that covers the largest number of RR sets
6: Set $S_j \leftarrow S_j^*$
7: Output $\mathbb{S} = \{S_1, .., S_j, .., S_{|C|}\}$

remain influenced by $c_{j'}$ rather than c_j. Intuitively, creating an RR set for v_i is not beneficial for computing a good seed set S_j for c_j because v_i does not contribute to $I_j^X(\mathbb{S}^j)$.

Let $AP(v_i, c_{j'}|S_{j'})$ be the probability that v_i is aware of $c_{j'}$, given strategy $S_{j'}$. We define as

$$\overline{AP}(v_i|\mathbb{S}^{j-1}) = \prod_{j'<j \,|\, w_{i,j'} \geq w_{i,j}} (1 - AP(v_i, c_{j'}|S_{j'})), \tag{10}$$

the probability that, immediately before the creation of seed set S_j, v_i is unaware of any product $c_{j'}$, with $j' < j$ and $w_{i,j'} \geq w_{i,j}$. The probability that c_j influences user v_i given $\mathbb{S}^j$ is then:

$$IP(v_i, c_j|\mathbb{S}^j) = \overline{AP}(v_i|\mathbb{S}^{j-1}) \cdot AP(v_i, c_j|S_j), \tag{11}$$

i.e., v_i must be c_j-aware, but unaware of any product with higher similarity. Note that in each best response j we only need to compute and store $AP(v_i, c_j|S_j)$ for each $v_i \in V$. We can then trivially compute (10) by using the stored awareness probabilities $AP(v_i, c_{j'}|S_{j'})$, $1 \leq j' < j$ from the previous best responses.

In order to create the RR sets for c_j, we sample user v_i with probability $\frac{w_{i,j}}{W_j} \cdot \overline{AP}(v_i|\mathbb{S}^{j-1})$, where $W_j = \sum_{v_i \in V} w_{i,j} \cdot \overline{AP}(v_i|\mathbb{S}^{j-1})$ is a normalization constant. Let $F_{R,j}(\mathbb{S}^j)$ be the fraction of RR sets created for product c_j that are covered by S_j, given $\mathbb{S}^j$. Proposition 1 proves that $F_{R,j}(\mathbb{S}^j) \cdot W_j$ is an unbiased estimator of $\mathbb{E}\Big[\sum_{v_i \in I_j^X(\mathbb{S}^j)} w_{i,j}\Big]$. Therefore, the seed set S_j^* of cardinality k_j with the maximum $F_{R,j}(\mathbb{S}^{j-1} \cup S_j^*)$ becomes the current seed set of c_j.

Proposition 1. *During the best response of product c_j, $F_{R,j}(\mathbb{S}^j) \cdot W_j$ is an unbiased estimator of $\mathbb{E}\Big[\sum_{v_i \in I_j^X(\mathbb{S}^j)} w_{i,j}\Big]$.*

Proof. It holds that the expected value of the total similarity for product c_j equals:

$$\mathbb{E}\Big[\sum_{v_i \in I_j^X(\mathbb{S}^j)} w_{i,j}\Big] = \sum_{v_i \in V} IP(v_i, c_j|\mathbb{S}^j) \cdot w_{i,j}$$

For product $c_j \in C$, we sample a user v_i with probability $\frac{w_{i,j}}{W_j} \cdot \overline{AP}(v_i|\mathbb{S}^{j-1})$ and create a set $RR_{i,j}$. It follows that:

$$\mathbb{E}[F_{R,j}(\mathbb{S}^j)] = \sum_{v_i \in V} \frac{w_{i,j}}{W_j} \cdot \overline{AP}(v_i|\mathbb{S}^{j-1}) \cdot Pr(RR_{i,j} \cap S_j \neq \emptyset)$$
$$= \sum_{v_i \in V} \frac{w_{i,j}}{W_j} \cdot \overline{AP}(v_i|\mathbb{S}^{j-1}) \cdot AP(v_i, c_j|S_j)$$
$$= \sum_{v_i \in V} \frac{w_{i,j}}{W_j} \cdot IP(v_i, c_j|\mathbb{S}^j) = \frac{\mathbb{E}\Big[\sum_{v_i \in I_j^X(\mathbb{S}^j)} w_{i,j}\Big]}{W_j}.$$

The proof uses Equation (11) and the fact that $Pr(RR_{i,j} \cap S_j \neq \emptyset) = AP(v_i, c_j|S_j)$ [29]. □

Next, we discuss the time and space complexity of GCW, assuming the same budget k for each player. A full round consists of $|C|$ best responses. Best response j must first construct the RR sets in order to generate a seed set of size k. There are efficient quasi-linear methods for that in terms of $|V|, |E|$; e.g., [37] provides a $O((l+k)(|V|+|E|)\log|V|/\epsilon^2)$ algorithm in expectation, where l, ϵ are related to the quality of the approximation. The next step is the computation of the awareness probability $AP(v_i, c_j|S_j)$ for all nodes $v_i \in V$, which is used by best response $j+1$ to obtain $\overline{AP}(v_i|\mathbb{S}^{j-1})$ via (10). This is achieved by $M = 1K$ MC simulations, each applying BFS, for a total cost of $O(M(|V|+|E|))$, which is dominated by the cost of the first step. Considering that $|E| \geq |V|$ and a fixed approximation quality as input, by summing up over the $|C|$ best responses, we obtain an asymptotic time complexity of $O\Big(|C|k|E|\log|V|\Big)$. For the space complexity, note that the expected total size of the RR sets constructed for each best response is $O(k|E|\log|V|)$; see, e.g., [37]. Since we do not need to store the RRs of previous competitors, the total space for RR sets is also $O(k|E|\log|V|)$. The MC simulations for a best response use $O(|V|)$ space, since we just need to know how many times a given node was influenced in the M simulations. This is dominated by the matrix of size $|V||C|$ that stores the awareness probabilities. To summarize, the expected total space complexity is $O(k|E|\log|V| + |V||C|)$.

Finally, we discuss some general aspects of our scheme. First, Theorem 3 is valid for any competitor ordering. We apply a random ordering because it converges fast to solutions of high quality. Alternative schemes such as selecting the player that increases the social welfare by the largest amount are quadratic in the number of competitors, as opposed to the linear complexity of random order. Second, the game is inherently sequential, since the best response of the current competitor depends on the completion of those of previous players. Nevertheless, there are parallelization opportunities within each best response. Both Monte Carlo simulations and the computation of RRs can benefit from parallel threads, each performing an independent simulation in the graph. Third, GCW assumes fixed budgets for each product. This is realistic, as it captures the situation that an advertiser has a limited budget distributed to various products according to their marketing importance. Moreover, it enables an efficient single-round method, where each competitor selects a seed set of predefined size. The case of a total budget that we must be split on-the-fly among the $|C|$ products is more challenging and expensive. Furthermore, a single budget may allocate very few or zero seeds to some products, resulting in unbalanced or unfair allocations. An interesting direction for future work is to explore efficient algorithms for the AtI model with fairness guarantees given a constraint on the total budget.

6 EXPERIMENTAL EVALUATION

Section 6.1 presents the experimental settings and the implementation details of the evaluated methods. Section 6.2 compares the effectiveness and efficiency of GCW against the baseline algorithm. Section 6.3 investigates the approximation bound and convergence of GCW.

6.1 Experimental Settings

To the best of our knowledge, there do not exist data sets that contain real influence probabilities, together with information about competing products and similarities of those products to users. To overcome this challenge, we simulate two scenarios over geosocial and social networks, by computing incomplete information in a realistic manner. As a result, we obtain six different settings, covering distinct applications, graph sizes and structures, and similarity measures. For the geosocial scenario, we use Gowalla[8] or Foursquare. Gowalla contains 12,748 users, connected through 96,836 edges, who checked-in at Austin and Dallas during a weekend in February 2009. Foursquare [28] contains 2,127,093 users over the world in September 2013. The number of edges is 17,280,702. Isolated nodes (without adjacent edges) were removed. Since the networks do not contain edge probabilities, in order to generate them, we first applied two clustering algorithms: DGCD [42] and k-means. For DGCD[9], we set the algorithm parameters to $\gamma = 300$, $\epsilon = 0.45$, and $\mu = 3$ for both data sets; the number of generated clusters is 701 in Gowalla, and 47903 in Foursquare. For k-means, where the number of clusters is an input parameter, we select a relatively small number of clusters, 32 in Gowalla and 128 in Foursquare, to diversify our settings. Then, we fix the probabilities of all edges between clusters to 0.001. For the intra-cluster edges we follow [10, 11, 29, 38], and set $p(v,u) = \frac{1}{in\text{-}degree(u)}$ [10]. For LT, the thresholds are randomly selected in the range [0,1]. This methodology avoids the issue of diminishing returns, i.e., when a small seed set suffices to reach most of the users on the graph. The competing products are events. Specifically, for Gowalla, we collected 128 social events from Eventbrite[11] at Austin and Dallas during the same period as the check-ins. Foursquare already contains 1,143,092 events. We set the similarity $w_{i,j}$ between each user v_i and event c_j as $w_{i,j} = 1 - \frac{d_{i,j}}{d_{max}}$, where $d_{i,j}$ is the Euclidean distance between v_i and c_j and d_{max} is the maximum Euclidean distance between all users and events. Consequently, a user is influenced by the event closest to his location, among those that he became aware of.

For the social scenario, we use two social networks: Flixster (flixster.com) and Last FM (lastfm.com). Flixster has 96,369 and Last FM 1,318 users, who rate movies or like songs, respectively. The number of edges is 377,868 in Flixster (10,852 in Last FM), and the

[8] https://snap.stanford.edu
[9] DGCD is a geosocial clustering algorithm that groups users based on their social connectivity and location.
[10] The average in-degree of Gowalla is 7.59 and that of Foursquare is 8.12.
[11] https://www.eventbrite.com

number of ratings is 8,196,078 (1,208,640 in Last FM). Since neither network contains edge probabilities, they were learned according to [18]. The competing products are the movies (Flixster) and songs (Last FM), where the similarity between each user and movie/song is the normalized rating. If there is no record for some (user,movie) or (user,song), we generate the rating using SVD, a collaborative filtering algorithm based on probabilistic matrix factorization [32].

Since there is no previous work on CWIM, we compare GCW against the Naïve baseline (NA), discussed in Section 3.3. NA is also based on Reverse Reachable sets, but each seed set is generated independently of other products, while GCW performs a single round of best response dynamics, where a seed set takes into account the previous ones. In order to create the RR sets in GCW and NA, we apply the D-SSA algorithm [34], with parameters $\epsilon = 0.01$ and $\delta = 0.01$, where ϵ refers to the error, and δ to the probabilistic guarantee. After computing a seed set, we estimate the social welfare (i.e., total similarity/influence $\sigma(\mathbb{S})$), by performing 5K Monte Carlo simulations. Each simulation propagates awareness according to the model used, ATIIC or ATILT. The total similarity is computed based on the largest weight between each user and the competitors that he becomes aware of. We report the average $\sigma(\mathbb{S})$ over all the simulations. This similarity evaluation is identical for both GCW and NA, and does not constitute part of their running time.

All algorithms are implemented in C++ and evaluated on two parameters: the number of competitors $|C|$ and the budget k (i.e., the seed set size) of each competitor. The value of $|C|$ ranges between 4 and 64, and that of k between 1 and 50. When we fix the number of competitors or the budget, we set $|C| = 16$ and $k = 10$, respectively. For the geosocial scenario, we select $|C|$ random events among those available for each data set. For the social scanario, the competitors are chosen randomly among the 128 movies or songs with the largest number of ratings. The experiments are conducted on a Linux Ubuntu 20.04.1 server with AMD Ryzen Threadripper 3960X 24-Core Processor @ 3.8GHz and 64 GB main memory.

6.2 Comparison with Baseline

Figures 2 - 7 show the gain of GCW over NA as a function of the number of competing products for ATIIC or ATILT in all data sets ($k = 10$). Specifically, the gain corresponds to the increase of the total similarity/influence $\sigma(\mathbb{S})$. The gain grows with $|C|$, and in some cases it exceeds 40% for $|C| = 64$. This happens because a large number of seed nodes leads to extensive overlap among the seed sets generated by NA. GCW avoids this problem because, when choosing a seed set S_j for c_j, it does not consider users who are likely to be aware of a more similar product $c_{j'}$ as these users would not be beneficial for c_j. With the exception of Flixster, the gains are similar for ATIIC and ATILT. Note that each diagram illustrates the mean and standard deviation (vertical lines) of 100 experiments in Gowalla, Flixster, Last FM and 10 in Foursquare (due to its larger size). The gain exhibits rather small standard deviation, despite the fact that the set of competitors is different for each of the averaged experiments.

Figures 8 and 9 illustrate the gain versus the seed set size for Gowalla (k-means) and Last-FM for $|C| = 16$. Similar to the case of $|C|$, the gain, in general, increases with k because large seed sets exhibit more overlap for NA, while they present better optimization

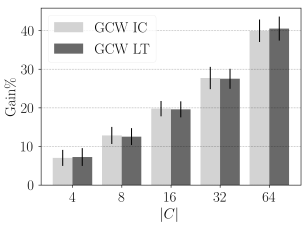


Figure 2: Percentage gain of GCW over NA as a function of $|C|$ (Gowalla, k-means)

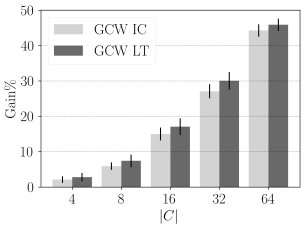


Figure 3: Percentage gain of GCW over NA as a function of $|C|$ (Gowalla, DGCD)

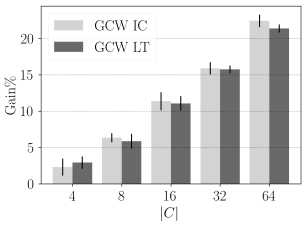


Figure 4: Percentage gain of GCW over NA as a function of $|C|$ (Foursquare, k-means)

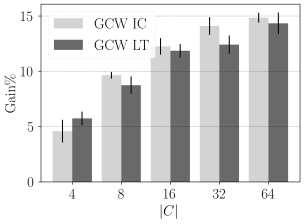


Figure 5: Percentage gain of GCW over NA as a function of $|C|$ (Foursquare, DGCD)

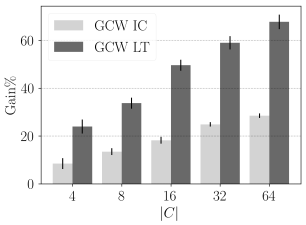


Figure 6: Percentage gain of GCW over NA as a function of $|C|$ (Flixster)

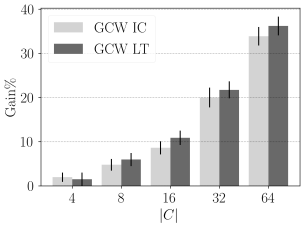


Figure 7: Percentage gain of GCW over NA as a function of $|C|$ (Last FM)

opportunities for GCW. For instance, in the geosocial scenario, two events that are close (in terms of Euclidean distance) compete for users in their vicinity. The number of users who become aware of the events grows with the seed set size, increasing the overlap and the competition between the events. The diagrams for the other data sets are similar and omitted due to limited space.

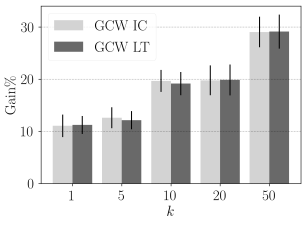


Figure 8: Percentage gain of GCW over NA as a function of k (Gowalla, k-means)

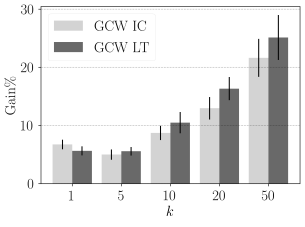


Figure 9: Percentage gain of GCW over NA as a function of k (Last FM)

The bars in Figures 10 and 11 correspond to the normalized running time of GCW (i.e., time of GCW divided by time of NA) versus the number of competing products and the budget, respectively, in Gowalla. For large values of $|C|$ and k, GCW is slower than NA due to the Monte Carlo simulations that each competitor performs at the end of its best response in order to estimate the awareness probabilities. However, for small values, GCW outperforms NA because the most costly part is the computation of RR sets. GCW guides the node selection for RR set generation, by excluding users likely to be influenced by other competitors, while NA picks random nodes. Consequently, GCW requires a smaller number of RR sets to meet the quality guarantees of the underlying D-SSA, employed by both NA and GCW. The diagrams also include the absolute values of the running time in seconds for GCW (as numbers above the bars). Interestingly, the absolute time drops as k increases. This is due to the D-SSA algorithm [34] used by both GCW and NA to generate the RR sets. As shown in [34], and more clearly in a revised version [24], D-SSA needs to form more RR sets for small values of k.

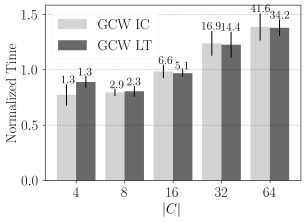


Figure 10: Running time of GCW over NA vs. $|C|$ (Gowalla, k-means)

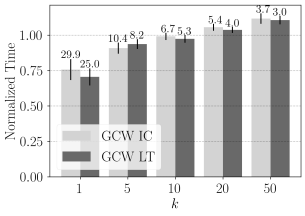


Figure 11: Running time of GCW over NA vs. k (Gowalla, k-means)

6.3 Lower Bound and Convergence

The next set of experiments examines the quality of the approximation lower bound b, from Theorem 3, after a single round, when competitors start with empty strategies and select seed sets in a random order. Figures 12 - 13 show the average value of b, versus the number of competitors $|C|$, for Gowalla and Last FM, respectively ($k = 10$). In all cases b is at least 0.2, and sometimes its value approaches the optimal $b^* \approx 0.387$ for basic games. Accordingly, even though GCW is not a basic game, in practice its lower bound is not far of that for basic games.

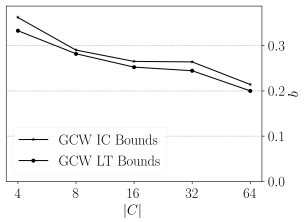


Figure 12: Lower bound b vs. $|C|$ (Gowalla, k-means)

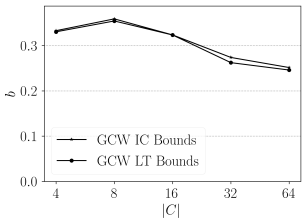


Figure 13: Lower bound b vs. $|C|$ (Last FM)

The observed values of b indicate that GCW can reach a solution with good quality guarantees within a single round. The last set of experiments investigates the behavior of best response dynamics with multiple rounds. Figures 14 and 15 show the total gain over NA versus the number of rounds in Gowalla and Last FM, respectively ($|C| = 16$ and $k = 10$). There is some small improvement at round 2, and to a lesser degree at round 3, after which the gain remains stable. As expected, very few rounds of best responses suffice for convergence. Since each round performs roughly the same amount of computations, the running time is proportional to the number of rounds.

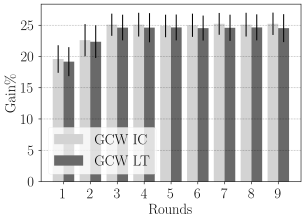


Figure 14: Gain over NA vs. rounds (Gowalla, k-means)

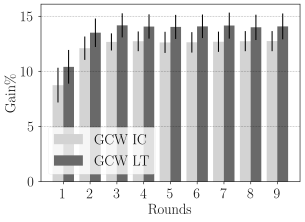


Figure 15: Gain over NA vs. rounds (Last FM)

Summarizing the evaluation, GCW significantly extends the total influence compared to the baseline algorithm, especially in the presence of numerous competing products with big budgets. Moreover, a single round of best responses suffices to reach high quality solutions, enabling the application of GCW to large graphs.

7 CONCLUSION

In collective influence maximization, the owner of multiple competing products wishes to select a seed set for each product, so that the total influence is maximized. We propose an Awareness-to-Influence (AtI) diffusion model that separates the awareness and influence processes. To compute the seed sets, we introduce GCW, a monotone utility game with bounded price of anarchy. Based on that, we show that a single round of best-response dynamics can produce solutions with quality guarantees. An extensive experimental evaluation demonstrates the effectiveness and efficiency of the proposed methods. Our model assumes independent awareness phases for the various products. Product interactions occur only during the influence phase, when users adopt the product of highest similarity among those that they became aware of. This assumption enables monotonicity, sumodularity and the monotone utility game property. Future work can investigate models with dependencies in the awareness phase. Alternative models may also involve different types of influence. For instance, we may allow users to be influenced simultaneously by multiple products to capture applications with non-exclusive influence requirements; e.g., when a user can buy several advertised products. Another direction is to replace the fixed budgets per product with a total budget that must be split fairly among all products. Finally, it would be interesting to compare GCW to other paradigms (e.g., local search, dynamic programming) in terms of quality or efficiency.

ACKNOWLEDGMENTS

This work was supported by GRF grants 16231216 and 16205117 from Hong Kong RGC.

REFERENCES

[1] N. Alon, M. Feldman, A. D. Procaccia, and M. Tennenholtz. 2010. A Note on Competitive Diffusion Through Social Networks. In *Inf. Process. Lett. (vol. 110, no. 6)*. 221–225.

[2] A. Arora, S. Galhotra, and S. Ranu. 2017. Debunking the Myths of Influence Maximization: An In-Depth Benchmarking Study. In *Proceedings of SIGMOD international conference on Management of data (ACM)*.

[3] Eli Berger. 2001. Dynamic Monopolies of Constant Size. *J. Comb. Theory Ser. B* 83, 2 (Nov. 2001), 191–200. https://doi.org/10.1006/jctb.2001.2045

[4] S. Bharathi, D. Kempe, and M. Salek. 2007. Competitive influence maximization in social networks. In *WINE*. 306–311.

[5] C. Borgs, M. Brautbar, J. Chayes, and B. Lucier. 2014. Maximizing social influence in nearly optimal time. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 946–957.

[6] A. Borodin, Y. Filmus, and J. Oren. 2010. Threshold models for competitive influence in social networks. In *WINE*. 539–550.

[7] Ceren Budak, Divyakant Agrawal, and Amr El Abbadi. 2011. Limiting the Spread of Misinformation in Social Networks. In *Proceedings of the 20th International Conference on World Wide Web* (Hyderabad, India) *(WWW '11)*. 665–674.

[8] Tim Carnes, Chandrashekhar Nagarajan, Stefan M. Wild, and Anke van Zuylen. 2007. Maximizing Influence in a Competitive Social Network: A Follower's Perspective. In *Proceedings of the Ninth International Conference on Electronic Commerce (ICEC '07)*. 351–360.

[9] Wei Chen, Alex Collins, Rachel Cummings, Te Ke, Zhenming Liu, David Rincón, Xiaorui Sun, Yajun Wang, Wei Wei, and Yifei Yuan. 2011. Influence Maximization in Social Networks When Negative Opinions May Emerge and Propagate. In *Proceedings of the Eleventh SIAM International Conference on Data Mining, SDM 2011, April 28-30, 2011, Mesa, Arizona, USA*. 379–390.

[10] W. Chen, C. Wang, and Y. Wang. 2010. Scalable influence maximization for prevalent viral marketing in large-scale social networks. In *KDD*. 1029–1038.

[11] W. Chen, Y. Wang, and S. Yang. 2009. Efficient influence maximization in social networks. In *KDD*. 199–208.

[12] W. Chen, Y. Yuan, and L. Zhang. 2010. Scalable influence maximization in social networks under the linear threshold model. In *ICDM*. 88–97.

[13] P. Dagum, R. Karp, M. Luby, and S. Ross. 2000. An optimal algorithm for monte carlo estimation. In *SIAM J. Comput (vol. 29)*. 1484–1496.

[14] S. R. Etesami and T. Başar. 2014. Complexity of equilibrium in diffusion games on social networks. In *2014 American Control Conference*. 2065–2070.

[15] Naoka Fukuzono, Tesshu Hanaka, Hironori Kiya, Hirotaka Ono, and Ryogo Yamaguchi. 2020. Two-Player Competitive Diffusion Game: Graph Classes and the Existence of a Nash Equilibrium. In *SOFSEM 2020: Theory and Practice of Computer Science*. Springer International Publishing, 627–635.

[16] Sainyam Galhotra, Akhil Arora, and Shourya Roy. 2016. Holistic Influence Maximization: Combining Scalability and Efficiency with Opinion-Aware Models. In *Proceedings of the 2016 International Conference on Management of Data (SIGMOD '16)*. 743–758.

[17] M. Goemans, V. Mirrokni, and A. Vetta. 2005. Sink Equilibria and Convergence. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. 142–154.

[18] Amit Goyal, Francesco Bonchi, and Laks V.S. Lakshmanan. 2010. Learning Influence Probabilities in Social Networks. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining* (New York, New York, USA) *(WSDM '10)*. Association for Computing Machinery, New York, NY, USA, 241–250. https://doi.org/10.1145/1718487.1718518

[19] A. Goyal, W. Lu, and L. Lakshmanan. 2011. Optimizing the greedy algorithm for influence maximization in social networks. In *20th International World Wide Web Conference (WWW)*. 47–48.

[20] A. Goyal, W. Lu, and L. Lakshmanan. 2011. SIMPATH: An efficient algorithm for influence maximization under the linear threshold model. In *ICDM*. 211–220.

[21] S. Goyal and M. Kearns. 2012. Competitive contagion in networks. In *STOC*. 759–774.

[22] X. He and D. Kempe. 2013. Price of anarchy for the n-player competitive cascade game with submodular activation functions. In *WINE*. 232–248.

[23] Xinran He, Guojie Song, Wei Chen, and Qingye Jiang. 2012. Influence Blocking Maximization in Social Networks under the Competitive Linear Threshold Model.. In *SDM*. SIAM / Omnipress, 463–474.

[24] Keke Huang, Sibo Wang, Glenn Bevilacqua, Xiaokui Xiao, and Laks V. S. Lakshmanan. 2017. Revisiting the Stop-and-stare Algorithms for Influence Maximization. *Proc. VLDB Endow.* 10, 9 (May 2017), 913–924. https://doi.org/10.14778/3099622.3099623

[25] Satoru Iwata, Shin-ichi Tanigawa, and Yuichi Yoshida. 2016. Improved Approximation Algorithms for K-Submodular Function Maximization. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '16)*. 404–413.

[26] D. Kempe, J. M Kleinberg, and É. Tardos. 2003. Maximizing the spread of influence through a social network. In *KDD*.

[27] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance. 2007. Cost-effective outbreak detection in networks. In *KDD*. 420–429.

[28] J. J. Levandoski, M. Sarwat, A. Eldawy, and M. F. Mokbel. 2012. Lars: A location-aware recommender system. In *ICDE*.

[29] Y. Li, D. Zhang, and K. L Tan. 2015. Real-time targeted influence maximization for online advertisements. In *PVLDB (vol. 8, no. 10)*.

[30] Wei Lu, Francesco Bonchi, Amit Goyal, and Laks V.S. Lakshmanan. 2013. The Bang for the Buck: Fair Competitive Viral Marketing from the Host Perspective. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '13)*. 928–936.

[31] V. Mirrokni and A. Vetta. 2004. Convergence issues in competitive games. In *In APPROX-RANDOM*. 183–194.

[32] Andriy Mnih and Russ R Salakhutdinov. 2008. Probabilistic matrix factorization. In *Advances in neural information processing systems*. 1257–1264.

[33] Stephen Morris. 2000. Contagion. *The Review of Economic Studies* 67, 1 (2000), 57–78.

[34] Hung T. Nguyen, My T. Thai, and Thang N. Dinh. 2016. Stop-and-Stare: Optimal Sampling Algorithms for Viral Marketing in Billion-scale Networks. In *Proceedings of the 2016 International Conference on Management of Data* (San Francisco, California, USA) *(SIGMOD '16)*. ACM, New York, NY, USA, 695–710. https://doi.org/10.1145/2882903.2915207

[35] Naoto Ohsaka and Yuichi Yoshida. 2015. Monotone K-Submodular Function Maximization with Size Constraints. In *Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS '15)*. 69–702.

[36] Nishith Pathak, Arindam Banerjee, and Jaideep Srivastava. 2010. A Generalized Linear Threshold Model for Multiple Cascades. In *Proceedings of the 2010 IEEE International Conference on Data Mining (ICDM '10)*. 965–970.

[37] Y. Tang, Y. Shi, and X. Xiao. 2015. Influence maximization in near-linear time: A martingale approach. In *Proceedings of SIGMOD international conference on Management of data (ACM)*. 1539–1554.

[38] Y. Tang, X. Xiao, and Y. Shi. 2014. Influence maximization: Near-optimal time complexity meets practical efficiency. In *Proceedings of SIGMOD international conference on Management of data (ACM)*. 75–86.

[39] V. Tzoumas, C. Amanatidis, and E. Markakis. 2012. A game-theoretic analysis of a competitive diffusion process over social networks. In *WINE*. 1–14.

[40] Vijay Vazirani. 2003. *Approximation Algorithms*. Springer-Verlag Berlin Heidelberg.

[41] Adrian Vetta. 2002. Nash Equilibria in Competitive Societies, with Applications to Facility Location, Traffic Routing and Auctions. In *Proceedings of the 43rd Symposium on Foundations of Computer Science (FOCS)*.

[42] Kai Yao, Dimitris Papadias, and Spiridon Bakiras. 2019. Density-based community detection in geo-social networks. In *Proceedings of the 16th International Symposium on Spatial and Temporal Databases*. 110–119.