*Article*

JIS

Journal of Information Science
2018, Vol. 44(5) 671–682

DOI: 10.1177/0165551517748289
journals.sagepub.com/home/jis
SAGE

# Influence maximisation in social networks: A target-oriented estimation

**Yun-Yong Ko**
Department of Computer and Software, Hanyang University, Korea

**Dong-Kyu Chae**
Department of Computer and Software, Hanyang University, Korea

**Sang-Wook Kim**
Department of Computer and Software, Hanyang University, Korea

**Abstract**
Influence maximisation (IM) is the problem of finding a set of *k*-seed nodes that could maximize the amount of influence spread in a social network. In this article, we point out that the existing methods are taking the *source-oriented estimation* (SOE), which is the main reason of their failure in accurately estimating the amount of potential influence spread of an individual node. We propose a novel *target-oriented estimation* (TOE) that understands information diffusion more accurately as well as remedies the drawback of the existing methods. Our extensive experiments on four real-world datasets demonstrate that our proposed method outperforms the existing methods consistently with respect to the quality of the selected seed set.

**Keywords**
Influence maximisation; information diffusion; social network analysis

## 1. Introduction

With an increasing number of users exploiting online social networks (OSN) such as Facebook and Twitter, a viral marketing based on the word-of-mouth effect (i.e. a person's decision to buy a product is often strongly influenced by her friends, acquaintances and business partners) in OSN has attracted a lot of interest these days [1–3]. For example, in order to promote its new product, a company may carefully select some *influential users* in OSN, provide free samples to them and let them post positive reviews about the product, expecting its positive influence to spread over the entire social network [4,5]. Along this line, it is important to select truly influential users so as to maximize the amount of influence spread within a limited budget. Formally, this problem is called *influence maximisation* (IM), which is to find a *k*-seed set that incurs the maximum influence spread [6–13]. Here, a *k*-seed set indicates a set of *k* nodes that initiate influence propagation (i.e. those who receive free samples); the influence is propagated in accordance with an information diffusion model (e.g. *Independent Cascade* (IC) model and *Linear Threshold* (LT) model); the amount of influence spread by a seed set corresponds to the number of non-seed nodes eventually getting activated by the seed set.

Kempe et al. [6] established that finding the optimal solution to IM is non-deterministic polynomial-time (NP) hardness. They also presented a greedy algorithm (henceforth, referred to as *SimpleGreedy*) that gradually selects a seed node maximising the influence spread at every step, eventually making up *k*-seed nodes, which guarantees up to 63% of influence spread by the optimal solution. However, SimpleGreedy still suffers from the low efficiency due to the following two problems with respect to the micro and macro levels. At the macro level, after selecting one seed node at each step, SimpleGreedy needs to re-evaluate the influence spread of *every* non-seed node because their influence spread can be

**Corresponding author:**
Sang-Wook Kim, Department of Computer and Software, Hanyang University, 222 Wangsimni-ro, Seongdong-gu, Seoul 133-791, Korea.
Email: wook@hanyang.ac.kr

reduced (i.e. changed) when a seed was selected among them in the previous step. At the micro level, SimpleGreedy estimates the influence spread of a node by running Monte-Carlo (MC) simulations 10,000 times, which is also a very time-consuming task.

A number of studies have been done to tackle such performance issues [14–23]. Aiming at the macro-level problem, Leskovec et al. [14] proposed cost-effective lazy forward (CELF) optimisation that enormously reduces the number of re-evaluations on non-seed nodes required after selecting one seed node by exploiting the sub-modularity property of IM's objective function. It remedies the macro-level problem significantly; however, it still requires much time when applied to large-scale graphs due to the micro-level problem (i.e. the costly MC simulations). SIMPATH [15] and IPA [16] have been proposed to remedy the micro-level problem. Rather than MC simulations, these methods exploit the weight (i.e. influence probability) of every path starting from each seed in order to estimate the influence spread of a seed set. Hereafter, we call this category of methods *path-based methods*. As a result, the path-based methods are an order of magnitude faster than SimpleGreedy while providing comparable accuracy in IM.

In this article, we point out the problem with existing path-based methods in estimating influence spread of a seed set with respect to the accuracy of influence estimation and propose an approach to address the problem. Recall that the $k$-seed set of SimpleGreedy is considered as a ground truth in evaluating other approximate IM algorithms; SimpleGreedy defines the influence spread of a seed set as *the number of non-seeds* (i.e. *targets*) activated by the seed set (i.e. *sources*). In this sense, the influence spread of the seed set should be evaluated from the *target nodes' perspective*, more specifically, based on *the total amount of influence that every target node receives*.

Here, how to aggregate all the influence to an individual target node from multiple neighbours should be dependent on diffusion models employed. However, when aggregating the influence, existing path-based methods just take a simple linear sum regardless of diffusion models since they first compute the influence from each individual seed node (i.e. source) to all other nodes and then summate the influence from all the seed nodes. We claim that this causes existing path-based methods to estimate influence spread incorrectly under *some* diffusion model.

Motivated by the problem with existing path-based methods, referred to as *source-oriented estimation* (SOE) approach, we propose a novel *target-oriented estimation* (TOE) approach that remedies the drawback of SOE. TOE (1) *aggregates* the amount of influence each individual target (non-seed) node receives from a whole set of seed nodes and (2) takes the linear sum of those of all target nodes. Here, in step (1), TOE takes different schemes to aggregate the influence received by an individual target node from multiple neighbours, according to diffusion models. In addition, this approach is conceptually equivalent to that of SimpleGreedy, which estimates the influence spread based on the total amount of influence received by every target node from the whole seed nodes. We thus expect that TOE results in more accurate computation of influence spread than SOE.

We demonstrate how our TOE is applied to each of IC and LT diffusion models. We show that TOE and SOE perform in the exactly *identical* way under the LT model when computing the influence spread of a seed set, but in a *different* way under the IC model. Our extensive experiments on five real-world datasets demonstrate that our TOE consistently outperforms SOE as well as other IM algorithms in terms of the quality (i.e. the amount of influence spread) of the derived $k$-seed set, activating up to 1200 more nodes than SOE. Also, the additional time cost of our TOE compared with SOE is shown to be insignificant.

The rest of this article is organized as follows. Section 2 introduces the IM problem in a social network and reviews its related studies. Section 3 points out the problem of existing path-based methods and section 4 presents our TOE approach to remedy the problem. We show our experimental results in section 5 and finally conclude this article in section 6.

## 2. Preliminaries

### 2.1. IM problem

Kempe et al. [6] formulated the IM as a discrete optimisation problem as follows.

#### Definition 1

*IM*. Given a network $G$ having $n$ nodes and a limited budget $k$, it is to find a set $S$ consisting of $k$ users, called a seed set, that maximizes $\sigma(S)$, which corresponds to *influence spread* over $G$

$$S = \text{argmax}_{|S|=k}\sigma(S) \tag{1}$$

To solve the IM problem, we need to have a diffusion model that describes how influence spreads over the network. The LT model and the IC model are widely used diffusion models [6]. Both the two models have the common rules as follows:

1. Nodes can have either of two states, active or inactive.
2. As time goes by, inactive nodes can be activated, but active nodes cannot become inactive.
3. The diffusion process is finished if any nodes do not become active state.

In the LT model, each node has its own random threshold and gets activated if the cumulative influence given from its active neighbouring nodes becomes larger than the threshold. In the IC model, each node receives influence from each of its active neighbouring nodes independently and its activation depends on the weight of the edge from each neighbouring node. The weight indicates the activation probability.

Finding the optimal solution to the IM problem is NP-hard because it is required to compare the influence spreads of all possible ${}_nC_k$ $k$-seed sets $S$ from $n$ nodes in a social network. Kempe et al. [6] proposed SimpleGreedy, which gradually picks up a new seed node $v$ maximising the marginal gain (i.e. $\sigma(S+\{v\})-\sigma(S)$) in each step and repeats this step $k$ times to find $k$-seed nodes. This greedy solution is proven to guarantee 63% of influence spread obtained by the optimal solution if $\sigma(\cdot)$ is *non-negative, monotone* and *submodular* [6]. A function $\sigma(\cdot)$ is monotone if $\sigma(S)\leq\sigma(T)$ whenever $S\subset T$, and it is submodular if $\sigma(S+\{v\})-\sigma(S)>\sigma(T+\{v\})-\sigma(T)$ for all $S\subset T$.

### 2.2. Existing solutions

As explained in the previous section, however, SimpleGreedy still suffers from the low efficiency in both micro and macro levels. A number of studies have been done to improve the performance of finding influential nodes in social networks. Several methods have been proposed to address the macro-level problem [14,17–23]. For example, CELF performs a 'lazy-forward' optimisation in picking up a new seed to greatly reduce the number of re-evaluations of influence spread by exploiting the *submodular* property of the objective function. CGA and INCIM exploit the property of a community structure in a network [24] (i.e. the nodes in the same community are densely connected but those in different communities are sparsely connected) to re-evaluate the influence spread of only those nodes (i.e. rather than all the nodes) in the community where a new seed was selected in the previous step.

However, path-based methods have been proposed to address the micro-level problem [15,16,23]. The main idea behind these methods is to aggregate the weights for all the paths starting from a seed node in estimating its influence spread, rather than running costly MC simulations. Formally, the influence spread of a node $v$ is defined by the following equation in path-based methods

$$\sigma(\{v\})=\sum_{p\in\text{paths from }v}W_p \tag{2}$$

In equation (2), $v$, $p$ and $W_p$ indicate a node, a path starting from $v$ and a weight of path $p$, respectively. Here, $W_p$ is computed by multiplying the weights on all the edges included in $p$ (i.e. $W_p=\prod_{e\in p}w_e$); $p$ should be acyclic and should include only one seed node according to the assumption of diffusion models; therefore, paths including loop or multiple seeds are ignored in estimating influence spread of a set of nodes. The problem of finding all possible paths, however, is #P-hard [25]. Thus, existing methods prune a path $p$ if its weight becomes smaller than a pre-defined threshold with the intuition that as nodes are located farther, less influence diffusion appears between the nodes [15,16]. After influence spread of each seed node, the influence spread of a seed set $S$ is computed by the linear sum of all seeds' influence spread as in equation (3) where $s$ indicates a seed node

$$\sigma(S)=\sum_{s\in S}\sigma(\{s\}) \tag{3}$$

The path-based methods achieve an order of magnitude speed up compared with SimpleGreedy while providing comparable accuracy.

## 3. Motivation

In this section, we point out the problem with the approach of existing path-based methods in estimating influence spread of a seed set with respect to the accuracy of influence estimation, which becomes our motivation. As shown in equations

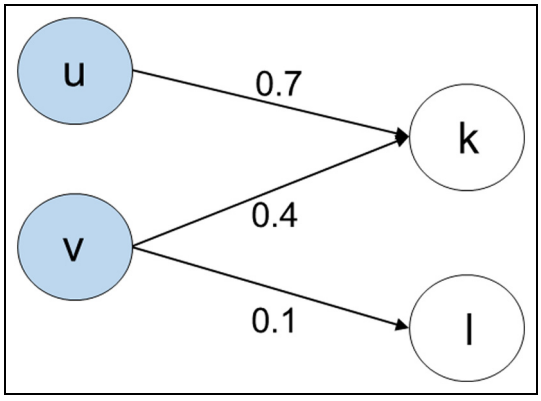


**Figure 1.** A toy example.

(2) and (3), the path-based methods (1) compute $\sigma(\{v\})$, indicating the amount of influence spread each individual seed $v$ (i.e., *source*) gives over all the non-seed nodes, and then (2) linearly summate $\sigma(\{v\})$ of all the seed nodes. Indeed, their approach is SOE: they estimate the influence spread of the seed set based on how much individual source (i.e. seed) node propagates influence to target (i.e. non-seed) nodes.

However, we claim that such an SOE approach should be modified since SimpleGreedy, which the path-based methods try to follow, takes an approach opposed to such an SOE approach when estimating influence spread: SimpleGreedy defines the influence spread of a seed set as the number of *non-seed* nodes (i.e. *targets*) influenced by the seed set; this indicates that *the total amount of influence* (i.e. *aggregated influence*) received by *target nodes* determines the influence spread of the seed set, which we call as TOE. Moreover, in the SOE approach, the total amount of influence that a target node receives is just computed by the linear sum, which should be different according to diffusion models. As a result, SOE cannot estimate the influence spread of a seed set correctly under some diffusion models that do not employ the linear sum as the aggregation scheme.

For example, as described in Figure 1, suppose $u$ and $v$, which are active nodes, propagate influence to a non-seed node $k$ with the weights 0.7 and 0.4, respectively. Here, SOE estimates the influence spread of the seed set as follows: $\sigma(S) = \sigma(\{u\}) + \sigma(\{v\})$. In other words, the amount of influence received by node $k$ becomes 1.1 regardless of diffusion models. Under the LT model, fortunately, the amount of influence received by node $k$ is equivalent to the linear sum of 0.7 and 0.4, which is the same with the SOE's scheme. Under the IC model, however, the amount of influence received by node $k$ should be computed as $1 - (1 - 0.7)(1 - 0.4) = 0.82$, rather than 1.1 by the linear sum, because $u$ and $v$ influence node $k$ *independently*. In the following section, we propose our TOE approach that can fundamentally solve the problem with SOE regardless of diffusion models.

## 4. The proposed approach

The proposed TOE estimates the amount of influence spread of a seed set as follows: (1) to aggregate the amount of influence received by each individual non-seed node (i.e. target) from the whole seed set and (2) to summate those of all non-seed nodes linearly. To this end, we first define $\sigma_d(S)$, the aggregated value of influence that a non-seed node, $d$, receives from all the seed nodes $S$. Here, the aggregation scheme should be dependent on diffusion models, namely, the LT or IC models.

Under the LT model, according to its property, the influences from all the seed nodes towards a given target node are summated linearly. Therefore, a target node $d$'s aggregated influence, $\sigma_d(S)$, is computed as follows

$$\sigma_d(S) = \sum_{p \in P_{S \to d}} W_p \tag{4}$$

In equation (4), $d$, $P_{S \to d}$ and $W_p$ indicate a non-seed node reachable from $S$, a set of paths from $S$ to $d$ and a weight of path $p$, respectively. Next, we add up $\sigma_d(S)$ of all the target nodes $d \in \{V - S\}$, in order to get the total amount of influence spread by the seed set, as follows

$$\sigma(S) = \sum_{d \in V-S} \sigma_d(S) \tag{5}$$

Under the LT model, we note that SOE and TOE compute the amount of influence spread of a seed set in an *identical* way. This is because the amount of influence towards a node is a linear sum of influences from its neighbouring nodes in the LT model. Therefore, the seed sets obtained from SOE and TOE should be identical as well.

*Lemma 1*. TOE and SOE compute the amount of influence spread of a seed set in the same way under the LT model.

*Proof for Lemma 1*. $\sigma(S)_{TOE}$ and $\sigma(S)_{SOE}$ are influence spread of seed set $S$ by TOE and SOE, respectively

$$\begin{aligned}
\sigma(S)_{TOE} &= \sum_{d \in V-S} \sigma_d(S) = \sum_{d \in V-S} \sum_{p \in P_{S \to d}} W_p \\
&= \sum_{p \in P_{S \to d_1}} W_p + \sum_{p \in P_{S \to d_2}} W_p + \cdots + \sum_{p \in P_{S \to d_n}} W_p \\
&= \left(W_{P_{s_1 \to d_1}} + W_{P_{s_2 \to d_1}} + \cdots + W_{P_{s_k \to d_1}}\right) + \cdots + \left(W_{P_{s_1 \to d_n}} + W_{P_{s_2 \to d_n}} + \cdots + W_{P_{s_k \to d_n}}\right) \\
&= \left(W_{P_{s_1 \to d_1}} + W_{P_{s_1 \to d_2}} + \cdots + W_{P_{s_1 \to d_n}}\right) + \cdots + \left(W_{P_{s_k \to d_1}} + W_{P_{s_k \to d_2}} + \cdots + W_{P_{s_k \to d_n}}\right)
\end{aligned} \tag{6}$$

$$\begin{aligned}
\sigma(S)_{SOE} &= \sum_{s \in S} \sigma(\{s\}) = \sum_{s \in S} \sum_{p \in P_{s \to V-S}} W_p \\
&= \sum_{p \in P_{s_1 \to V-S}} W_p + \sum_{p \in P_{s_2 \to V-S}} W_p + \cdots + \sum_{p \in P_{s_k \to V-S}} W_p \\
&= \left(W_{P_{s_1 \to d_1}} + W_{P_{s_1 \to d_2}} + \cdots + W_{P_{s_1 \to d_n}}\right) + \cdots + \left(W_{P_{s_k \to d_1}} + W_{P_{s_k \to d_2}} + \cdots + W_{P_{s_k \to d_n}}\right)
\end{aligned} \tag{7}$$

$$\therefore \sigma(S)_{TOE} = \sigma(S)_{SOE} \tag{8}$$

Next, under the IC model where the influences of all the seed nodes towards a target node are considered independently, $\sigma_d(S)$ is computed as follows

$$\sigma_d(S) = 1 - \prod_{p \in P_{s \to \mathrm{d}}} (1 - W_p) \tag{9}$$

Again, by adding up $\sigma_d(S)$ of all the target nodes, we get the total amount of influence spread of the seed set in the same way as equation (5). Under the IC model, we note that TOE and SOE compute the amount of influence spread of a seed set in *different* ways or producing different seed sets. As pointed out before, this is due to the difference in the viewpoints in computations of influence spread: TOE aggregates the total amount of influence received by all the target nodes (i.e. from the target nodes' perspective) while SOE does the amount given by source nodes (i.e. from the source nodes' perspective).

As a result, TOE successfully follows the philosophy of diffusion models when aggregating influences received by a target node from source nodes, while SOE has no chance to follow the philosophy but *always sums up those influences linearly*.

*Lemma 2*. Under the IC model, the total amounts of influence spread of a seed set estimated by TOE and SOE could be different. TOE estimates the influence spread correctly considering the rule of the IC model.

*Proof for Lemma 2*. We prove Lemma 2 by showing the following counter example.

In Figure 1, a two-seed set $S = \{u, v\}$ influences target nodes $k$ and $l$. First, seed $u$ influences node $k$ with a probability of 0.7 and seed $v$ influences nodes $k$ and $l$ with probability of 0.4 and 0.1, respectively. The influence spread of $S$ by SOE and TOE is computed as follows

$$\sigma(S)_{SOE} = \sum_{s \in S} \sigma(\{s\}) = \sigma(\{u\}) + \sigma(\{v\}) = 0.7 + 0.5 = 1.2 \tag{10}$$

$$\sigma(S)_{TOE} = \sum_{d \in V-S} \sigma_d(S) = \sigma_k(\{u, v\}) + \sigma_l(\{u, v\}) = 1 - (1 - 0.7)(1 - 0.4) + 0.1 = 0.92 \tag{11}$$

$$\therefore \sigma(S)_{TOE} \neq \sigma(S)_{SOE} \tag{12}$$

Note that in Figure 1, both nodes $u$ and $v$ try to influence node $k$ at once. In the case of SOE, however, influences of $u$ and $v$ towards $k$ are not considered *independently*; rather, the linear sum of the two influences is taken regardless of the IC model. As a result, the aggregation property of the IC model is not preserved, which makes the influence spread incorrectly predicted. In contrast, our TOE computes the amount of influence spread correctly by aggregating influence received by target nodes as performed in the IC model.

In summary, TOE and SOE compute the amount of influence spread by a seed set in an *identical* way under the LT model but in *different* ways under the IC model. In the next section, we compare TOE with SOE in terms of both accuracy and efficiency.

Algorithm 1 shows the whole process of selecting a $k$-seed set using the TOE. For the first seed, marginal gains (each of which is denoted as *u.mg*) of all nodes are calculated by the TOE and the nodes are inserted into $Q$ in descending order of their marginal gains (lines 2–5). Then, the top node with the largest marginal gain is selected as the first seed and removed from $Q$ (lines 6–7). After a new seed node is selected, the CELF algorithm is applied to select the next seed node (lines 8–17): the marginal gain of the top node is recalculated; if the value is greater than that of the next node, the top node is selected as the next seed. Otherwise, $Q$ is reordered. This process is repeated until the size of the seed set is $k$.

**Algorithm 1.** The process of the seed selection.

**Input**: network *G(V, E)*, seed size *k*
**Output**: a seed set *S*
1: $S \leftarrow \emptyset$; $Q \leftarrow \emptyset$;
2: **for each** $u \in V$ **do**
3: $u.mg = \sigma(S + \{u\})_{TOE} - \sigma(S)_{TOE}$;
4: Add $u$ to $Q$;
5: **end for**
6: $u =$ top node in $Q$;
7: $S \leftarrow S \cup \{u\}$; $Q \leftarrow Q - \{u\}$
8: **while** $|S| < k - 1$ **do**
9: $u =$ top node in $Q$;
10: $v =$ next node in $Q$;
11: $u.mg = \sigma(S + \{u\})_{TOE} - \sigma(S)_{TOE}$;
12: **if** $u.mg > v.mg$ then
13: $S \leftarrow S \cup \{u\}$; $Q \leftarrow Q - \{u\}$
14: **else**
15: Heapify $Q$;
16: **end if**
17: **end while**
18: **return** $S$;

## 5. Evaluation

In this section, we evaluate the effectiveness of our approach with five real-world datasets. The purpose of our experiments is to answer the three questions in the following:

1. What is the optimal pruning threshold for each dataset?
2. Does our TOE provide more accurate results than those of SOE as well as other IM algorithms?
3. How much time does our TOE spend compared with SOE?

**Table 1.** Dataset statistics

| | Number of nodes | Number of edges | Maximum in-degree | Maximum out-degree | Average degree |
|---|---|---|---|---|---|
| NetHEPT | 15K | 58K | 341 | 341 | 7.7 |
| NetPHY | 37K | 231K | 286 | 286 | 12.4 |
| Epinion | 75.8K | 508K | 3032 | 1798 | 6.7 |
| Stanford | 281K | 2.31M | 38,606 | 255 | 8.2 |
| DBLP | 655K | 3.98M | 588 | 588 | 6.1 |

**Table 2.** The optimal pruning thresholds of datasets

| NetHEPT | NetPHY | Stanford | Epinion | DBLP |
|---|---|---|---|---|
| 1/160 | 1/80 | 1/160 | 1/320 | 1/160 |

## 5.1. Experimental setup

*Dataset*. NetHEPT, NetPHY and DBLP are datasets of a co-authorship network consisting of authors and their co-authorships [6,23]; a node represents an author of a paper and an edge does the existence of co-work between the two authors. Epinion [5] is a dataset of a customer trust network where a node represents an individual customer and an edge does a customer's trust to another customer. Stanford [26] is a web graph where a node represents a web page and an edge does a hyperlink between two pages. Table 1 provides the statistics of the five datasets.

*5.1.1. Diffusion model.* As explained in the previous section, TOE and SOE compute the amount of influence spread by a seed set in the same way under the LT model, thus derive the same seed set in every case. Therefore, we conducted all the experiments only under the IC model. In our evaluation, we exploited the *weighted cascade* (WC) model [6], which is a widely used variation of the IC model. It assigns a propagation probability to an edge $(u, v)$ by $w_{u,v} = 1/d_{in}(v)$, where $d_{in}(v)$ is the in-degree of a node $v$. The WC model aims to penalise the nodes more with a higher in-degree since the nodes of higher in-degree cause them to have higher probability to be chosen as a seed.

*5.1.2. Algorithms.* For evaluation, we compared the following algorithms for seed selections: *Random* selects nodes randomly for seeds (a baseline). MC-Greedy is the same as SimpleGreedy [6]. Single degree discount (SDD) selects nodes of the highest degree (whenever a node is selected as a seed, the degree of all its neighbours decreases by 1) [27]. SOE is the existing path-based method. Finally, TOE is our proposed approach.

## 5.2. Q1: Optimal pruning threshold

In this experiment, before evaluating the effectiveness of TOE, we try to find the optimal pruning threshold for each dataset. A pruning threshold indicates the minimum of the weight on a path to be considered in path-based influence spread estimation: if there is a path whose weight is smaller than the threshold, the path is ignored when the influence spread is estimated. Intuitively, as the threshold gets lower, the influence spread of a seed set becomes larger but requires more time to be computed. To find the optimal threshold, we select seed sets using our TOE multiple times while varying the threshold from 1/20 to 1/640. Then, we compute the influence spread of each seed set by 10,000 times of MC simulations. Since existing work already provided the optimal pruning thresholds for DBLP, Epinion and Stanford datasets [16], we conducted this experiment only for NetHEPT and NetPHY datasets.

Figure 2(a) and (b) shows the results, representing relationship between seed selection time and influence spread according to the different pruning thresholds, where the $x$-axis indicates the seed selection time and the $y$-axis does the total amount of influence spread (i.e. the number of nodes activated by the seed set). Among various candidate values for the optimal pruning threshold, we selected 1/160 and 1/80 for NetHEPT and NetPHY, respectively. We finally used the pruning thresholds summarised in Table 2 for the rest of our experiments.

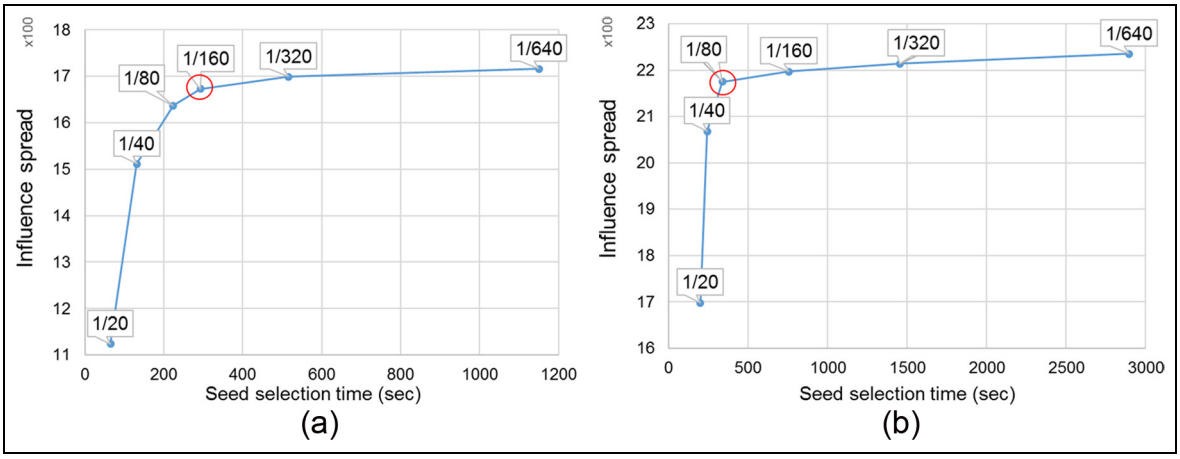


**Figure 2.** The pruning thresholds of (a) NetHEPT and (b) NetPHY.

## 5.3. Q2: Influence spread estimation

In this experiment, we compare the amount of influence spread of our TOE with those of existing IM algorithms summarised in section 5.1. To this end, for each dataset, we first selected 100 seeds obtained by different IM algorithms. Then, we ran 10,000 MC simulations for each seed set and took the average of the amount of influence spread.

Figure 3 (a)–(e) shows the experimental results on the total amount of influence spread ($y$-axis) according to the size of a seed set ($x$-axis) for different datasets. Among the four algorithms, *Random* provides the lowest influence spread. SDD provides influence spread higher than that of *Random* but lower than those of SOE and TOE. Our TOE universally shows the biggest influence spread in all the five datasets.

The difference in the amounts of influence spread from the seed sets obtained by the TOE and the SOE is more clearly shown in Figure 4 (a)–(e). The $x$-axis represents the size of the seed set and the $y$-axis represents ($T$-$S$) where $T$ and $S$ indicate the amounts of influence spread of the TOE and the SOE, respectively. Note that the values of the $y$-axis in both graphs show positive values in all cases, which indicates that our TOE consistently outperforms the SOE at any size of seed sets. Quantitatively, our method using the TOE selected a $k$-seed set with greater influence spread up to 37, 35, 176, 1202 and 1216 for NetHEPT, NetPHY, Epinion, Stanford and DBLP datasets, respectively. These results confirm that our TOE computes influence spread in a much more elaborated way and thus finds a more accurate result of top $k$-seeds that provides larger influence spread over a whole network. In addition, even though the differences in influence spread between the TOE and the SOE look insignificant, especially in the NetHEPT, NetPHY and Epinion datasets, we can observe their meaningful difference in Stanford and DBLP datasets, where graphs are large enough to be similar in size to real-world networks. Moreover, as the size of the seed set increases, their difference in influence spread tends to be larger (i.e. the larger the size of a seed set is, the more the TOE outperforms the SOE). This implies that our proposed TOE is fairly effective in those graphs with sizes of a real-world network.

## 5.4. Q3: Seed selection time

In this experiment, we compare the computational efficiency of TOE with those of existing algorithms. We perform the seed selection on a Windows 7 64-bit operating system equipped with 3.3-GHz Intel Core 2 Quad CPUs and 8 GB RAM. We report the experimental results in Figure 5, where the $y$-axis indicates consumed time (in seconds) for selecting 100 seeds by each algorithm.

Although Random and SDD provide high efficiency, seed sets selected by them provide low-influence spread as shown in the previous sub-section. MC-Greedy is the best performer in terms of the influence spread of a seed set, and thus, it is generally used as a ground truth [6,15,16]. However, it is computationally inefficient; for example, in DBLP dataset which consists of 655K nodes and 3.98M edges, it consumes more than a week for selecting just a 100-seed set. In spite of its influence spread gain, such a long processing time is a great burden from the business perspective.

However, the TOE and the SOE are an order of magnitude faster than MC-Greedy, consuming around an hour for selecting a 100-seed set. It is because the TOE and the SOE exploit path-based influence estimation rather than running the costly MC simulations. Here, the TOE spends about 22 more minutes on average than the SOE because the TOE

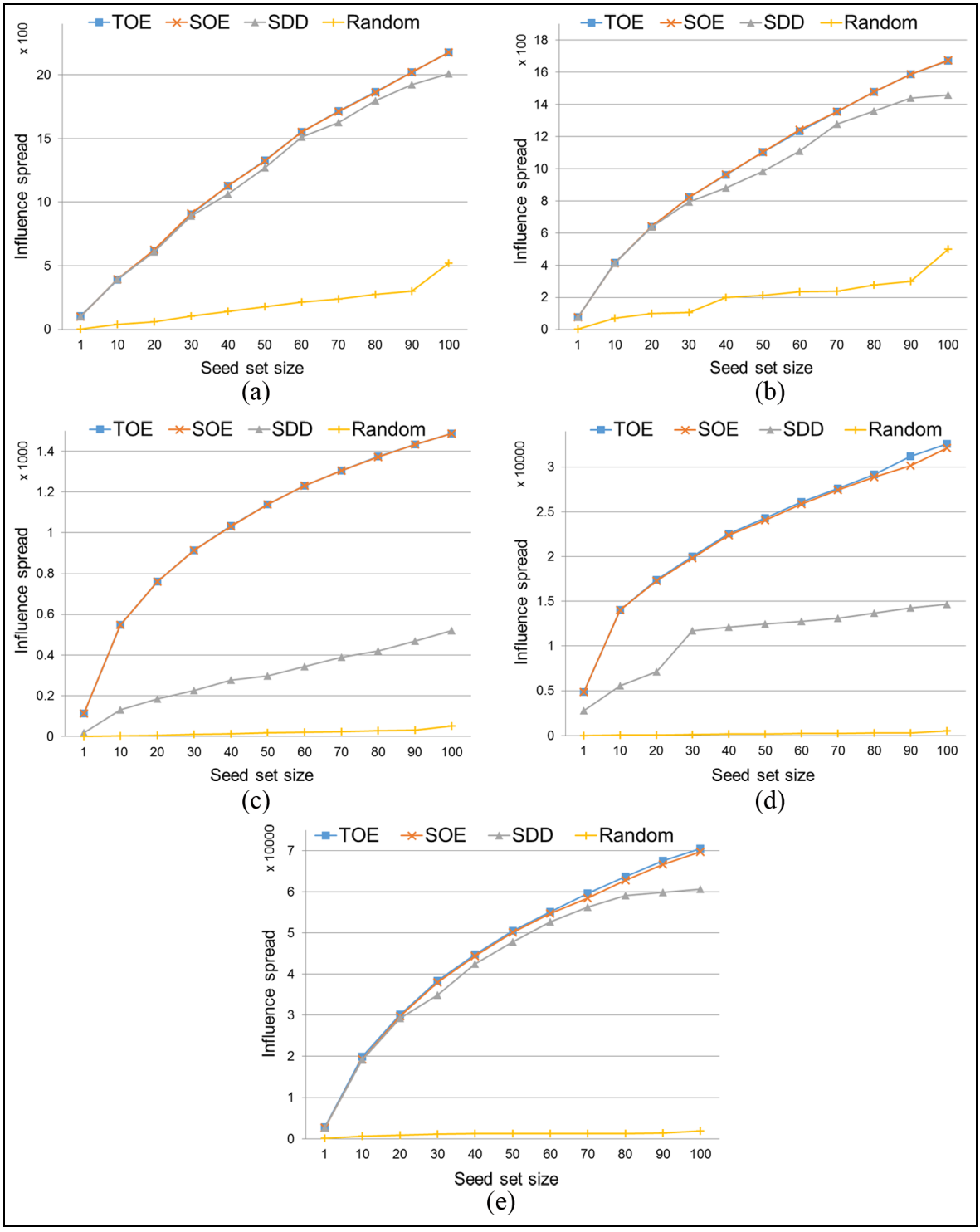


**Figure 3.** Influence spread of the seed set by each algorithm: (a) NetHEPT, (b) NetPHY, (c) Epinion, (d) Stanford and (e) DBLP.

requires additional computations which update influence received by each target when a new seed is selected. This is because aggregated influences of target nodes are changed by the influence from a new seed to targets in the TOE unlike the SOE. Nevertheless, regarding more influence spread (e.g. promotional effect) gained by our TOE, this loss of 22 min would not be a big deal for real-world businesses. In addition, once our TOE has derived a $k$-seed set, then the TOE can

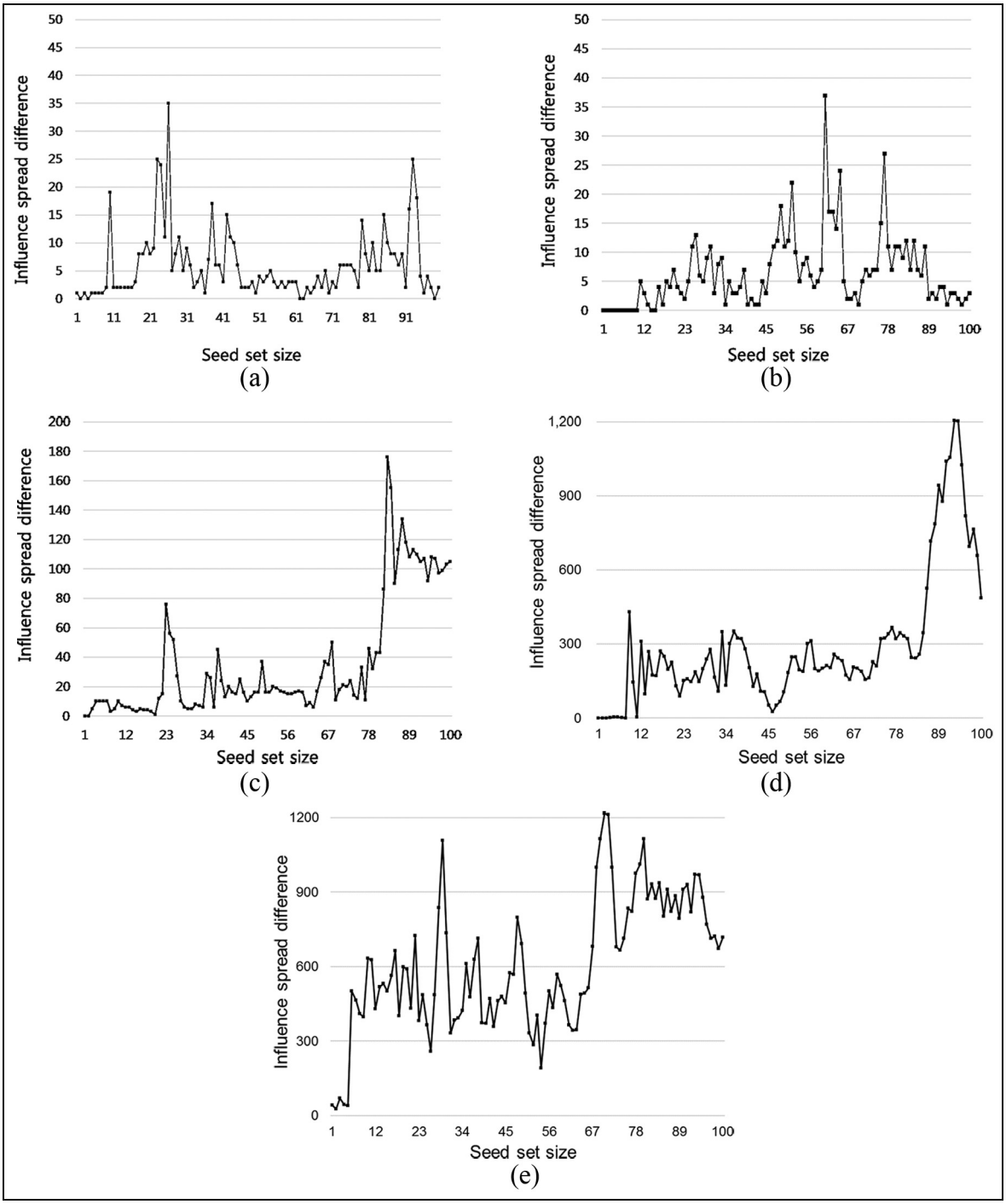


**Figure 4.** The difference in influence spread between TOE and SOE: (a) NetHEPT, (b) NetPHY, (c) Epinion, (d) Stanford and (e) DBLP.

efficiently find additional seeds (if needed) based on the pre-found $k$-seed set, which consumes less than $1/k$ of the total running time, rather than selecting the seed set from the scratch.

## 6. Conclusion

IM is to find the most influential nodes (i.e. seed nodes) in social networks. Finding the optimal set of seed nodes is known NP-hard. While a simple greedy algorithm (*SimpleGreedy*) based on MC simulations was proposed, it still

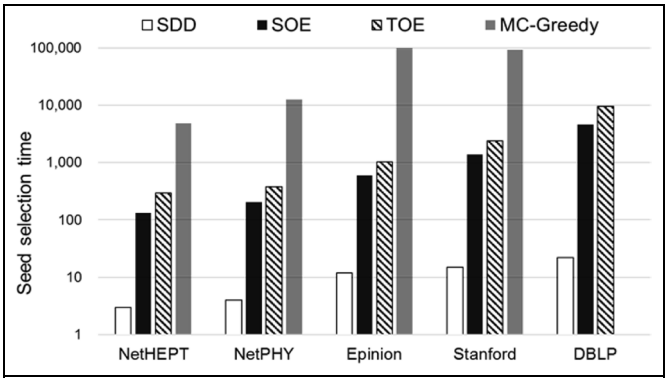


**Figure 5.** Seed selection time (s).

suffers from performance issues in micro and macro levels. A lot of studies have been done to address performance issues of SimpleGreedy. The path-based methods successfully resolve the performance issue in the micro level by estimating the influence spread of nodes by aggregating the weights of paths from the nodes rather than running costly MC simulations, but they do not estimate the influence spread of seed nodes accurately compared with SimpleGreedy.

In this article, we showed that the existing path-based methods are an SOE approach and pointed out the problem with SOE which estimates the influence spread of a seed set by taking the linear sum of all the seeds' influence in the seed nodes' (i.e. source node) perspective. Such SOE is opposed to SimpleGreedy whose results are considered as the ground truth because SimpleGreedy defines the influence spread of a seed set as the number of non-seed nodes influenced by the seed set; this indicates that the total amount of influence received by target nodes determines influence spread of the seed set. To remedy the problem of SOE, we proposed a novel TOE approach. Our TOE aggregates the amount of influence received by each individual non-seed node (i.e. target node) from a whole set of seed nodes and then adds up those of all non-seed nodes linearly. This approach enables TOE to consider the property of a diffusion model in the aggregation step, while SOE does not. Our experimental results demonstrate that TOE outperforms SOE as well as existing IM algorithms in terms of the quality of the seed set on four real-world datasets, while TOE maintains comparable running time with SOE. Moreover, the difference in the seed set quality between TOE and SOE tends to be bigger as the size of a seed set gets larger and the seed set selected by TOE activated up to 1200 more nodes than that by SOE.

Our work on TOE mainly focuses on estimating the influence spread of nodes more accurately than existing IM algorithms. Our TOE has a drawback of requiring higher computational overhead, leading to performance degradation. As our future work, we plan to tackle the performance issue to make our TOE not only effective but also efficient.

## Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship and/or publication of this article.

## Funding

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (Ministry of Science, ICT & Future Planning; No. NRF-2017R1A2B3004581).

## References

[1] Kwon Y, Kim S, Park S et al. The information diffusion model in the blog world. In: *Proceedings of the 3rd workshop on social network mining and analysis*, Paris, 28 June 2009, article no. 4.

[2] Kwon Y, Kim S and Park S. An analysis of information diffusion in the blog world. In: *Proceedings of the 1st ACM international workshop on complex networks meet information & knowledge management*, Hong Kong, China, 2–6 November 2009, pp. 27–30. New York: ACM.

[3] Ha J, Kim S, Faloutsos C et al. An analysis on information diffusion through blogcast in a blogosphere. *Inform Sciences* 2015; 290(1): 45–62.

[4] Domingos P and Richardson M. Mining the network value of customers. In: *Proceedings of the 7th ACM SIGKDD international conference on knowledge discovery and data mining*, San Francisco, CA, 26–29 August 2001, pp. 57–66. New York: ACM.

[5] Richardson M and Domingos P. Mining knowledge-sharing sites for viral marketing. In: *Proceedings of the 8th ACM SIGKDD international conference on knowledge discovery and data mining*, Edmonton, AB, Canada, 23–26 July 2002, pp. 61–70. New York: ACM.
[6] Kempe D, Kleinberg J and Tardos E. Maximizing the spread of influence through a social network. In: *Proceedings of the 9th ACM SIGKDD international conference on knowledge discovery and data mining*, Washington, DC, 24–27 August 2003, pp. 137–146. New York: ACM.
[7] Chen S, Fan J, Li G et al. Online topic-aware influence maximization. *Proc VLDB Endow* 2015; 8(6): 666–677.
[8] Gong M, Yan J, Shen B et al. Influence maximization in social networks based on discrete particle swarm optimization. *Inform Sciences* 2016; 367: 600–614.
[9] Zhu T, Wang B, Wu B et al. Maximizing the spread of influence ranking in social networks. *Inform Sciences* 2014; 278: 535–544.
[10] Tang Y, Shi Y and Xiao X. Influence maximization in near-linear time: a martingale approach. In: *Proceedings of the ACM SIGMOD international conference on management of data*, Melbourne, VIC, Australia, 31 May–4 June 2015, pp. 1539–1554. New York: ACM.
[11] Li Y, Zhang D and Tan K. Real-time targeted influence maximization for online advertisements. *Proc VLDB Endow* 2015; 8(10): 1070–1081.
[12] Song C, Hsu W and Lee M. Targeted influence maximization in social networks. In: *Proceedings of the 25th ACM international conference on information and knowledge management*, Indianapolis, IN, 24–28 October 2016, pp. 1683–1692. New York: ACM.
[13] Yang Y, Mao X, Pei J et al. Continuous influence maximization: what discounts should we offer to social network users? In: *Proceedings of the international conference on management of data*, San Francisco, CA, 26 June–1 July 2016, pp. 727–741. New York: ACM.
[14] Leskovec J, Krause A, Guestrin C et al. Cost-effective outbreak detection in networks. In: *Proceedings of the 13th ACM SIGKDD international conference on knowledge discovery and data mining*, San Jose, CA, 12–15 August 2007, pp. 420–429. New York: ACM.
[15] Goyal A, Lu W and Lakshmanan L. SIMPATH: an efficient algorithm for influence maximization under the linear threshold model. In: *Proceedings of the 11th IEEE international conference on data mining*, Vancouver, BC, Canada, 11–14 December 2011, pp. 211–220. New York: IEEE.
[16] Kim J, Kim S and Yu H. Scalable and parallelizable processing of influence maximization for large-scale social networks? In: *Proceedings of the 29th IEEE international conference on data engineering*, Brisbane, QLD, Australia, 8–12 April 2013, pp. 266–277. New York: IEEE.
[17] Goyal A, Lu W and Lakshmanan L. CELF ++: optimizing the greedy algorithm for influence maximization in social networks. In: *Proceedings of the 20th international conference on world wide web*, Hyderabad, India, 28 March–1 April 2011, pp. 47–48. New York: ACM.
[18] Wang Y, Cong G, Song G et al. 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*, Washington, DC, 25–28 July 2010, pp. 1039–1048. New York: ACM.
[19] Song G, Zhou X, Wang Y et al. Influence maximization on large-scale mobile social network: a divide-and-conquer method. *IEEE T Parall Distr* 2015; 26: 1379–1392.
[20] Bozorgi A, Haghighi H, Zahedi M et al. INCIM: a community-based algorithm for influence maximization problem under the linear threshold model. *Inform Process Manag* 2016; 52(6): 1188–1199.
[21] Hosseini-Pozveh M, Zamanifar K and Naghsh-Nilch A. A community-based approach to identify the most influential nodes in social networks. *J Inform Sci* 2017; 43(2): 204–220.
[22] Galhotra S, Arora A, Virinchi S et al. ASIM: a scalable algorithm for influence maximization under the independent cascade model. In: *Proceedings of the 24th international conference on world wide web*, Florence, 18–22 May 2015, pp. 35–36. New York: ACM.
[23] Chen W, Yuan Y and Zhang L. Scalable influence maximization in social networks under the linear threshold model. In: *Proceedings of the 10th IEEE international conference on data mining*, Sydney, NSW, Australia, 13–17 December 2010, pp. 88–97. New York: IEEE.
[24] Girvan M and Newman M. Community structure in social and biological networks. *Proc Natl Acad Sci U S A* 2002; 99(12): 7821–7826.
[25] Valiant L. The complexity of enumeration and reliability problems. *SIAM J Comput* 1979; 8(3): 410–421.
[26] Leskovec J, Lang K, Dasgupta A et al. Community structure in large networks: natural cluster sizes and the absence of large well-defined clusters. *Internet Math* 2009; 6(1): 29–123.
[27] Chen W, Wang Y and Yang S. Efficient influence maximization in social networks. In: *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining*, Paris, 28 June–1 July 2009, pp. 199–208. New York: ACM.