Knowledge-Based Systems 62 (2014) 57–68

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

CT-IC: Continuously activated and Time-restricted Independent Cascade model for viral marketing

Jinha Kim, Wonyeol Lee, Hwanjo Yu *

Pohang University of Science and Technology (POSTECH), Pohang, South Korea

ARTICLE INFO

Article history:
Received 27 January 2013
Received in revised form 18 February 2014
Accepted 25 February 2014
Available online 5 March 2014

Keywords:
Influence maximization
Viral marketing
Social networks
Influence diffusion model
Graph mining

ABSTRACT

Influence maximization problem has gained much attention, which is to find the most influential people. Efficient algorithms have been proposed to solve influence maximization problem according to the proposed diffusion models. Existing diffusion models assume that a node influences its neighbors *once*, and there is *no time constraint* in activation process. However, in real-world marketing situations, people influence his/her acquaintances *repeatedly*, and there are often *time restrictions* for a marketing. This paper proposes a new realistic influence diffusion model *Continuously activated and Time-restricted IC (CT-IC) model* which generalizes the IC model. In CT-IC model, every active node activate its neighbors *repeatedly*, and activation continues until a given time. We first prove CT-IC model satisfies *monotonicity* and *submodularity* for influence spread. We then provide an efficient method for calculating *exact* influence spread for a directed tree. Finally, we propose a scalable influence evaluation algorithm under CT-IC model CT-IPA. Our experiments show CT-IC model finds seeds of higher influence spread than IC model, and CT-IPA is four orders of magnitude faster than the greedy algorithm while providing similar influence spread.

1. Introduction

Due to the rapid growth of online social network sites such as Facebook or Twitter, we now experience that individuals' information and ideas are spread to others extremely fast via online social networks. It enables us to use online social networks as a stage of viral marketing which exploits word-of-mouth effect. However, when applying viral marketing, we face several important difficulties including *influence maximization problem* which aims to find the most influential people. Let us look at the classic example of influence maximization problem. Suppose that a company develops a new product and wants to sell it to the general as many as possible. In the viral marketing, the company gives the new product to the "initial" people for free, and expects them to use it as well as to persuade their friends to use it together. Moreover, there is a chance that their friends may also recommend their friends' friends to use it and so on. In this situation, the company could think of the following question: "who should be the initial people to make the largest profit?". This question is so called influence maximization problem. Given a graph representing a social network, a parameter k denoting company's budget, and a stochastic process model of how influence is spread through people, the influence maximization problem aims at finding k *seeds* (initial nodes) which maximizes *influence spread* (the number of people who use the new product at the final state).

Kempe et al. [1] first proposed the influence maximization problem and suggested two basic *influence diffusion models* – *Independent Cascade (IC) model* and *Linear Threshold (LT) model*. In IC model, an *active* node tries to activate its neighbors with a given probability and, in LT model, a node is activated only if some portion of its neighbors are already active. Along with IC and LT diffusion models, novel influence diffusion models which reflect different aspects of influence diffusion. Chen et al. [2] proposed IC model with negative opinions (IC-N) which extends IC model by considering the propagation of both negative and positive opinions. He et al. [3] and Borodin et al. [4] proposed competitive LT (CLT) model in which two competing opinions are spread in a LT model manner. Li et al. [5] proposed voter model in a signed network.

Although several novel diffusion models have been suggested, they miss two important aspects of influence diffusion in the real-world viral marketing applications. First, in IC and IC-N models, when a node becomes active, it can activate its neighbors *only once*. However, in real-world marketing situations, people

* Corresponding author. Tel.: +82 542792388.
E-mail addresses: goldbar@postech.ac.kr (J. Kim), wylee@postech.ac.kr (W. Lee), hwanjoyu@postech.ac.kr (H. Yu).

http://dx.doi.org/10.1016/j.knosys.2014.02.013
0950-7051/

influence his or her acquaintances *repeatedly*. For example, when you write a post about a new product on Facebook wall, your friends will see it not only right after you write but also few days later. Secondly, in IC, IC-N and LT models, activation process is continued until *no more* activation happens at all. However, in the real world, we often have *time restriction* and thus cannot wait until the influence is spread "completely". For example, a cellphone company, which releases a new product every only six months, does not expect much profit from the existing product after six months later because the company will move the focus of its marketing on the new product.

This paper proposes a more down-to-earth influence diffusion model for viral marketing applications called *Continuously activated and Time-restricted IC (CT-IC) model*. CT-IC model is a generalization of IC model, and it differs in two aspects: (a) every active node can activate its neighbors *repeatedly* and (b) activations are processed until *a given time T*. Thus, CT-IC model provides two controllable parameters for the repeatable activation and the time constraint, and IC model becomes a special case of CT-IC model with a single activation and infinite time constraint.

After defining CT-IC model, we prove CT-IC model satisfies two crucial properties – monotonicity and submodularity – for influence spread, which leads to guaranteeing $(1-1/e)$-approximation solution of the influence maximization problem under CT-IC model when a simple greedy algorithm is applied. Our proof exploits an alternative activation process which is equivalent to activation process of CT-IC model. In CT-IC model, we *flip a coin* to decide the success of an activation trial whenever decision is required. However, in the alternative model, we decide the number of activation trials by flipping all coins *before* influence propagation process starts. When flipping coins, we replace each edge's weight of propagation probability with a natural number which represents how many trials are required for a node to activate its neighbors. In this modified graph, a node is activated if and only if the distance between seed nodes and non-seed nodes is no more than *T*. By using the alternative model, we can easily prove the two important properties.

We then provide an efficient method for calculating *exact* influence spread when a graph is restricted to a directed tree. Because CT-IC model is a generalization of IC model, the equations computing the exact influence spread are more involved than those in IC model. We apply these equations to a special case of a directed tree, a simple path, to get a useful way to compute one node's influence on another node only through a path. Influence spread of a path is calculated as follows. A *matrix* weight which is related to propagation probability is assigned to each edge. Then, the sum of the first row of the matrix, which is obtained by multiplying matrix weights along the path, is the influence spread of the path. Using this result, we also show that it is hard to define a local tree structure, such as MIA and MIA-N (for IC and IC-N models) [2,6].

By using influence spread evaluation of a simple path, we propose an influence evaluation algorithm CT-IPA for CT-IC model which extends a scalable algorithm, *independent path algorithm* (IPA), for IC model [7]. IPA is based on two simple assumptions. Influence is propagated only through *critical paths*, and activation process through each critical path is independent of each other. More precisely, critical paths are defined by the simple paths whose influence spread is no less than a threshold θ. Since influence spread of a critical path is computed by multiplying matrix weights of its edges under CT-IC model, CT-IPA seamlessly extends IPA with additional treatments for merging multiple edges.

Extensive experiments are conducted on four real networks to find characteristic of CT-IC model and to compare CT-IPA with other algorithms. For the same dataset, CT-IC model and IC model produce seed sets of quite different nodes, and the nodes shared by two models have different ranks. Also, when seed sets produced by the two models are applied to CT-IC model, CT-IC seed set always shows higher influence spread than IC seed set. This result supports that CT-IC model always produces better results than IC model in more realistic viral marketing situations which allows continuous activation and time constraint. In addition, CT-IPA shows over four orders of magnitude faster than greedy algorithm without sacrificing influence spread.

This paper is organized as follows. After describing related work in Section 2, we propose CT-IC model and show its properties in Section 3. Section 4 presents efficient methods to compute exact influence spread. Section 5 proposes a scalable algorithm for influence maximization problem under CT-IC model. Section 6 illustrates the experiment results, and Section 7 concludes this paper.

2. Related work

Various influence diffusion models. Three representative influence diffusion models are studied in the early study of the influence diffusion model. Kempe et al. [1] suggested General Cascade (GC) model and General Threshold (GT) model which are generalized version of IC and LT models, and show that two models are equivalent. In GC model, the propagation probability of a node depends on the history of activation trials while in IC model it is constant. In GT model, threshold function, which determines whether each node becomes active or not, is a general function of active neighbors' weights while in LT model it is a summation of active neighbors' weights. Different from IC and LT model in which active nodes try to influence inactive nodes, voter model [8] deals with the situation that every node has one of two different opinions and two opinions compete for occupying more nodes.

Along with the traditional influence diffusion models, various extensions of those models were proposed recently. IC-N model [2] considers the propagation of negative opinion. In IC-N model, a successful activation trial of an positively active node to its inactive neighbor results in either positive activation or negative activation. On the contrary, a successful activation trial of an negatively active node result in only negative activation. CLT model [3] extends LT model by considering two competing opinions in networks. In CLT models, seed nodes are activated and have one of two competing opinions. An active nodes tries to persuade inactive neighbors to have its supporting opinion. Signed voter model [5] extends voter model by allowing negative influence of a node. In signed voter model, when two nodes of an edge have friend relationship, one node's successful trial to influence the other node results in having the same opinion of the other node. Otherwise, the other node has the opposite opinion. The characteristic embedded in IC-N and CLT model is similar in that the successful influence trial results in the negative influence – having the opposite opinion.

Although all the above influence diffusion models reflect various aspects of influence propagation in real world, none of them consider the crucial characteristics in real influence propagation – repeated activation trials and time restriction. Our proposing CT-IC model embraces these two essential characteristics.

The relationship between the existing influence diffusion models and CT-IC model is shown in Fig. 1. The models located in upper rows are basic model and their extensions are located in lower rows and are connected by directed edges. Each edge label indicates the characteristics additionally embedded in the extended model.

Learning parameters of influence diffusion models. Along with designing diffusion models described above, learning the propagation probability is also important. Goyal et al. [9] and Saito et al. [10] study how to learn such probability from the past action

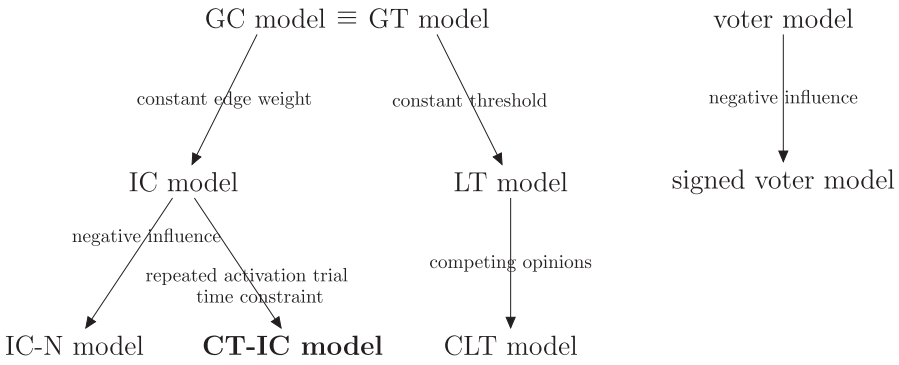

Fig. 1. Relationship between influence diffusion models.

logs. However, they stick to an instance of GC model which has different behavior from CT-IC model.

For learning parameters of CT-IC model including the propagation probability, Dynamic Bayesian Networks (DBNs) and its inference techniques are useful. DBNs [11] are widely used in learning generative models of sequence data such as bio-sequence [12], voice [13], office activity [14]. CT-IC model is an instance of DBNs in that the state of a node in the current time step depends on the state of its adjacent nodes in the previous time step. Accordingly, various inference techniques [11–15] can learn parameters of CT-IC model from observed data. However, since our goal is proposing a novel influence diffusion model and its efficient influence maximization processing, learning parameters is out of scope in this paper.

Influence maximization problem and its efficient processing methods. When an influence diffusion model is given, the influence maximization problem aims to find the most k influential nodes in a directed graph. Let $G(V,E)$ a directed graph and $\sigma(S)$ the quantified influence of a node set $S \subseteq V$. The influence maximization problem is formalized as follows.

$$\arg \max_{S \subseteq V, |S|=k} \sigma(S) \tag{1}$$

The influence maximization processing confronts two major challenges. The first one is the combinatorial optimization of Eq. (1) is NP-Hard. To detour the NP-Hardness of the influence maximization problem, Kempe et al. [1] show that the greedy algorithm guarantees $(1-1/e)$ approximation ratio. To apply the greedy algorithm, it is required to prove that the underlying influence diffusion model satisfies three properties – *non-negativity*, *monotonicity* and *submodularity*. Various influence diffusion models hold these three properties and are applicable to the greedy algorithm – IC and LT model in [1], GC and CT models in Mossel and Roch [16], IC-N model in Chen et al. [2], CLT model in [3], and the signed voter model in [5]. As a further optimization for the greedy algorithm, CELF-greedy [17], CELF++ [18], NewGreedy [19], and community-based greedy algorithm are suggested.

The second challenge of the influence maximization processing is that the exact influence evaluation cannot be achieved in a polynomial time. Because most of the influence diffusion models does not have closed form of $\sigma(S)$, the influence evaluation exploits time-consuming Monte Carlo simulation, which repeats the actual influence diffusion simulation until a stable influence is acquired. Thus, there have been many studies to reduce the running time of the original greedy algorithm. Several efficient algorithms are proposed based on approximating diffusion models. For IC model, Shortest Path Model [20], PMIA [6] and IPA [7] are proposed. For LT model, LDAG [6] is proposed. For IC-N model, MIA-N [2] is proposed. For CLT model, CLDAG [3] is proposed. For signed voter model, SVIM [5] is proposed.

3. CT-IC model

In this section, we describe the motivation of CT-IC model with several examples (Section 3.1). Then, we formally define CT-IC model (Section 3.2), and prove its important properties (Section 3.3).

3.1. Motivation

Although various existing models are proposed to reflect the real influence diffusion dynamics, they omits two major aspects. Specifically, in IC model which is the most widely used influence diffusion model [1,6,17,19,21], time limitation of marketing is ignored and every node has only single chance to activate it out-neighbors. From now on, several examples are provided to illustrate the importance of these two aspects in the real world which are not considered in IC model and other existing models.

First, every viral marketing campaign has time limit or constraint. Let us take an example of Apple's iPhone marketing. After iPhone 4 was release, Apple's marketing focus was promoting the sale of iPhone 4. With the active marketing support, most people were interested in iPhone 4 and started to purchase it. Consequently, a number of iPhone 4 were sold out for a while.[1] However, Apple does not expect that iPhone 4 lead the cellular phone market forever. Apple definitely developed another cutting edge phone, iPhone 4S. When Apple launched iPhone 4S, Apple obviously moved its advertising focus to iPhone 4S and the public also moves their interest to the new product. Apple has never advertised their old products after the new product's release. Even though Apple sold iPhone 4 after iPhone 4S release, the position of iPhone 4 was just for emerging markets.

From this particular example, it is important to get maximum profit *within a time limit*. In other words, when planning marketing, we have to set a time limit by considering the lifetime of the product as a market competitor. Since the above situation is applied to most marketing situations, time restriction should be considered in a realistic influence diffusion model.

Second, there exists repeated chances to influence friends or acquaintances in the real-world situations. Let us consider another example. Suppose you buy a new product and write a positive post about it in your Facebook wall. Then, the post appears to your friends and persuades them to have a positive opinion about the new product, which may lead them to buy it. The important thing here is that when revisiting your wall later, your friends may be persuaded to buy the product although they are not persuaded at the posting moment. In other words, your positive post will have

[1] Actually, iPhone4 was sold out over 2 weeks after its release. http://gizmodo.com/5564420/att-iphone-4-pre+orders-sold-out.

continuous influence on your friends. From this example, we observe that people typically have multiple chances to affect others on the same item. This observation is supported by the group joining behavior of Flickr network [9]. Hence, we should take the possibility of continuous activation chances into a new influence diffusion model.

Time constraint and continuous activation chances in influence diffusion process have not been contained in the existing models and our proposing CT-IC model's main contribution is to embrace these two crucial aspects in the influence diffusion model.

3.2. Model definition

CT-IC model is modeled on an abstracted directed graph. Let $G(V,E)$ of its vertex set V and its edge set E with a propagation probability $pp_0 : E \to [0,1]$ be a directed graph representing a social network. $pp_0(u,v)$ denotes the probability that a node u activates a node v one time step after u is activated. Given a seed set $S \subseteq V$ and time restriction T, *Continuously activated and Time-restricted IC (CT-IC) model* works as follows.

Every seed node $s \in S$ is activated at time step $t = 0$ and the activation is propagated through its neighbors at time $t = 1, 2, 3, \ldots$. Let A_t be the set of active nodes at time t with $A_0 = S$. At time t, every active node $u \in A_t$ tries to activate its inactive out-neighbors $v \in \{w \in N_{out}(u) \text{ and } w \notin A_t\}$ with probability $pp_{t-t_u}(u,v)$, where t_u is the activation time of u and $pp_t(u,v)$ is defined as

$$pp_t(u,v) = pp_0(u,v) \cdot f_{uv}(t). \tag{2}$$

Here, $f_{uv} : \mathbb{N}_0 \to \mathbb{R}_0^+$ is monotonically decreasing function and $f_{uv}(0) = 1$.

The monotonically decreasing property of f_{uv} is based on the observation that persuading friends is getting harder after each trial to persuade them. Suppose that your buy a new iPad and you friends do not have it yet. When you first show it to your friends, some of them probably have a strong impression on it and decide to buy it. After some days, when you show your friends it again, some of them who does not buy it probably purchase it due to the multiple exposure to it. but the number of influenced friends are not as many as compared to the first time. In sum, as time goes by the number of persuaded people decreases. This phenomena is supported by the group joining behavior of Flickr network and the shape of this decrease follows the exponential function [9]. Accordingly, in this paper, we use $f_{uv}(x) = \exp(-\alpha_u x)$ with a non-negative constant $\alpha_u > 0$ which represents how fast u's influence on its neighbors decreases.

After all activation trials are finished at time t, newly activated nodes S_t are included in the activated node set, so we have $A_{t+1} = A_t \cup S_t$ and a time step $t+1$ starts. This activation process is repeated until we arrive at time step T.

The big difference between CT-IC and IC model is that (a) *all* activation processes *stop* at global time limit T, not at time ∞ and (b) every active node has *multiple* chances to activate its neighbors until its neighbor becomes active or T is reached.

CT-IC model is a generalized version of IC model. This is because IC model is obtained by taking $\alpha_v \to \infty$ for all $v \in V$ (or $f_{uv}(x) = \delta(x)$ for all $(u,v) \in E$, where δ denotes Kronecker delta function) and $T = |V|$.

One might guess that CT-IC model can be reduced to the modified IC model by setting (u,v)'s propagation probability to $\sum_{t=0}^{T-1} pp_t(u,v)$ and giving time restriction. However, the modified IC model is *not* the same as CT-IC model. The reason is that this modified IC model ignores how long it takes for each node to activate others, which is an important factor in the reality. For example, suppose $t_u = 0$, u takes 3 time steps to activate v, and v takes 5 time steps to activate w (i.e. $t_v = 3$, $t_w = 8$). This event is converted into the event in the modified IC model that u, v, w are activated at time 0, 1, 2, respectively. Thus, when $T = 5$, w is not activated in CT-IC model while w is activated in the modified IC model, and such difference results in completely different consequence because each active node could produce large cascading effect. Hence, IC model cannot simulate CT-IC model without loss of CT-IC model's key features.

3.3. Properties of CT-IC model

To apply CT-IC model to the real viral marketing, the greedy algorithm should be applicable to the influence maximization problem under CT-IC model. The satisfactory conditions for the greedy algorithm are non-negativity, monotonicity, and submodularity of the influence spread under CT-IC model. The influence spread of a given seed set S at time t, $\sigma(S,t)$, is the expected number of active nodes when time step t starts. Then, given the number of seed nodes k and time constraint T, the influence maximization problem under CT-IC model is to find a set $S^* \in \arg\max_{S \subseteq V, |S| = k} \sigma(S,T)$. In the following, monotonic and submodular properties of CT-IC model are proven. The non-negativity property holds trivially by the definition of influence spread under CT-IC model.

Monotonicity and submodularity. In order to ensure that greedy algorithm produces $(1 - 1/e)$-approximation solution for influence maximization problem under CT-IC model, monotonicity and submodularity of CT-IC model should be proven. Here, for a given function $f : 2^V \to \mathbb{R}$, f is called *monotone* if $f(S) \leqslant f(S')$, $\forall S \subseteq S'$, and *submodular* if $f(S \cup \{v\}) - f(S) \geqslant f(S' \cup \{v\}) - f(S')$, $\forall S \subseteq S'$, $v \in V$.

To prove monotonicity and submodularity, we conceive an easy-to-analyze process which is equivalent to CT-IC model. Consider a specific edge $(u,v) \in E$. After u is newly activated at t_u, u tries to activate its inactive out-neighbors $v \notin A_t$ repeatedly until v becomes active. For easy demonstration, assume that u is the only in-neighbor node of v. Then, the probability that v is activated exactly at $t_u + t$ by u is equal to $pp_{t-1}(u,v)\prod_{i=0}^{t-2}(1 - pp_i(u,v))$. In order to decide when v becomes active, we only need to determine t for the above probability expression. Since probability function of t for each $(u,v) \in E$ is given as above in advance, we can decide t before activation process starts, and we have an equivalent activation process to CT-IC model.

Suppose that we decide t for each $(u,v) \in E$ before activation process starts, have a function $h : E \to \mathbb{N}$ that decides t for each edge. Let $G' = (V,E,h)$ be a graph with weight $h(u,v)$ for $(u,v) \in E$. Then, $v \in V$ is active at time t if and only if there exists $u \in S$ and a path from u to v in G' whose length is equal to or less than t, where S is a seed set. This fact also holds when $|N_{in}(v)| > 1$ because every activation trial is independent of each other. Based on this observation, after choosing h, we can compute influence spread deterministically. Theorem 1 proves monotonic and submodular properties of influence spread under CT-IC model based on the above observation.

Theorem 1. *The influence spread function $\sigma(\cdot,t)$ under CT-IC model is monotone and submodular for all $t \geqslant 0$.*

Proof. Let $h : E \to \mathbb{N}$ be a function of $(u,v) \in E$ which returns the number of time steps (influence trials) taken by u to activate v. We choose a specific h from $\mathcal{H} = \{h \mid h : E \to \mathbb{N}\}$ which follows $\Pr[h(u,v) = t] = pp_{t-1}(u,v)\prod_{i=0}^{t-2}(1 - pp_i(u,v))$. For any $S \subseteq V$, let $R_h(S,t)$ be the set of active nodes at time t when seed nodes are S and the successful influence trials follow h. Then, $R_h(S,t)$ is computed as

$R_h(S,t) = \{v \in V |\ \exists u \in S$ such that $d(u,v) \leqslant t\}$,

where $d(u,v)$ is the length of the shortest path from u to v under given h. Then, $\sigma_h(S,t)$, which is the influence spread at time t with seed set S under h, becomes $\sigma_h(S,t) = |R_h(S,t)|$. In this context, influence spread on G under CT-IC model can be computed as

$$\sigma(S,t) = \sum_{h \in \mathcal{H}} \Pr[h] \cdot \sigma_h(S,t).$$

$R_h(S,t)$ is monotone because adding a new node to S always results in more reachable nodes and $\sigma_h(S,t)$ is monotone because $R_h(S,t)$ is monotone. $\sigma(S,t)$ is also monotone because linear combination of monotone function is also monotone.

Since $\sigma_h(S\cup\{v\},t) - \sigma_h(S,t) = |R_h(S\cup\{v\},t)\setminus R_h(S,t)| = |R_h(\{v\},t)\setminus R_h(S,t)|$ holds and $R_h(\cdot,t)$ is monotone, we have $R_h(\{v\},t)\setminus R_h(S,t) \supseteq R_h(\{v\},t)\setminus R_h(S',t)$ for any $S' \supseteq S$. So, $\sigma_h(\cdot,t)$ is submodular for all $t \geqslant 0$ and thus $\sigma(\cdot,t)$ is also submodular for all $t \geqslant 0$ since $\sigma(S,t)$ is the linear combination of non-negative submodular functions.

In sum, $\sigma(\cdot,t)$ is monotone and submodular for all $t \geqslant 0$. □

Algorithm 1. Greedy (G,k,T).

1: $S = \phi$
2: **for** $i = 1$ **to** k **do**
3: $\quad u = \arg\max_{v \in V\setminus S} \sigma(S \cup \{v\}, T) - \sigma(S,T)$
4: $\quad S = S \cup \{u\}$
5: **end for**
6: **return** S

Because $\sigma(\cdot,\cdot)$ under CT-IC model is monotone and submodular by Theorem 1 and it is trivially non-negative, Greedy algorithm (Algorithm 1) guarantees a $(1-1/e)$-approximation solution for influence maximization problem by Theorem 2.1 in [1]. Its time complexity is $O(knRmT)$ where n, m, R are the number of nodes, the number of edges, and the number of iterations of Monte Carlo simulation to get the approximation value of σ.

Difference between IC and CT-IC models. To investigate how different CT-IC model is from IC model in a specific situation, we now introduce a measure called *difference ratio between IC and CT-IC model* as follows.

Assume that $G = (V,E)$, k, and T are given. Define the set of optimal solutions for CT-IC model and that of IC model as $\mathfrak{S}_I^*(G,k) = \text{argmax}\{\sigma_I(S) | S \subseteq V, |S| = k\}$, $\mathfrak{S}_T^*(G,k) = \text{argmax}\{\sigma(S,T) | S \subseteq V, |S| = k\}$, respectively, where $\sigma_I(S)$ is the influence spread of seed set S in IC model. Then, we define the difference ratio as

$$dr(G,k,T) = \frac{\sigma(S_T^*,T)}{\max\{\sigma(S_I^*,T) | S_I^* \in \mathfrak{S}_I^*\}} \geqslant 1,$$

where $S_T^* \in \mathfrak{S}_T^*$. dr tells that whether we can get good solution for influence maximization under CT-IC model even if we just treat CT-IC model as IC model. This ratio can be used as a measure to quantify the difference between IC and CT-IC models. If CT-IC model is not much different from IC model, dr would be close to 1, otherwise, it might be greater than 1.

The following Lemma says that for small k, T, there exist infinitely many graphs for which dr is sufficiently large.

Lemma 1. *For any positive k, N, T such that $k < N/4$, $T < (N/4k) - 1 = O(N/k)$, there exists a graph $G = (V,E)$ such that $|V| = N$ and $dr(G,k,T) = \Omega(N/kT)$.*

Proof. For a given k, N and T, construct a graph $G = \bigcup_{i=1}^{k}\left(G_i^1 \cup G_i^2\right)$, where $G_i^1 = \left(V_i^1, E_i^1\right)$ is a star graph with $(N/2k)-1$ nodes, $G_i^2 = \left(V_i^2, E_i^2\right)$ is a simple path with $(N/2k)+1$ nodes. Set $pp_0(u,v) = 1$ for every $(u,v) \in E$.

Then, $\mathfrak{S}_I^* = \left\{\{v_1,\ldots,v_k\} | v_i \in V_i^2\right\}$ as $\sigma_I(\{v\}) = (N/2k)+1 > (N/2k)-1 = \sigma_I(\{v'\})$ for any $v \in V_i^2, v' \in V_j^1$. However, $\mathfrak{S}_T^* = \left\{\{v_1,\ldots,v_k\} | v_i \in V_i^1\right\}$ as $\sigma(\{v\},T) = (N/2k) - 1 > 2T + 1 \geqslant \sigma(\{v'\},T)$ for any $v \in V_i^1, v' \in V_j^2$. Therefore, $dr(G,k,T) = \frac{k[(N/2k)-1]}{k(2T+1)} = \Omega(N/kT)$. □

4. Exact computation of influence spread

In this section, we provide an exact influence evaluation under CT-IC model when a graph has special topology, arborescence or simple path. Because computing influence spread under IC model is #P-Hard [22] and IC model is a special case of CT-IC model, computing influence spread under CT-IC model is also #P-Hard. However, its computation is still tractable if we restrict the whole graph to an arborescence, a directed graph in which there exists a unique path from every node to a root node. We first present equations for computing influence spread in an arborescence (Section 4.1), and then by using these equations, give a useful way to evaluate influence spread for a simple path which is a special case of an arborescence (Section 4.2).

4.1. Case of an arborescence

Consider an arborescence $G_A = (V,E)$ with a seed set $S \subseteq V$ and time restriction T. For any $v \in V$ and $0 \leqslant t \leqslant T$, let $ap_S(v,t)$ be a probability that v is activated *exactly* at time t, and $ap_{S,T}(v)$ be a probability that v is activated before activation process ends (i.e. $ap_{S,T}(v) = \sum_{i=0}^{T} ap_S(v,i)$). Then, it is obvious that

$$ap_S(v,t) = \begin{cases} 1 & \text{if } v \in S \text{ and } t = 0 \\ 0 & \text{if } v \notin S \text{ and } t = 0 \\ 0 & \text{if } v \in S \text{ and } t > 0 \end{cases}.$$

However, when $v \notin S$ and $0 < t \leqslant T$, computing $ap_S(v,t)$ is not trivial. The following Lemma 2 tells that in this case, $ap_S(v,t)$ has a complex formula.

Lemma 2. *For any $v \in V \setminus S$ and $0 < t \leqslant T$,*

$$ap_S(v,t) = \prod_{u \in N_{in}(v)}\left[1 - \sum_{i=0}^{t-2} ap_S(u,i) g_{uv}(t-2-i)\right] - \prod_{u \in N_{in}(v)}\left[1 - \sum_{i=0}^{t-1} ap_S(u,i) g_{uv}(t-1-i)\right]$$

holds, where $g_{uv}(t) = 1 - \prod_{i=0}^{t}[1 - pp_i(u,v)]$.

Proof. Consider a node $v \in V$. For $0 < t \leqslant T$, let $A_t(v,u)$ be an event that v is activated by $u \in N_{in}(v)$ *exactly* at time t, and $NA_t(v)$ be an event that v is not activated *until* time t. Then, $NA_t(v) = \bigcap_{u \in N_{in}(v)}\left(\bigcap_{i=1}^{t}\overline{A_i(v,u)}\right) = \bigcap_{u \in N_{in}(v)}\left(\overline{\bigcup_{i=1}^{t} A_i(v,u)}\right)$. $\overline{\bigcup_{i=1}^{t} A_i(v,u)}$ and $\overline{\bigcup_{i=1}^{t} A_i(v,u')}$ are independent for any $u \neq u'$ because every activation trial is independent of each other. Thus, we have $\Pr[NA_t(v)] = \prod_{u \in N_{in}(v)} \Pr\left[\overline{\bigcup_{i=1}^{t} A_i(v,u)}\right]$.

Let $\Lambda_t(v)$ be an event that v is activated *exactly* at time t, and $\varepsilon_{uv}(i,j)$ be an event that given $\Lambda_i(u)$, edge $(u,v) \in E$ is activated *exactly* at time j (so v must be active at time $j+1$), for any $0 \leqslant i \leqslant j$.

Then, $A_i(v,u) = \cup_{j=0}^{i-1}(\Lambda_j(u) \cap \varepsilon_{uv}(j,i-1))$. We compute $\Pr\left[\cup_{i=1}^{t} A_i(v,u)\right]$ by using the fact that $\Lambda_j(u)$ and $\Lambda_{j'}(u)$ are mutually exclusive for any $j \neq j'$, and $\Pr[\varepsilon_{uv}(i,j)] = pp_{j-i}(u,v)$ as follows.

$$\begin{aligned}\Pr\left[\bigcup_{i=1}^{t} A_i(v,u)\right] &= \Pr\left[\bigcup_{i=1}^{t}\bigcup_{j=0}^{i-1}(\Lambda_j(u) \cap \varepsilon_{uv}(j,i-1))\right] \\ &= \sum_{j=0}^{t-1}\Pr\left[\Lambda_j(u) \cap \left(\bigcup_{i=j+1}^{t}\varepsilon_{uv}(j,i-1)\right)\right] \\ &= \sum_{j=0}^{t-1}\Pr[\Lambda_j(u)]\left(1 - \Pr\left[\overline{\bigcup_{i=j+1}^{t}\varepsilon_{uv}(j,i-1)}\right]\right) \\ &= \sum_{j=0}^{t-1} ap_S(u,j)\left(1 - \prod_{i=0}^{t-1-j}[1 - pp_i(u,v)]\right)\end{aligned}$$

So, $\Pr[NA_t(v)] = \prod_{u\in N_{in}(v)}\left[1 - \sum_{i=0}^{t-1} ap_S(u,i)g_{uv}(t-1-i)\right]$ and $ap_S(v,t) = \Pr[NA_{t-1}(v)] - \Pr[NA_t(v)]$ hold. □

We know that $\sigma(S,T) = \sum_{v\in V}\sum_{i=0}^{T} ap_S(v,i)$ holds. Therefore, when a given graph is an arborescence, we can compute the exact value of $\sigma(S,T)$ in a polynomial time. In fact, by using simple dynamic programming, $\sigma(S,T)$ is computed in $O(|V|T^2)$ time since computing $\Pr[NA_i(v)]$ for all $v \in V$ takes $O(|V|T)$ time for each $i = 0,\ldots,T$.

4.2. Case of a simple path

Let us consider the influence spread of a simple path p which is a sequence of nodes. For an edge (u,v) of p, by Lemma 2, the activation probability of v at time t, $ap(v,t)$ is the sum of the product of (1) the probability that u is activated at $i (0 \leqslant i < t)$ and (2) the probability that u activate v at the $(t-i)$th activation trial. Accordingly, $ap(v,t)$ is derived as follows.

$$ap(v,t) = \sum_{i=0}^{t-1} c_{uv}^{(t-i)} ap(u,i) = \begin{bmatrix} ap(u,0) \\ ap(u,1) \\ \vdots \\ ap(u,t-1)\end{bmatrix}^{\mathrm{Tr}} \begin{bmatrix} c_{uv}^{(t)} \\ c_{uv}^{(t-1)} \\ \vdots \\ c_{uv}^{(1)}\end{bmatrix},$$

where $c_{uv}^{(t-i)} = pp_{t-i-1}(u,v)\prod_{j=0}^{t-i-2}(1 - pp_j(u,v))$ which is the probability that u activates v at the $(t-i)$th trial. Obvious subscript S in $ap_S(v,t)$ is omitted. After putting $ap(v,i)$'s for $i = 0,\ldots,T$ into a matrix, we have

$$\begin{bmatrix} ap(v,0) \\ ap(v,1) \\ ap(v,2) \\ \vdots \\ ap(v,T)\end{bmatrix}^{\mathrm{Tr}} = \begin{bmatrix} ap(u,0) \\ ap(u,1) \\ ap(u,2) \\ \vdots \\ ap(u,T)\end{bmatrix}^{\mathrm{Tr}} \begin{bmatrix} 0 & c_{uv}^{(1)} & \cdots & c_{uv}^{(T)} \\ 0 & 0 & \cdots & c_{uv}^{(T-1)} \\ 0 & 0 & \cdots & c_{uv}^{(T-2)} \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 0\end{bmatrix},$$

or $\mathbf{AP}(v) = \mathbf{AP}(u)\mathbf{C}_{uv}$ equivalently, where $\mathbf{AP}(v)$, $\mathbf{AP}(u)$, and $\mathbf{C}_{uv}$ represent corresponding matrices.

Now, for any $u \in S$ and $v \in V \setminus S$, consider a simple path $p = (u = u_0, u_1, \ldots, u_{l-1}, u_l = v)$ where $u_i \in V \setminus S$ for all $i = 1,\ldots,l$. Suppose that influence is spread only through p (i.e. each u_{i+1} is activated only by u_i for all $i = 0,\ldots,l-1$). In this situation, define $inf_p(u,v)$ be the probability that u activates v in time T, i.e. $ap_{S,T}(v)$. By using the above result, we have $\mathbf{AP}(v) = \mathbf{AP}(u)\mathbf{C}_{u_0u_1}\cdots\mathbf{C}_{u_{l-1}u_l}$. However, we know that $\mathbf{AP}(u) = [1\ 0\ \cdots\ 0]$ and $ap_{S,T}(v) = \sum_{i=0}^{T} ap_S(v,i) = \mathbf{AP}(v)[1\ \cdots\ 1]^{\mathrm{Tr}}$. Therefore, we finally obtain the following Lemma.

Lemma 3. *The probability that $u \in S$ activates $v \in V \setminus S$ only through a path $p = (u = u_0, u_1, \ldots, u_{l-1}, u_l = v)$ is*

$$inf_p(u,v) = [1\ 0\ \cdots\ 0]\left(\prod_{i=0}^{l-1}\mathbf{C}_{u_iu_{i+1}}\right)[1\ 1\ \cdots\ 1]^{\mathrm{Tr}}, \tag{3}$$

where $u_i \in V \setminus S$ for all $i = 1,\ldots,l$, and the order of matrix multiplication is from $i = 0$ to $l-1$.

Let us consider the relationship between the above equation and the corresponding equation in IC model. In IC model, each edge has a *real* weight which represents propagation probability, and the probability that one node activates the other node only through a path is computed by multiplying each edge's *real* weight along the path. However, in CT-IC model, each edge has a $(T+1)\times(T+1)$ *matrix* weight, and the same probability is calculated by summing the first row of the matrix obtained by multiplying each edge's *matrix* weight along the path. Thus, we can think of the above Lemma as the generalized version of equation for IC model.

However, the existing influence approximation methods, which depends on the shortest path, such as MIA [6] for IC model and MIA-N [2] for IC-N model cannot be extended to CT-IC model. IC and IC-N models, the principle of optimality [23], which says that all sub-paths of any maximum probability path are also maximum probability paths, holds. Thus, we could make a reasonable local tree structure, such as MIA [6] and MIA-N [2], for efficient algorithms. However, as the below Lemma tells us, CT-IC model does not have such property. Therefore, obtaining similar local arborescences of MIA or MIA-N for CT-IC model is computationally intractable because shortest path algorithm such as Dijkstra's algorithm cannot be used for finding maximum probability path.

For any $u,\ v \in V$, define p^* be a *maximum probability path* from u to v if $p^* \in \arg\max_p\{inf_p(u,v) | p :$ a simple path from u to $v\}$. By using this definition and Lemma 3, we give one more property of CT-IC model, described in the following lemma.

Lemma 4. *In CT-IC model, the principle of optimality does not hold.*

Proof. Let us prove the lemma using a counter example in which the principle of optimality is violated.

An example graph is given as Fig. 2. Assume $pp_0(e_0) = pp_0(e_1) = 0.6$, $pp_0(e_2) = pp_0(e_3) = 0.3$, $\alpha_{u_i} = 1$ $(i = 0,\ldots,3)$, $S = \{u_0\}$, and $T = 3$. Then, (u_0,u_1,u_2,u_3) is the maximum probability path from u_0 to u_3. However, one of its sub-paths, (u_0,u_1,u_2), is not the maximum probability path form u_0 to u_2. In fact, (u_0,u_2) is the maximum probability path from u_0 to u_2. □

5. Influence spread processing algorithm

From the existing works [6,17–21], we know that the greedy algorithm for IC model is very slow in practice due to the heavy calculation of $\sigma(S)$. So, it is obvious that Greedy algorithm for CT-IC model is absolutely not scalable. We need a new scalable algorithm for CT-IC model. Although PMIA algorithm [6] is one of state-of-the-art algorithms for IC model, it is hard to generalize it to CT-IC model as described in Section 4.2.

In this section, we propose *Continuously activated and Time-restricted influence path algorithm* (CT-IPA) for CT-IC model by extending a highly scalable algorithm for IC model – independent path algorithm (IPA) [7]. We first describe how IPA works briefly and then demonstrate several treatments for extending IPA into CT-IPA.

IPA evaluates influence spread of seed nodes by considering an independent influence path as a basic unit of influence spread evaluation. The #P-hardness of influence spread evaluation is based on the fact that we cannot find all paths between any two nodes in a

tractable time. Thus, IPA scales up influence spread evaluation by controlling the number of influence paths which amounts to dropping out negligible influence paths which have propagation probability less than a pre-defined threshold θ. In addition, for scalable evaluation of influence spread, IPA assumes influence paths are independent of each other.

he extension from IPA to CT-IPA is seamlessly done by changing the influence spread definition of an influence path. In IPA for IC model, influence spread of an influence path is obtained by multiplying real-valued propagation probability of each edge in the path. In CT-IC model, influence spread of an influence path is $inf_p(\cdot,\cdot)$ of Eq. (3) which involves matrix multiplication. Therefore, embedding $inf_p(\cdot,\cdot)$ into IPA, we get CT-IPA algorithm for CT-IC model.

Let us define *critical paths* starting from node u as $P_u = \{p \in SP_u | p = (u,\dots,v),\ v \in V \setminus \{u\}, inf_p(u,v) \geqslant \theta\}$, where $SP_u = \{\text{simple paths starting from } u\}$. Then, critical path set from node u to v is defined by $P_{u\to v} = \{p \in P_u | p = (u,\dots,v)\}$. $P_{u\to v}$ means that u activates v through one of the paths in $P_{u\to v}$. Finally, *influenced area* of node u is defined by $O_u = \{v | (u,\dots,v) \in P_u\}$. By using these definitions and the above assumptions, influence spread is approximated in CT-IPA as follows.

$$\widehat{ap}_{\{u\},T}(v) = 1 - \prod_{p \in P_{u\to v}} (1 - inf_p(u,v)) \tag{4}$$

$$\hat{\sigma}(\{u\},T) = 1 + \sum_{v \in O_u} \widehat{ap}_{\{u\},T}(v) \tag{5}$$

Note that by considering critical paths as influence spread evaluation units, only paths in $P_{u\to v}$ are considered in Eq. (4), and by independence between critical paths, $\widehat{ap}_{\{u\},T}(v)$ has an explicit and simple formula Eq. (4).

To compute the influence spread of a seed set, we define critical paths from a seed set S as $P_S = \{p | p \in P_u, u \in S, p \cap S = \{u\}\}$, and critical paths from a seed set S to a specific node v as $P_{S\to v} = \{p \in P_S | p = (u,\dots,v)\}$. Finally, define influenced area of a seed set S as $O_S = \{v | (u,\dots,v) \in P_S\}$. Then, the influence spread of S is computed as follows.

$$\widehat{ap}_{S,T}(v) = 1 - \prod_{p \in P_{S\to v}} (1 - inf_p(u,v)) \tag{6}$$

$$\hat{\sigma}(S,T) = |S| + \sum_{v \in O_S} \widehat{ap}_{S,T}(v) \tag{7}$$

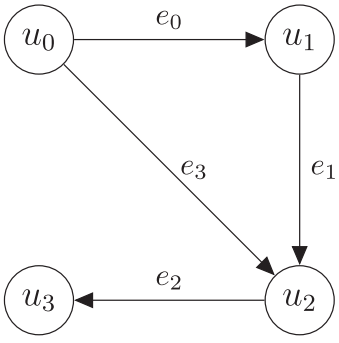

Fig. 2. A counter example that violates the principle of optimality.

Table 1
Basic information of four real dataset.

Dataset	HEP	PHY	EPINION	AMAZON
Directedness	Undir	Undir	Dir	Dir
# of Nodes	15 K	37 K	76 K	262 K
# of Edges	59 K	232 K	509 K	1235 K
# of Connected components	1781	3883	2	1
Average size of components	8.6	9.6	38 K	262 K
θ for CT-IPA	1/32	1/64	1/64	1/16

Algorithm 2. CT-IPA(G,k,T,θ).

Input: G: a graph, k: a required size of a seed set, T: time restriction, θ : a threshold controlling the size of a local structure
Output: seed set of size k
1 /* Initialize */
2 **for** $u,\ v \in V$ **do** $P_{u\to v} = O_u = \phi$
3 **for** $u \in V$ **do**
4 compute P_u with T and θ
5 **for** $p = (u,\dots,v) \in P_u$ **do**
6 $P_{u\to v} = P_{u\to v} \cup \{p\}$
7 $O_u = O_u \cup \{v\}$
8 **end**
9 compute $\Delta_u = \hat{\sigma}(\{u\},T)$ /* by using Eqs. (3)–(5) */
10 **end**
11 /* Greedy Loop */
12 $S = \phi$
13 **for** $i = 1$ **to** k **do**
14 $v = \arg\max_{u \in V-S} \Delta_u$
15 $S = S \cup \{v\}$
16 **for** $u \in V - S$ **do** $\Delta_u = \text{Calc} - \Delta(S,u)$
17 **end**
18 **return** S

Putting the above equations together, we get CT-IPA (Algorithm 2). While the basic structure of CT-IPA is greedy algorithm, influence spread is computed more efficiently by the above equations. Line 4 is easily done by BFS (breath-first search) starting from node u. Line 9 is computed by Eqs. (3)–(5) with O_u, $P_{u\to v}$, which are obtained in lines 5–8. Lines 13–17 are the loop of greedy algorithm.

Algorithm 3. Calc-$\Delta(S,u)$.

Input: S: selected seed nodes until now, u: a node
Output: $\hat{\sigma}(S \cup \{u\},T) - \hat{\sigma}(S,T)$
1 $\Delta_u = 1$
2 **for** $v \in O_u$ **do**
3 $new\_ap = cur\_ap = 1$
4 **for** $p \in P_{u\to v}$ with $p \cap S = \phi$ **do**
5 $new\_ap *= (1 - inf_p(u,v))$
6 **end**
7 **for** $s \in S$ **do**
8 **for** $p \in P_{s\to v}$ with $p \cap S \subseteq \{s,u\}$ **do**
9 $old\_ap *= (1 - inf_p(s,v))$
10 **if** $p \cap S = \{s\}$ **then**
11 $new\_ap *= (1 - inf_p(s,v))$
12 **end**
13 **end**
14 $\Delta_u += (1 - new\_ap) - (1 - old\_ap)$
15 **end**
16 **return** Δ_u

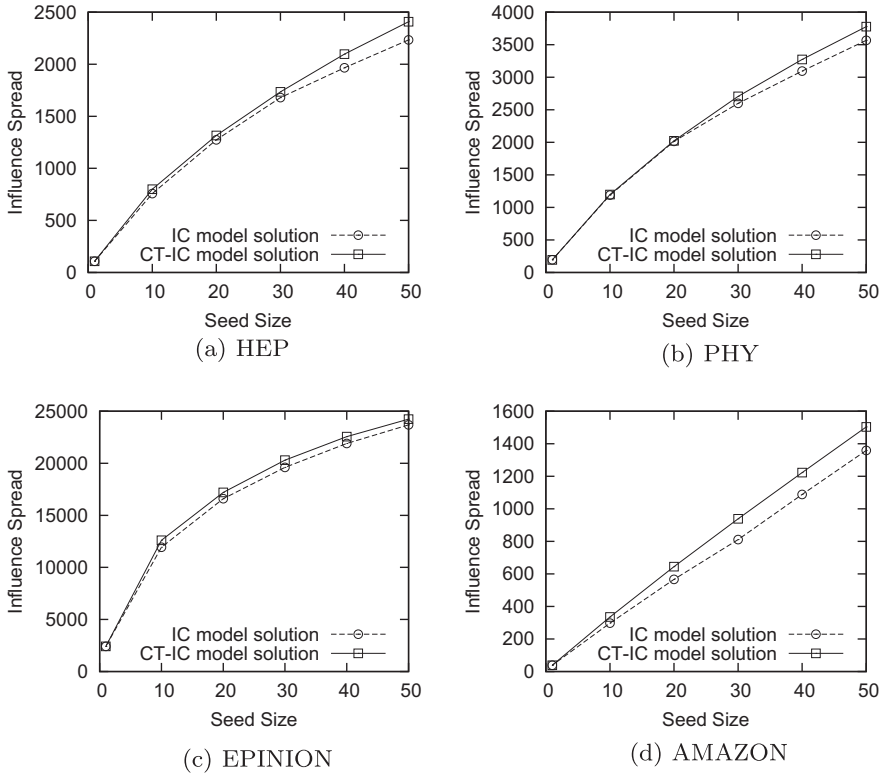

Fig. 3. Comparison between IC and CT-IC models.

Calc-Δ (Algorithm 3) outlines how to compute $\hat{\sigma}(S\cup\{u\},T)-\hat{\sigma}(S,T)$ which is used in line 16 of Algorithm 2. In Calc-Δ, $(1-new\_ap)$ and $(1-old\_ap)$ finally equal to $\widehat{ap}_{S\cup\{u\},T}(v)$ and $\widehat{ap}_{S,T}(v)$ in line 14, respectively. Line 5 is the case that a path from u to v is not *blocked* by nodes in S. Similarly, line 9 is the case that a path p from a seed node s to v is not blocked by other seed nodes, and line 11 is the case that the path p is also not blocked by u.

Merging multiple edges. To reduce the processing time of CT-IPA, merging multiple edges into a single edge is required as a pre-processing task. However, unlike IC model, this task is not obvious.

Suppose that we have multiple edges $e_1,\ldots,e_l$ from node u to v. In IC model, these multiple edges are equivalent to a single edge e' with propagation probability $1-\prod_{i=1}^{l}(1-pp_0(e_i))$. However, in CT-IC model, this is not the case. Let e' be an equivalent edge to these multiple edges in CT-IC model. Then, we have $pp_t(e')=1-\prod_{i=1}^{l}(1-pp_t(e_i))$, and it is not the form of $c\cdot f_{uv}(t)$ of Eq. (2) with constant c. It means that we cannot merge multiple edges into a single one e' with a constant weight $pp_0(e')$. Fortunately, CT-IPA only requires $\mathbf{C}_{uv}$ instead of $pp_0(e')$ being constant, and $\mathbf{C}_{uv}$ can be computed by using $pp_t(e')$. Thus, we can merge multiple edges into a single edge having a matrix weight $\mathbf{C}_{uv}$, which is quite different from IC model.

Time complexity. First, computing the multiplication of two matrix weights, $\mathbf{C}_{uv}\mathbf{C}_{vw}$, takes only $O(T^2)$ time because both matrices are upper triangular and the elements of each diagonal of each matrix has the same value. Therefore, computing P_u and $P_{u\to v}$ for all possible u,v takes $O(nn_pT^2)$ time, where $n=|V|$, n_p is the average number of critical paths starting from each node. Next, it takes $O(|O_S|n_p)$ time to calculate $\hat{\sigma}(S,T)$. This is because, we have to look up $|O_S|$ nodes $u\in O_S$ (Eqs. (5) and (7)), and for each u, we have to look up $|P_{S\to u}|$ paths (Eqs. (4) and (6)). Thus, calculating $\hat{\sigma}(S\cup\{v\},T)-\hat{\sigma}(S,T)$ for all $v\in V\setminus S$ (line 3 in Algorithm 1) takes $O(nn_on_p)$ time, where n_o is the average number of influenced nodes of each node. To sum up, the time complexity of the CT-IPA integrated greedy algorithm is $O(nn_pT^2+knn_on_p)=O(nn_p(kn_o+T^2))$.

Table 2
Top-20 seed nodes of IC model and CT-IC model solution.

On PHY				
IC model solution				
4840	1568	5192	5120	7387
12,081	2356	10,653	4115	23,571
3460	3808	969	809	5567
2443	3566	5312	6342	3673
CT-IC model solution				
4840	5192	5120	1568	809
4115	2356	3460	23,571	12,081
7132	**3842**	10,653	**4109**	3673
6342	**3712**	**2928**	**3982**	**2289**
On AMAZON				
IC model solution				
17,747	222,839	25,699	18,076	168,039
18,337	232,448	7266	11,129	45,391
176,067	9657	64,815	183,084	27,562
59,541	14,461	238,375	114,241	1385
CT-IC model solution				
17,747	176,067	**56,415**	**51,234**	**200,657**
238,375	18,076	**236,670**	**259,011**	222,839
6290	**205,434**	**143,531**	**199,539**	**59,541**
25,699	**178,335**	**82,533**	**114,241**	**95,315**

6. Experiments

In this section, we conduct experiments to figure out characteristic of CT-IC model and to compare the performance of CT-IPA with other algorithms. Specifically, the goal of our experiments is

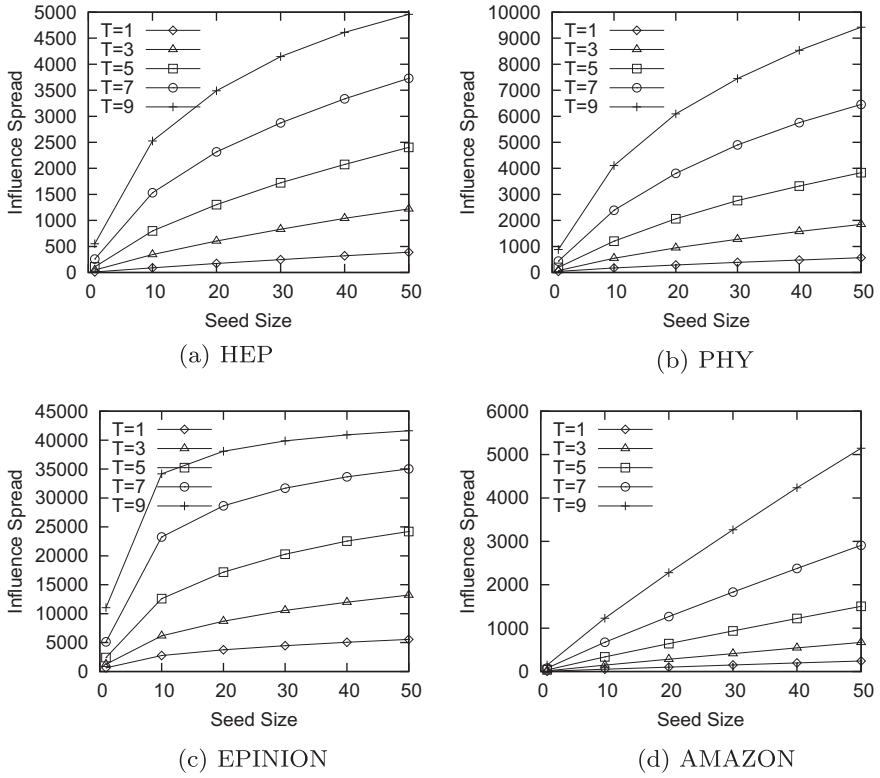

Fig. 4. The change of influence spread with respect to T.

twofold: (a) to check how much CT-IC model is different from IC model, we compare seed set and its influence spread of CT-IC model with those of IC model, and measure the change of influence spread of CT-IC model solution with respect to T (Section 6.2) and (b) to compare CT-IPA with other algorithm, processing time and influence spread are measured (Section 6.3).

6.1. Experiment setup

Datasets. We chose four widely used real datasets in influence maximization problem. HEP and PHY are co-authorship graphs obtained from "High Energy Physics - Theory" and "Physics" section of arXiv site (http://arxiv.org) where nodes and edges represent authors and coauthor relationships, respectively. EPINION is a who-trust-whom graph of `epinions.com`, where a node u represents a user of the site and an edge (u, v) represents that v trusts u, so there is chance for u to influences v. AMAZON is a co-purchasing graph of `amazon.com` on March 2, 2003, in which a node u represents a product and an edge (u, v) represents that v is usually bought with u, u may influence v. We get HEP, PHY data from Wei Chen's site,[2] and EPINION, AMAZON from Stanford's SNAP site.[3] The basic statistics of each graph is presented in Table 1 where EPINION and AMAZON are considered as undirected graphs.

Propagation probabilities. Since propagation probabilities are not available on our data set, we use WC (weighted cascade) model [1] for generating edges' probabilities. In WC model, propagation probabilities are assigned as $pp_0(u, v) = 1/\deg_{in}(v)$ for all edges $(u, v) \in E$, where $\deg_{in}(v)$ denotes the in-degree of node u.

Algorithms. In Section 6.3, we compare CT-IPA algorithm with the other algorithms. We do not include any algorithms for IC model because they are not extendable to CT-IC model as described in Section 4.2.

- Random: A baseline algorithm which selects k nodes uniformly at random from the overall $|V|$ nodes.
- MaxDegree: A simple heuristic algorithm which selects k nodes in non-increasing order of node's out-degree.
- Greedy: Algorithm 1 with lazy-forward optimization [17]. We use $R = 10{,}000$, where R denotes the number of iterations for Monte-Carlo simulation to compute $\sigma(S, T)$.
- CT-IPA: Our proposed algorithm integrated with lazy-forward greedy optimization. The last row of Table 1 shows tuned θ values used on each dataset.[4]

In this experiment, we set $\alpha_v = 0.1$ for all v. Different α values produced similar results. When we calculate the influence spread of each seed set produced by each algorithm, we do 10,000 Monte-Carlo simulations and get the average of the values. We conduct the following experiments in a Linux machine with two Intel Xeon CPUs and 24 GB memory.

6.2. Characteristic of CT-IC model

Comparison between IC and CT-IC models. We show that CT-IC model is a novel influence diffusion model by comparing CT-IC model to IC model in both quantitative and qualitative ways. In order to check whether CT-IC model is novel compared to IC model, we run the greedy algorithm under "IC model" ($\text{Greedy}_{\text{IC}}$) and the greedy algorithm under "CT-IC model" ($\text{Greedy}_{\text{CT-IC}}$). In the experiment, we vary seed size k from 1 to 50, and set $T = 5$. Note that, since it is not feasible to get a solution by greedy algorithms for large graphs (EPINION, AMAZON), we use IPA [7] and CT-IPA instead of greedy algorithm for IC and CT-IC models, respectively.

[2] http://research.microsoft.com/en-us/people/weic/graphdata.zip.
[3] http://snap.stanford.edu/data.

[4] We find that there is trade-off between processing time and influence spread as θ changes. Thus, by varying $\theta = 1/8, 1/16/, \ldots, 1/512$, we select the first θ at which an increment in influence spread becomes much smaller than that in processing time.

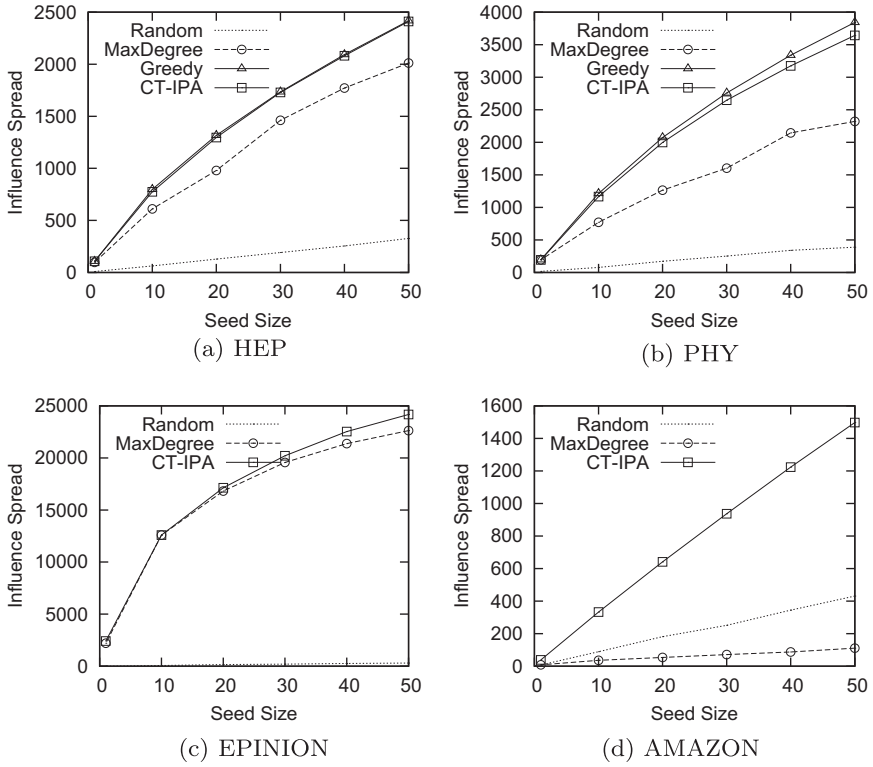

Fig. 5. Influence spread of various algorithms.

To show the difference between two models quantitatively, after obtaining two difference seed sets from $\text{Greedy}_{\text{IC}}$ and $\text{Greedy}_{\text{CT-IC}}$, we compare the influence spreads of them under **"CT-IC model"**. Fig. 3 shows the influence spread of two methods' solutions on four datasets. On HEP, PHY and EPINION, the influence spread of CT-IC model solution is always larger than that of IC model solution, and moreover the gap between them becomes larger as k increases. On much larger graph AMAZON, the similar result is obtained. However, the gap between CT-IC and IC model is much larger than that on other graphs. These results show that (1) CT-IC model is a different model from IC model and (2) time constraint and continuous activation trials of CT-IC model are meaningful consideration for a realistic influence diffusion model.

One thing to note is that even though the difference ratio between IC model and CT-IC model,*dr*, is close to 1 on HEP, PHY and EPINION, it does not mean that our CT-IPA algorithm is unnecessary. The reason is that we cannot know that IC model solution really works well in CT-IC model before computing the optimal solution for CT-IC model. Moreover, there exist cases where *dr* becomes very large like Lemma 1.

To show the difference between two models qualitatively, we compare elements of two seed node sets which are obtained by $\text{Greedy}_{\text{IC}}$ and $\text{Greedy}_{\text{CT-IC}}$. In this comparison, we select the first 20 seed nodes under IC model and CT-IC model, which are identified by $Greedy_{IC}$ and $Greedy_{CT-IC}$, respectively. The results on PHY and AMAZON are listed in Table 2. In the node id list, the top-left node is top-1st node of solution and the bottom-right node is the top-20th node, and node ids in bold type are ones which are included in CT-IC model solution but not in IC model solution. Among top-20 nodes, only 13 and 6 nodes are in common for both solutions on PHY and AMAZON, respectively. Moreover, the ranking of top-20 nodes in CT-IC model solution is largely different from that in IC model solution. Thus, CT-IC model is a more different model from IC model than it appears in Fig. 3.

To sum up, we conclude that there exists definite distinction between CT-IC model and IC model even though they are sometimes superficially similar in terms of influence spread.

Change of influence spread when varying T. To find out how influence spread changes as T increases, influence spread is measured when $T = 1, 3, \ldots, 9$. k is also varied from 1 to 50. We select seed nodes by Greedy for small graphs (HEP, PHY) and by CT-IPA for large graphs (EPINION, AMAZON).

Fig. 4 illustrates the results of influence spread on four datasets. On every dataset, influence spread increases as T increases, which is an obvious result. However, as T increases, the increment of influence spread *increases* at first, and then starts to *decrease* at some point (on HEP and EPINION) or does not decrease at all (on PHY and AMAZON). The fact that the increment of influence spread *increases* is not intuitive but can be explained as follows.

Let $\Delta[\sigma](S,T) = \sigma(S,T+1) - \sigma(S,T)$ and $\Delta^2[\sigma](S,T) = \Delta[\sigma](S,T+1) - \Delta[\sigma](S,T)$. In this notation, the above statement is almost equivalent to that "$\Delta^2[\sigma](S,T)$ is at first positive but becomes negative at some point as T increases." In fact, two statements are not exactly equivalent since seed sets for each T are slightly different in our experiment. However, for simplicity, let us assume they are all equal for every T. There are two opposite effects on the sign of $\Delta^2[\sigma]$ – the effects of *already* active nodes and *newly* activated nodes. The nodes that are already active at T activate less nodes as time goes by because pp_t keep decreasing. Accordingly, such nodes try to make $\Delta^2[\sigma]$ negative. On the other hand, the nodes that are newly activated at $T+1$ have just started to activate other nodes. Because they are not active at T, their activation tries only increase $\Delta[\sigma](S,T+1)$. Therefore, they try to make $\Delta^2[\sigma]$ positive. By this argument, we can now explain the above observation – $\Delta^2[\sigma] < 0$ (resp. > 0) because the first effect (resp. the second one) is stronger than the other.

Knowing when $\Delta^2[\sigma](S,T)$ becomes negative or positive is very important for viral marketing. Suppose that a company plans to release product A and B at time step 0 and T, respectively. For a viral

marketing of product A, seed nodes S_A are selected by the influence maximization and product A succeeds in making big profit until T. Based on the success of product A, the company decides to delay the release of product B. However, if $\Delta^2[\sigma](S_A, T) < 0$, the company's expectation of product A's steady success is wrong because after T influence spread increment is diminishing. The opposite situation also holds.

6.3. Comparison between algorithms

Influence spread. We measure the influence spread of algorithms' solutions on four datasets by varying k from 1 to 50. We set $T = 5$. Greedy is only applied to HEP and PHY because of its excessive processing time on EPINION and AMAZON. The results are shown in Fig. 5.

On HEP, the influence spread of CT-IPA is almost close to that of Greedy. Also, there is a significant gap between CT-IPA and MaxDegree. Random is the worst one, which tells that randomly selecting seed nodes is not a good idea like in IC model [6].

On PHY, the result is almost the same as on HEP. The only difference is that MaxDegree, Random produce much smaller influence spread compared to Greedy and CT-IPA.

On EPINION, two interesting facts are observed. Unlike on the other graphs, MaxDegree almost matches CT-IPA when $k \leq 40$, and the influence spread of Random is almost 0. We guess that this result happens because EPINION has very few influential nodes, which has very high degree, and almost every node is not such influential and has low degree. Actually, the maximal degree of EPINION (3079) is the highest among our data set.

Finally, on AMAZON, CT-IPA is still overwhelmingly the best, and the influence spread of CT-IPA is almost linear to k, like in IC model [6]. However, in this case, the influence spread of MaxDegree is even much smaller than Random, which is completely opposite to EPINION case. The reason is the topology of AMAZON is quite different from that of EPINION. The influential nodes in AMAZON have not very high degree while high degree nodes in AMAZON may have very low propagation probabilities to their neighbor nodes.

In a nutshell, CT-IPA yields influence spread as high as Greedy, and always shows better influence spread than other algorithms. Additionally, MaxDegree is very unstable. Though it performs well in few cases, it does not in other cases and is sometimes worse than Random.

Processing time. We measure the processing time for all combinations of four algorithms and four datasets. In this experiment, we retrieve top-50 seed nodes while fixing $T = 5$. Fig. 6 shows the processing time of four algorithms on four datasets where the y-axis is log-scaled. Note that we ran each experiment up-to 10 h.

For all datasets, Greedy take the longest time to find seed nodes. Even on small datasets, Greedy, the processing times of it are 5.0 h on HEP and 10.0 h on PHY. On large dataset (EPINION and AMAZON), Greedy fail to provide seed node because it does not finish before 10 h of running. Thus, as in IC model, although Greedy identifies seed nodes which has better influence spread, it is not applicable to large datasets due to its poor scalability.

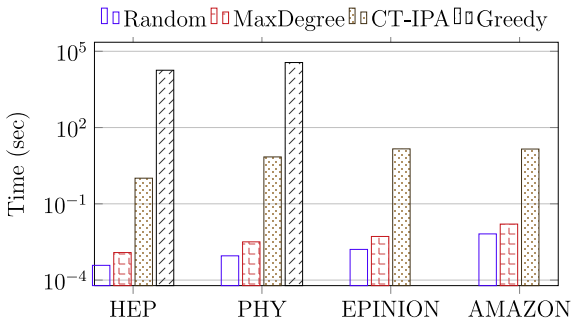

Fig. 6. Processing time of various algorithms.

On the other hand, CT-IPA takes less than 15 s in all datasets. Specifically, CT-IPA takes 1.0, 7.0, 14.5, and 14.3 s on HEP, PHY, EPINION, and AMAZON. Compared to Greedy, CT-IPA shows four orders of magnitude shorter processing time. Such efficient processing of CT-IPA comes from considering critical paths as influence evaluation unit of CT-IPA. For every $\sigma(S, T)$ evaluation, while Greedy requires 10,000 times of fresh Monte-Carlo simulation, CT-IPA reuses critical paths and saves the processing time.

Since MaxDegree and Random do not consider the influence diffusion, they always take less than one second. However, influence spread of their solutions is unstable and much worse than CT-IPA.

7. Conclusion

In this paper, we propose a realistic influence diffusion model – the time-considering independent cascade (CT-IC) model. Existing influence diffusion models and their efficient processing algorithms lack of two important aspects of influence propagation in real world – time constraint and continuous activation trials. CT-IC model embeds these two aspects into its activation process to reflect more realistic influence diffusion in social networks. By proving monotonicity and submodularity, the greedy algorithm which has $1 - 1/e$ approximation ratio can be applied to CT-IC model. Moreover, exact influence spread evaluation in CT-IC for a specific graph (e.g. arborescences and simple paths) are derived. By plugging the exact influence spread evaluation of simple paths to IPA algorithm for IC model, we have a highly scalable processing algorithm CT-IPA for CT-IC model. Extensive experiments on real datasets show that CT-IC model produces different results from IC model, and CT-IPA produces seed sets several orders of magnitude faster than the greedy algorithm without sacrificing influence spread.

Acknowledgement

This research was partially supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. 2012M3C4A7033344). This work was also supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No. 2013R1A2A2A01067425).

References

[1] D. Kempe, J. Kleinberg, E. Tardos, Maximizing the spread of influence through a social network, in: KDD '03: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, NY, USA, 2003, pp. 137–146. ISBN 1-58113-737-0.

[2] W. Chen, A. Collins, R. Cummings, T. Ke, Z. Liu, D. Rincón, X. Sun, Y. Wang, W. Wei, Y. Yuan, Influence maximization in social networks when negative opinions may emerge and propagate, in: SDM, 2011, pp. 379–390.

[3] X. He, G. Song, W. Chen, Q. Jiang, Influence blocking maximization in social networks under the competitive linear threshold model, in: SDM, 2012, pp. 463–474.

[4] A. Borodin, Y. Filmus, J. Oren, Threshold models for competitive influence in social networks, in: Proceedings of the 6th International Conference on Internet and Network Economics, WINE'10, Springer-Verlag, Berlin, Heidelberg, 2010, pp. 539–550. ISBN 3-642-17571-6, 978-3-642-17571-8. <http://dl.acm.org/citation.cfm?id=1940179.1940229>.

[5] Y. Li, W. Chen, Y. Wang, Z.-L. Zhang, Influence diffusion dynamics and influence maximization in social networks with friend and foe relationships, in: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, WSDM '13, ACM, New York, NY, USA, 2013, pp. 657–666. ISBN 978-1-4503-1869-3. http://dx.doi.org/10.1145/2433396.2433478.

[6] W. Chen, C. Wang, Y. Wang, Scalable influence maximization for prevalent viral marketing in large-scale social networks, in: KDD '10: Proceedings of the 16th

ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, NY, USA, 1957, pp. 1029–1038. ISBN 978-1-4503-0055-1.

[7] J. Kim, S.-K. Kim, H. Yu, Scalable and parallelizable processing of influence maximization for large-scale social network, in: Proceedings of the 2013 IEEE 29th International Conference on Data Engineering, ICDE '13, IEEE Computer Society, Washington, DC, USA, 2013, pp. 266–277.

[8] P. Clifford, A. Sudbury, A model for spatial conflict, Biometrika 60 (3) (1973) 581–588.

[9] A. Goyal, F. Bonchi, L.V. Lakshmanan, Learning influence probabilities in social networks, in: Proceedings of the Third ACM International Conference on Web Search and Data Mining, WSDM '10, ACM, New York, NY, USA, 2010, pp. 241–250. ISBN 978-1-60558-889-6. http://dx.doi.org/10.1145/1718487.1718518.

[10] K. Saito, R. Nakano, M. Kimura, Prediction of information diffusion probabilities for independent cascade model, in: Proceedings of the 12th International Conference on Knowledge-Based Intelligent Information and Engineering Systems, Part III, KES '08, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 67–75. ISBN 978-3-540-85566-8. http://dx.doi.org/10.1007/978-3-540-85567-5_9.

[11] Z. Ghahramani, Learning dynamic Bayesian networks, in: Adaptive Processing of Sequences and Data Structures, Springer, 1998, pp. 168–197.

[12] B.-E. Perrin, L. Ralaivola, A. Mazurie, S. Bottani, J. Mallet, F. d'Alche Buc, Gene networks inference using dynamic Bayesian networks, Bioinformatics 19 (suppl 2) (2003) ii138–ii148.

[13] G. Zweig, S. Russell, Speech recognition with dynamic Bayesian networks, in: AAAI, 1998, pp. 173–180.

[14] N. Oliver, E. Horvitz, A comparison of HMMs and dynamic Bayesian networks for recognizing office activities, in: User Modeling 2005, Springer, 2005, pp. 199–209.

[15] K.P. Murphy, Dynamic Bayesian Networks: Representation, Inference and Learning, Ph.D. Thesis, University of California, 2002.

[16] E. Mossel, S. Roch, On the submodularity of influence in social networks, in: Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing, STOC '07, ACM, New York, NY, USA, 2007, pp. 128–134. ISBN 978-1-59593-631-8. http://dx.doi.org/10.1145/1250790.1250811.

[17] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, N. Glance, Cost-effective outbreak detection in networks, in: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '07, ACM, New York, NY, USA, 2007, pp. 420–429. ISBN 978-1-59593-609-7.

[18] A. Goyal, W. Lu, L.V. Lakshmanan, CELF++: optimizing the greedy algorithm for influence maximization in social networks, in: Proceedings of the 20th International Conference Companion on World Wide Web, WWW '11, ACM, New York, NY, USA, 2011, pp. 47–48. ISBN 978-1-4503-0637-9. http://dx.doi.org/10.1145/1963192.1963217.

[19] W. Chen, Y. Wang, S. Yang, Efficient influence maximization in social networks, in: KDD '09: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, NY, USA, 2009, pp. 199–208. ISBN 978-1-60558-495-9.

[20] M. Kimura, K. Saito, Tractable models for information diffusion in social networks, in: J. Fürnkranz, T. Scheffer, M. Spiliopoulou (Eds.), Knowledge Discovery in Databases: PKDD 2006, Lecture Notes in Computer Science, vol. 4213, Springer, Berlin/Heidelberg, 2006, pp. 259–271.

[21] Y. Wang, G. Cong, G. Song, K. Xie, Community-based greedy algorithm for mining top-K influential nodes in mobile social networks, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '10, ACM, New York, NY, USA, 2010, pp. 1039–1048, 2010. ISBN 978-1-4503-0055-1.

[22] W. Chen, Y. Yuan, L. Zhang, Scalable influence maximization in social networks under the linear threshold model, in: Proceedings of the 2010 IEEE International Conference on Data Mining, ICDM '10, IEEE Computer Society, Washington, DC, USA, 2010, pp. 88–97. ISBN 978-0-7695-4256-0.

[23] R. Bellman, Dynamic Programming, first ed., Princeton University Press, Princeton, NJ, USA, 1957.