

Approximation Methods for Influence Maximization in Temporal Networks



Tsuyoshi Murata and Hokuto Koga

1 Introduction

Diffusion of rumors (or information) can be represented as information propagation in a social network where its nodes are people and its edges are contacts among the people. The scale of information propagation depends on where and when to start the propagation. In order to propagate information as much as possible, starting nodes should be carefully selected. Selecting starting nodes for large-scale information propagation is important as one of the methods for viral marketing.

From given network, selecting such starting nodes for large-scale information propagation was formalized as “influence maximization problem” by Kempe et al. [20]. The original formalization is for static networks. However, nodes and edges can be newly added or deleted in many real social networks. Therefore, influence maximization problem in temporal networks should be considered. Since the influence maximization problem in temporal networks is NP-hard, computing the best solution in realistic time is computationally intractable. Therefore, many approximation schemes based on Monte-Carlo simulation and other heuristic methods have been proposed. Methods based on Monte-Carlo simulation are more accurate but computationally expensive. On the other hand, other heuristic methods are fast but they are less accurate.

In order to find better solutions for the information maximization problem, we propose three new methods for temporal networks as the extension of the methods for static networks. Dynamic Degree Discount is a heuristic method based on node degree. Dynamic CI is a method based on a node’s degree and the degrees of reachable nodes from the node within specific time. Dynamic RIS uses many similar

T. Murata (✉) · H. Koga

Department of Computer Science, School of Computing, Tokyo Institute of Technology, Meguro, Tokyo, Japan

e-mail: murata@c.titech.ac.jp

networks generated by random edge removal. We compare the proposed methods with previous methods. Although the performance of MC greedy was better than the three methods, it was computationally expensive and intractable for large-scale networks. The computational time of our proposed methods was more than 10 times faster than MC greedy. When compared with Osawa, the performances of the three methods were better for most of the cases.

We discuss extended methods for influence maximization in temporal networks [25, 26]. This chapter includes detailed explanation of background knowledge, discussions of the effect of different values of parameters in the proposed methods, and detailed analysis of the advantages and disadvantages of the proposed methods.

The structure of this chapter is as follows. Section 2 shows related work. Section 3 presents proposed methods (Dynamic Degree Discount, Dynamic CI and Dynamic RIS), Sect. 4 explains our experiments, and Sect. 5 shows the experimental results. Section 6 shows discussions about the experimental results, and Sect. 7 concludes the chapter.

2 Related Work

2.1 Model of Information Propagation

We use the SI model as the model of information propagation on networks. In the SI model, each node in networks is either in state S (susceptible) or in state I (infected). Nodes in state S do not know the information and those in state I know the information. At the beginning of information propagation (at time $t = 1$), a set of nodes in state I is fixed as the seed nodes. For all edges (t, u, v) at time $t = 1, 2, \dots, T$, the following operations are performed. If node u is in state I and node v in state S, information is propagated from u to v with probability λ , which means the state of v is changed from S to I at time $t+1$. Probability λ is the parameter of susceptibility, and it controls the percentage of information propagation. At time $t = T + 1$, information propagation is terminated.

Based on the above notations, we can formulate influence maximization problem as follows. We define $\sigma(S)$ as the expected number of nodes of state I at time $T + 1$ when information propagation started at time 1 from seed nodes S of state I based on SI model. (Please keep in mind that S in $\sigma(S)$ is a set of seed nodes, and S in SI model is susceptible state.) Influence maximization problem in a temporal network is to search for a set of seed nodes S of size k that maximizes $\sigma(S)$ when a temporal network G , duration of the network T , susceptibility of SI model λ and the size of seed nodes k are given.

2.2 *Problems Related to Influence Maximization in Temporal Networks*

There are some problems related to influence maximization in temporal networks. Instead of giving item (or information) to seed nodes for free, revenue maximization [2] is the problems of finding seed customers (nodes) and offering discounts to them in order to increase total revenue. Although the problem is important in the field of marketing, it is more complicated than influence maximization problem since seed nodes are not treated as equal, and the amount of discount for each node may not be equal. The number of possible parameters increases greatly especially in the case of temporal networks.

Opinion formation [1, 16, 17] is another problem related to influence maximization problem. Each agent (node) has an opinion which might be a continuous or a discrete quantity. The underlying network represents the society where the agents have interactions. Each agent has an opinion in the society that is influenced by the society. Analyzing the increase and decrease of each opinion is important for modeling the dynamics of opinion formation and for opinion polarization [10].

It is often pointed out that the properties of temporal networks are quite different from those in static networks. Braha and Bar-Yam [4, 5] pointed out the overlap of the centrality in temporal networks and that in the aggregated (static) network is very small. Hill and Braha [13] propose dynamic preferential attachment mechanism that reproduce dynamic centrality phenomena. Holme presents good surveys of temporal networks [14, 15].

2.3 *Influence Maximization Methods for Static Networks*

Jalili presents a survey on spreading dynamics of rumor and disease based on centrality [18]. There are roughly three approaches for influence maximization problem in static networks. The first is Monte-Carlo simulation methods, the second is heuristic-based methods and the third is the methods to generate a large number of networks with random edge removal and select seed nodes based on the generated networks.

Monte-Carlo simulation method is proposed by Kempe et al. [20]. In Kempe's method, $\sigma(S)$ is estimated by repeating Monte-Carlo simulations. When S is given as a set of seed nodes, simulations of information propagation are repeated R times and the average number of infected nodes is defined as $\sigma(S)$. Next, the node v which maximizes the difference $\sigma(S \cup \{v\}) - \sigma(S)$ is added to seed nodes greedily based on the estimated $\sigma(S)$. This operation is repeated until $|S| = k$.

Since $\sigma(\cdot)$ is a monotonic and submodular function, when we denote strict solution of seed nodes as S^* , the seed nodes obtained by the above greedy algorithm S_{greedy} are proved to satisfy $\sigma(S_{greedy}) \geq (1 - 1/e)\sigma(S^*)$ [20]. Because of this property, qualities of the solutions by Kempe's method are good. However, more and more repetition of Monte-Carlo simulation is needed in order to estimate $\sigma(S)$ accurately. Since the computational cost for finding seed nodes with this method is high, it is not possible to find seed nodes in realistic time for large scale networks.

Heuristic methods are proposed in order to search for seed nodes at high speed. Chen et al.[7] proposes PMIA to find seed nodes focusing on the paths with high information propagation ratio. Jiang et al. [19] proposed SAEDV which searches for seed nodes by annealing method to obtain $\sigma(\cdot)$ from adjacent nodes in seed nodes. Chen et al. [8] proposed Degree Discount based on node degree where the nodes adjacent to already selected node are given penalty. This is because when node v is selected as one of seed nodes and u is its neighbor, it is highly likely that v propagates information to u , so selecting nodes other than u as seed nodes is better for information diffusion.

Algorithm of Degree Discount is shown as follows. t_i in the algorithm shows the penalty of node i . dd_i is the degree of node i after giving penalty. dd_i is smaller when the value of t_i is bigger.

Algorithm 1 Degree discount

Inputs

Static network G
 The size of seed nodes k
 Susceptibility λ

Outputs

Seed nodes S

Algorithm

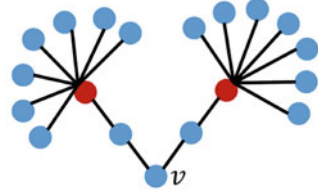
- (1) Initialize seed nodes as $S = \phi$, and initialize $dd_i = k_i$, $t_i = 0$ for each node i . k_i is the degree of node i in network G .
- (2) Add node v to seed nodes S such that $v = \operatorname{argmax}_i \{dd_i | i \in V \setminus S\}$. V is the nodes in network G .
- (3) Update dd_u and t_u for all nodes u adjacent to v .

$$t_u = t_u + 1$$

$$dd_u = k_u - 2t_u - (k_u - t_u)t_u\lambda$$

- (4) Repeat (2) and (3) until $|S| = k$.
-

Fig. 1 Example for explaining $CI_l(v)$. When $l = 2$, $CI_l(v) = 14$



Morone and Makse [24] proposed a method for finding seed nodes considering the degrees of distant nodes. The method calculates the following $CI_l(v)$ for each node and selects seed nodes based on the values:

$$CI_l(v) = (k_v - 1) \sum_{u \in \partial Ball(v,l)} (k_u - 1).$$

$\partial Ball(v, l)$ in the above formula represents nodes where the distance from node v is l . The example of $CI_l(v)$ is explained in Fig. 1. $\partial Ball(v, 2)$ when $l = 2$ are two nodes with distance 2 from node v and the degrees of both nodes are 8. Therefore, $CI_2(v) = (2 - 1) \times \{(8 - 1) + (8 - 1)\} = 14$.

The degree of node v itself is low in the network in Fig. 1, but the node v is effective for information propagation because it is connected with some high degree nodes with distance two. This method thus selects seed nodes causing wider propagation compared with the cases when seed nodes are selected based on the degree of node v only.

These heuristic methods compute seed nodes faster than the methods based on Monte-Carlo simulation. However, it is experimentally confirmed that the scale of propagation of the methods depends on network structures and parameters.

Ohsaka et al. [27] proposed a method to generate many networks with random edge removal in order to solve this problem. Ohsaka’s method is based on “coin flip” mentioned in Kempe’s paper [20]. The distribution of nodes to which information is propagated from seed nodes S in static network G is set as $D_G(S)$. And distribution of nodes where information is propagated from seed nodes S on network where edges are removed at constant ratio from the network G is set as $D'_G(S)$. “Coin flip” states as $D_G(S)$ equals to $D'_G(S)$ in this situation. $\sigma(\cdot)$ can be estimated by generating many networks with edges removed at constant ratio, not by repeating Monte-Carlo simulation. Ohsaka’s method estimates $\sigma(\cdot)$ by acquiring Strongly Connected Component (SCC) in each network generated by RR numbers of networks with edges removed at constant ratio. SCC is a subgraph where each node in the subgraph can be reachable to and from any other nodes.

Borgs et al. [3] and Tang et al. [30] also propose methods similar to Ohsaka’s method. The difference from Ohsaka’s method is $\sigma(\cdot)$, which is not estimated directly from generated networks. Reachable nodes from randomly selected node v are computed, and then seed nodes are selected based on the nodes. More specifically, the algorithm is as follows.

Algorithm 2 Algorithm by Borgs and Tang

Input

Static network G
 The size of seed nodes k
 Susceptibility λ
 Generated number of networks θ

Outputs

Seed nodes S

Algorithm

- (1) Initialize $S = \phi, U = \phi$. U is a set of all RR .
 - (2) Select node v at random.
 - (3) Remove edges with probability $1 - \lambda$ from network G and set as G_p .
 - (4) Acquire nodes RR reachable to v by G_p . Add RR to U .
 - (5) Repeat θ times from (2) to (4).
 - (6) Add node u with the highest frequency in U to S .
 - (7) Delete all RR containing u from U .
 - (8) Repeat (6) and (7) until $|S| = k$.
-

There are other approaches for influence maximization problem in different problem setting. Chen et al. [6] proposed a method to solve the problem with time limit. Feng et al. [9] solves the influence maximization problem in a situation where freshness of the information degrades as it spreads. Mihara et al. [23] proposed a method to influence maximization problem where the whole network structure is unknown.

2.4 Degrees in Temporal Networks

Notations of edges and paths in temporal networks are the same as the ones in [28]. (t, u, v) represents an edge from node u to v at time t . A path from node v_1 to v_k of length $k - 1$ is represented as $(t_1, v_1, v_2), (t_2, v_2, v_3), \dots, (t_{k-1}, v_{k-1}, v_k)$, where $t_1 < t_2 < \dots < t_{k-1}$ and $\forall i, j (i \neq j), v_i \neq v_j$. Duration of time from the start to the end of a path $t_{k-1} - t_1$ is the length of time of the path, and the smallest one is the minimum length of time.

Habiba et al. [12] define degrees in temporal network using symmetric difference of past connections and future connections. However, diffusion in temporal networks is from past to future only, and it is not bidirectional. We therefore define degree $D_T(v)$ of node v in temporal network as follows:

$$D_T(v) = \sum_{1 < t \leq T} \frac{|N(v, t-1) \setminus N(v, t)|}{|N(v, t-1) \cup N(v, t)|} |N(v, t)|,$$

where $N(v, t)$ is a collection of nodes adjacent to node v at time t . Figures 2 and 3 illustrate the examples of degrees on temporal networks. In Fig. 2, adjacent

Fig. 2 Example of low degree nodes in a temporal network. Nodes adjacent to node A do not change over time ($\{B, C\} \rightarrow \{B, C\} \rightarrow \{B, C\}$)

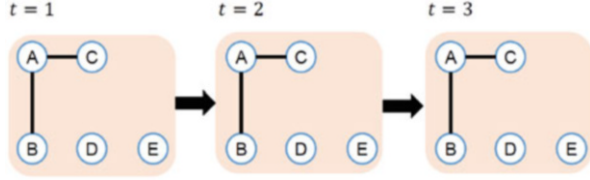
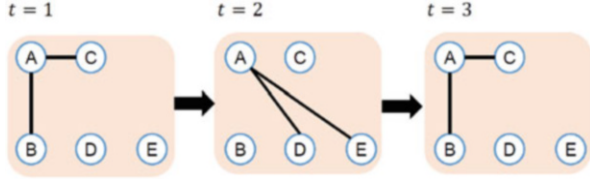


Fig. 3 Example of high degree nodes in a temporal network. Nodes adjacent to node A change at each time ($\{B, C\} \rightarrow \{D, E\} \rightarrow \{B, C\}$)



nodes of node A do not change during the period. The difference of adjacent nodes $N(A, 1) \setminus N(A, 2)$ and $N(A, 2) \setminus N(A, 3)$ are empty. Therefore, the degree of node A in Fig. 2 is calculated as follows.

$$\begin{aligned}
 D_3(A) &= \frac{|N(A, 1) \setminus N(A, 2)|}{|N(A, 1) \cup N(A, 2)|} |N(A, 2)| + \frac{|N(A, 2) \setminus N(A, 3)|}{|N(A, 2) \cup N(A, 3)|} |N(A, 3)| \\
 &= \frac{0}{2} * 2 + \frac{0}{2} * 2 = 0
 \end{aligned}$$

On the other hand, in Fig. 3, nodes adjacent to node A change over time. So the degree of node A is bigger than that in Fig. 2.

$$\begin{aligned}
 D_3(A) &= \frac{|N(A, 1) \setminus N(A, 2)|}{|N(A, 1) \cup N(A, 2)|} |N(A, 2)| + \frac{|N(A, 2) \setminus N(A, 3)|}{|N(A, 2) \cup N(A, 3)|} |N(A, 3)| \\
 &= \frac{2}{4} * 2 + \frac{2}{4} * 2 = 2
 \end{aligned}$$

In Figs. 2 and 3, the number of adjacent nodes of node A is the same every time, so the average degree of node A is the same in Figs. 2 and 3. On the other hand, if we employ $D_T(v)$ as the definition of node degree, $D_3(A) = 0$ in Fig. 2 and $D_3(A) = 2$ in Fig. 3. $D_T(v)$ captures the number of newly adjacent nodes, and this is important for influence maximization problem. We therefore employ $D_T(v)$ as the definition of node degree in temporal networks.

2.5 Influence Maximization Methods for Temporal Networks

There are two approaches for influence maximization problem in temporal networks: methods based on Monte-Carlo simulation and heuristic-based methods. The

former method is proposed by Habiba. The method estimates the scale of propagation $\sigma(\cdot)$ by repeating Monte-Carlo simulation just the same as in static networks. Since $\sigma(\cdot)$ is monotonic and deteriorated modular also in temporal networks, this method achieves large-scale propagation. However, the computational cost of this method is high as in static networks. Osawa [28] proposed a heuristic method for calculating $\sigma(\cdot)$ at high speed. His algorithm for computing $\sigma(S)$ for seed nodes S is shown as follows.

Algorithm 3 Osawa's algorithm

Input

Temporal network G
 Duration of temporal network T
 Seed nodes S
 Susceptibility λ

Output

The number of state I nodes $\sigma(S)$

Algorithm

- (1) Initialize $\hat{p}_v(1)$ as follows where the probability of node v being in state I is $\hat{p}_v(t)$ in time t ($1 \leq t \leq T + 1$).

$$\hat{p}_v(1) = \begin{cases} 1 & v \in S \\ 0 & v \notin S \end{cases}$$

- (2) Update $\hat{p}_v(t)$ in time t as follows.

$$\begin{aligned} \hat{p}_v(t+1) &= 1 - (1 - \hat{p}_v(t))R_v(t) \\ R_v(t) &= \prod_{u \in N(v,t)} (1 - \hat{p}_u(t)\lambda) \end{aligned}$$

where $R_v(t)$ is the probability of which information is not propagated to node v from any adjacent nodes at time t . $N(v, t)$ is the set of nodes adjacent to node v at time t .

- (3) $\sigma(S)$ is computed by adding probability $\hat{p}_v(T + 1)$ of each node which is in state I at time $T + 1$.

$$\sigma(S) = \sum_{v \in V} \hat{p}_v(T + 1)$$

After $\sigma(S)$ is computed, seed nodes are obtained by greedy algorithm as in the method by Monte-Carlo simulation. Osawa's method finds seed nodes in realistic computational time. However, the quality of its solution depends on given networks because $\sigma(\cdot)$ is calculated approximately, and it is worse compared with the solutions by Monte-Carlo simulation.

3 Proposed Methods

We propose new methods for influence maximization problem in temporal networks in this section. We propose three new methods (Dynamic Degree Discount, Dynamic CI and Dynamic RIS) which are the extensions of static network methods to temporal network methods. We use the following notations: G : temporal network, T : duration of the temporal network, k : the size of seed nodes, λ : susceptibility, θ : the number of generated networks, and S : seed nodes.

3.1 *Dynamic Degree Discount*

Dynamic Degree Discount is the extension of Degree Discount by Chen et al. [8] to temporal networks. In Dynamic Degree Discount, definition of degrees and adjacent nodes in the algorithm of Degree Discount are modified for temporal networks. Algorithm 4 shows the algorithm of Dynamic Degree Discount. Underlines show the parts modified from original Degree Discount.

3.2 *Dynamic CI*

Dynamic CI is an extension of Morone's method [24] for temporal networks. Morone's method focuses on the degree of node v and the degrees of nodes with distance l from v . Dynamic CI defines an index $D_CI_l(v)$ in which degree and distance are extended to temporal networks.

$$D_CI_l(v) = D_T(v) \sum_{u \in DBall(v,l)} D_T(u)$$

The differences between $CI_l(v)$ and $D_CI_l(v)$ are: (1) the definition of degree is changed to that for temporal networks and (2) $\partial Ball(v, l)$ in $CI_l(v)$ is changed to $DBall(v, l)$. $DBall(v, l)$ represents nodes where their shortest duration of time from node v is l . l is a parameter which takes the value within the range $1 \leq l \leq T$. In the algorithm of Dynamic CI, $D_CI_l(v)$ is computed for each node and top k nodes are selected as seed nodes.

Algorithm 4 Dynamic degree discount

Input

Temporal network G
 Duration of temporal network T
 The size of seed nodes k
 Susceptibility λ

Output

Seed nodes S

Algorithm

- (1) Initialize seed nodes as $S = \phi$. Also initialize the values of each node i as $dd_i = D_T(i)$ and $t_i = 0$.
- (2) Add node v where $v = \operatorname{argmax}_i \{dd_i | i \in V \setminus S\}$ to S . V is the set of nodes in the network.
- (3) For all nodes u where $u \in N_T(v)$, update dd_u and t_u as follows.

$$t_u = t_u + 1$$

$$dd_u = \frac{D_T(u) - 2t_u - (D_T(u) - t_u)t_u\lambda}{D_T(u)}$$

$N_T(v)$ represents a set of all nodes adjacent to v during the whole period of the temporal network.

$$N_T(v) = \bigcup_{t=1}^T N(v, t)$$

- (4) Repeat (2) and (3) until $|S| = k$.
-

3.3 Dynamic RIS

Dynamic RIS is an extension of Borgs's method [3] and Tang's method [30] for temporal networks.

The difference between Borgs's and Tang's algorithms and Dynamic RIS are where RR in their algorithm is set as $RR(v, d)$ in our algorithm. $RR(v, d)$ is a set of all nodes that are reachable to v within the shortest duration of time d in all durations of temporal networks, which is defined as follows:

$$RR(v, d) = \bigcup_{t=1}^T RR_t(v, d).$$

$RR_t(v, d)$ is a set of nodes which are reachable to "node v at time t " within the shortest period of d . The computational complexities of these methods are as follows.

Algorithm 5 Dynamic RIS

Inputs

- Temporal network G
- Duration of the temporal network T
- Size of seed nodes k
- Susceptibility λ
- The number of generated networks θ

Output

- Seed nodes S

Algorithm

- (1) Initialize as $S = \phi, U = \phi$, where U is the set containing all $RR(v, d)$.
 - (2) Select node v at random.
 - (3) Remove edges from temporal network G with probability $1 - \lambda$ and set as G_p .
 - (4) Acquire nodes $RR(v, d)$ reachable to v on G_p . Add $RR(v, d)$ to U .
 - (5) Repeat (2) to (4) θ times.
 - (6) Add the most frequent node u in U to S .
 - (7) Remove all $RR(v, d)$ with u from U .
 - (8) Repeat (6) and (7) until $|S| = k$.
-

Dynamic Degree Discount

According to the paper of Chen et al. [8], the computational complexity of Degree Discount is $O(k \cdot \log n + m)$, where k is the number of seed nodes, n is the number of nodes, and m is the number of edges, respectively. Dynamic Degree Discount is an extension of Degree Discount. Static degree is replaced with dynamic one ($D_T(i)$) and Static neighbors is replaced with dynamic one ($N_T(v)$). Computational complexity for dynamic degree and dynamic neighbors are $\frac{T \cdot m}{n}$, where T is the total duration of time of given temporal network. Therefore, the total computational complexity of Dynamic Degree Discount is $O(k \cdot \log n + m + \frac{T \cdot m}{n})$.

Dynamic CI

According to the paper of Morone and Makse [24], the computational complexity of CI is $O(n \cdot \log n)$, where n is the number of nodes. Dynamic CI is an extension of CI. Static degree is replaced with dynamic one ($D_T(i)$), and its computational complexity is $\frac{T \cdot m}{n}$, where T is the total duration of time of given temporal network. Therefore, the total computational complexity of Dynamic CI is $O(n \cdot \log n + \frac{T \cdot m}{n})$.

Dynamic RIS

According to the paper of Tang [30], the computational complexity of RIS is $O(k \cdot l^2(m+n) \log^2 n / \epsilon^3)$ which returns $(1 - \frac{1}{e} - \epsilon)$ -approximate solution with at least $1 - n^{-l}$ probability, where l and ϵ are the constants. Computational complexity of Dynamic RIS heavily depends on the parameters θ and d , which are the number of generated networks and the duration of time for computing $RR(v, d)$, respectively. Therefore, the total computational complexity of Dynamic RIS is $O(\theta \cdot d \cdot k \cdot l^2(m+n) \log^2 n / \epsilon^3)$.

4 Experiments

We perform experiments for comparing the proposed methods with previous ones in order to confirm their effectiveness. Temporal networks used for the experiments are shown in Table 1. These networks are the same as the ones used in previous research. Average degree in Table 1 shows the average of all nodes in the network, which is $\frac{1}{|V|} \sum_{v \in V} D_T(v)$. Hospital [31] is a network about contacts of patients and medical staffs at hospital with time. Primary School [11, 29] is a network about contacts of students and teachers at school. High School 2013 [22] is a network of contacts of students. The unit of the duration in these three datasets is 20 s. Each dataset is available at SocioPatterns (<http://www.sociopatterns.org>).

Methods used in the experiments are previous two methods (Monte-Carlo simulation (MC Greedy) and Osawa) for temporal network explained in Sect. 2.5 and our proposed methods (Dynamic Degree Discount, Dynamic CI and Dynamic RIS) in Sect. 3. Given a network as input, each method computes seed nodes S . The simulation of influence maximization based on SI model is repeated R times with the obtained seed nodes and set the average of the number of nodes in state I as $\sigma(S)$. The values of $\sigma(S)$ are compared in order to evaluate the methods.

Experiments are performed for the following purposes:

- (1) Comparison of $\sigma(S)$ when the size of seed nodes k changes
- (2) Comparison of computational time when the size of seed nodes k changes

Parameters in the experiments are set as follows. The number of repetition of the simulations for information propagation is set as $R = 50$. The number of repetition of Monte-Carlo simulation in MC Greedy is set as 1000. These two parameters are common in all experiments. The size of seed nodes k is set from 0 to 20%. Susceptibility λ is set as $\lambda = 0.01$. It is difficult to perform experiments for all the values as parameter l in Dynamic CI which takes the value of $1 \leq l \leq T$. We use the values $l = 1, 5, 10, 20$ in the experiments. As the parameters θ and d in Dynamic RIS, θ is set as $\theta = 1000$. As for d , values $d = 0, 5, 10, 20$ are used since it is difficult to perform experiments for all the value as in l of Dynamic CI.

CELFF [21] is used to speedup the experiments when greedy algorithms are used in MC Greedy and Osawa. CELFF is an algorithm used when the greedy algorithm is applied to the problem with inferior modularity, and the solution is the same as in normal greedy algorithm. According to the experiments by Leskovec et al. [21], computational time is 700 times faster than normal greedy algorithm when CELFF is used.

Table 1 Dataset for the experiments

	Nodes	Edges	Duration	Ave. deg.
Hospital	75	32,424	9,453	69.3
Primary school	242	125,773	3,100	142.7
High school 2013	327	188,508	7,375	63.0

5 Experimental Results

5.1 Comparison of $\sigma(S)$ When the Size of Seed Nodes k Changes

The results of information propagation for each size of seed nodes k with fixed susceptible $\lambda = 0.01$ of SI model are shown in Fig. 4. The x-axis of the Figure shows the percentage of seed nodes, and the y-axis shows the number of infected nodes. Values of the x-axis is $\frac{k}{|V|} * 100$, the percentage of seed nodes to all nodes in the network. Values of the y-axis is $\frac{\sigma(S)}{|V|} * 100$, the percentage of $\sigma(S)$ to all nodes in the network. The best values of l in Dynamic CI and d in Dynamic RIS are used in our experiments. As shown in Fig. 4, MC Greedy achieves the highest diffusion in all dataset. Diffusion of the proposed methods, Dynamic Degree Discount, Dynamic CI and Dynamic RIS are inferior to MC Greedy, but they are still better than Osawa. The scale of diffusion of Dynamic RIS in High School 2013 achieves 1.5 times as in Osawa.

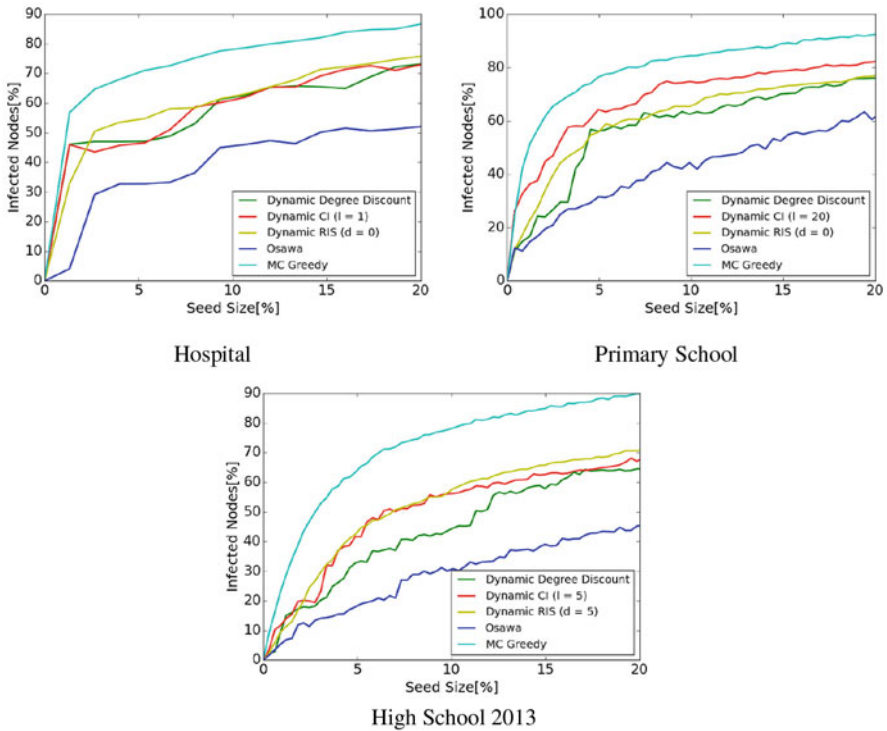


Fig. 4 Comparison of $\sigma(S)$ when the size of seed nodes k changes. Except (computationally expensive) MC Greedy, three proposed methods are better than Osawa

There is not much difference in the scale of diffusion among each of three proposed methods. Dynamic RIS achieves the highest in High School 2013 for example, but the difference among proposed methods is small compared with the difference between proposed methods and previous methods (MC Greedy and Osawa).

5.2 Comparison of $\sigma(S)$ When Susceptibility λ Changes

Figure 5 shows diffusion when the size of seed nodes is fixed as 20% of all nodes in the networks and susceptibility is changed as $\lambda = 0.001, 0.01, 0.05$. The x-axis shows the value of λ , and the y-axis shows the percentage of diffusion. Parameters l and d are the same as the ones used in the previous experiments. As shown in Fig. 5, MC Greedy achieves the highest diffusion regardless of the value of λ . The difference among three proposed methods are small.

As the result of comparison with proposed methods and Osawa, our proposed methods achieve higher scale of diffusion than Osawa in Hospital and High School 2013 when $\lambda = 0.05$. Osawa achieves higher diffusion than Dynamic RIS only in

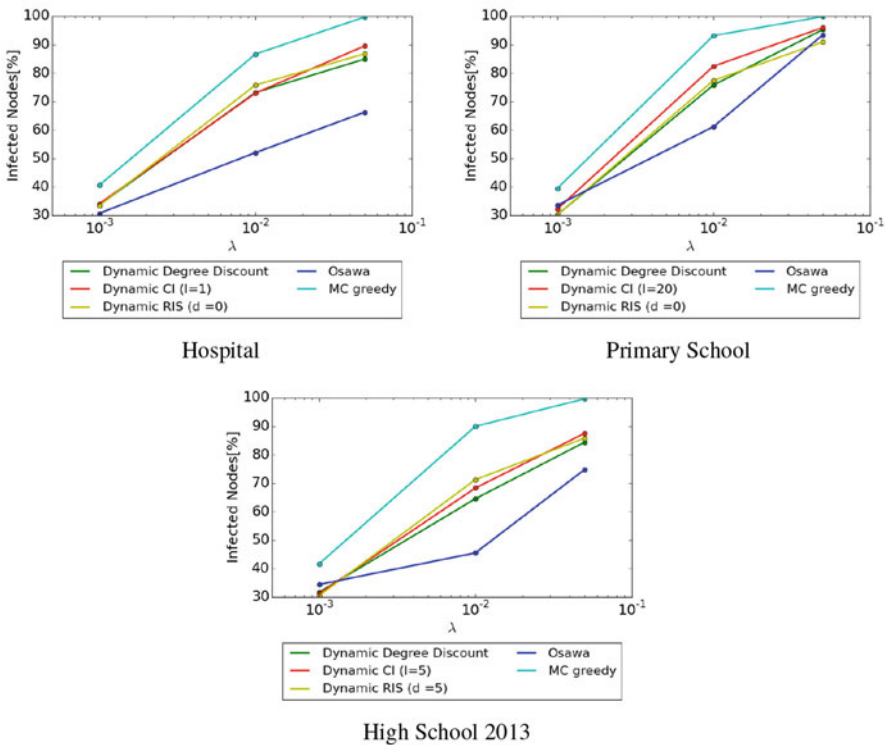


Fig. 5 Comparison of $\sigma(S)$ when susceptibility λ changes. Except (computationally expensive) MC Greedy, three proposed methods are better than Osawa for most of the cases

Primary School. When $\lambda = 0.001$, the difference between proposed methods and Osawa is very small compared with the cases of other λ values.

5.3 Comparison of Computational Time When the Size of Seed Nodes k Changes

Figure 6 shows the computational time when λ is set as $\lambda = 0.01$ and the sizes of seed nodes are changed. A PC of Intel Core i7 (3.4GHz) CPU and 8GB memory is used for the experiments. X-axis shows the percentage of seed nodes, and y-axis shows the computational time (log-scale).

Figure 6 shows that for all datasets, methods other than MC Greedy can compute seed nodes in realistic time. MC Greedy needs several hours to compute seed nodes. This shows that MC Greedy is intractable in realistic time for large scale networks.

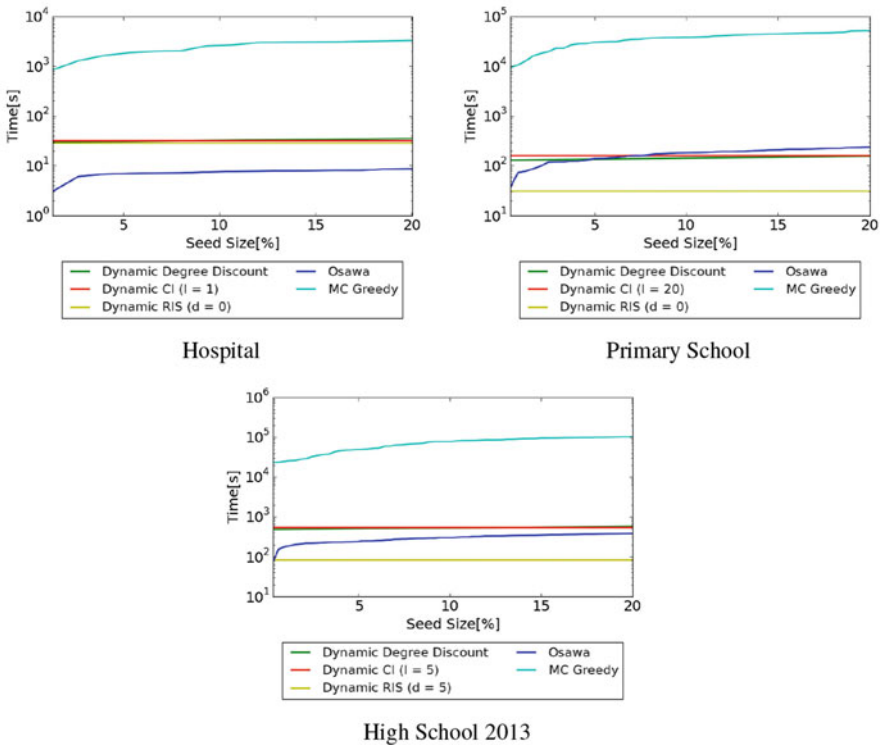


Fig. 6 Comparison of computational time when the size of seed nodes k changes. Methods other than MC Greedy can compute in realistic time

Regarding the comparison among three proposed algorithm, computational time of Dynamic Degree Discount and Dynamic CI are almost the same in all dataset. Dynamic RIS is about the same computational time as the other two proposed methods in Hospital, and is faster in Primary School and High School 2013. Regarding the comparison with proposed methods and Osawa, Dynamic RIS is approximately 7.8 times faster than Osawa except very small network (Hospital).

5.4 Parameters of Dynamic CI and Dynamic RIS

Diffusion of proposed methods with different parameters are shown in this section. We change parameters l of Dynamic CI, and θ and d in Dynamic RIS.

5.4.1 Diffusion and Computational Time of Different l in Dynamic CI

Diffusion and computational time when l in Dynamic CI changes to 1, 5, 10, 20 are shown in Fig. 7. Left line graphs show the size of diffusion when l is changed in each network. Right bar graphs show computational time. Left line graphs show that diffusion depends on the value of l . Therefore, it is important to find appropriate l in Dynamic CI. Since there is no simple correlation between the scale of diffusion and the value of l (such as diffusion becomes larger as l becomes large), diffusion for various values of l should be investigated and compared. Right bar graphs show that there are no big differences of execution time when the value of l changes.

5.4.2 Diffusion and Computational Time of Different θ in Dynamic RIS

In Dynamic RIS, θ is a parameter for the number of generated graphs in $RR(v, d)$. Diffusion and computational time when parameter θ is changed to 500, 1000, 1500, 2000 are shown in Fig. 8. Left line graphs show that diffusion does not change much when θ changes. However, the scale of diffusion is slightly small when $\theta = 500$ in Hospital and High School 2013. This means that bigger θ is desirable from the viewpoint of diffusion. On the contrary, right bar graphs show that higher value of θ results in the increase of computational time. From the viewpoint of computational time, smaller θ is better. Regarding the value of θ , there is a trade-off between the scale of diffusion and the computational time. It is important to find smaller θ for shorter computational time, but too small θ results in small-scale diffusion.

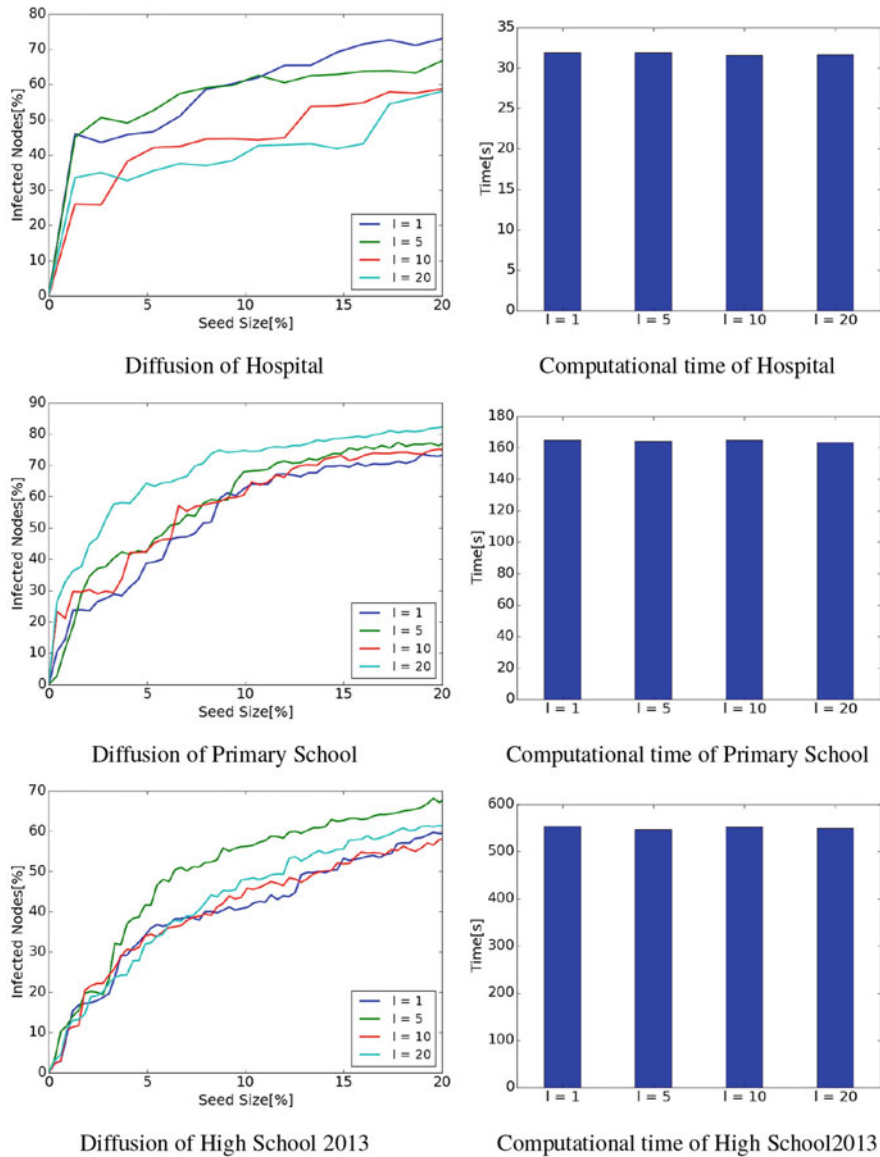
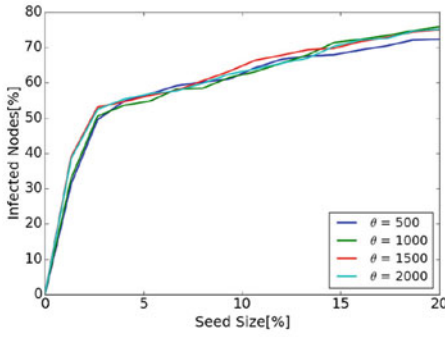
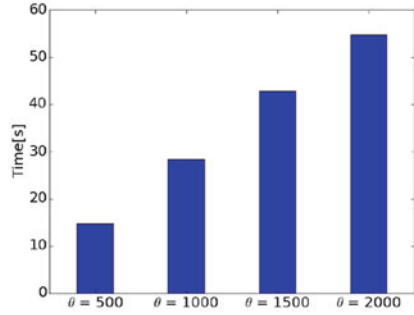


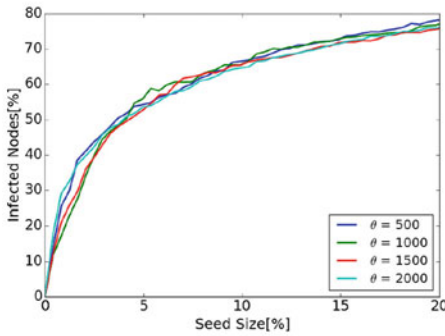
Fig. 7 Diffusion and computational time for different l in Dynamic CI. Left: there is no simple correlation between the scale of diffusion and the value of l . Right: there are no big differences of execution time when the value of l changes



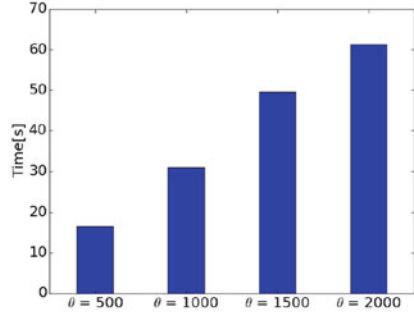
Diffusion of Hospital



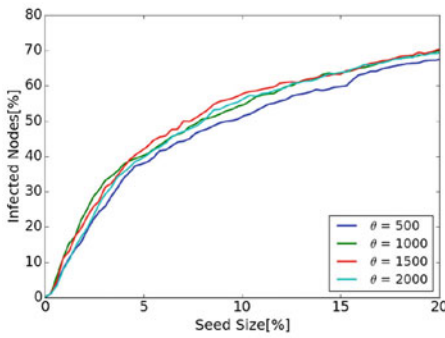
Computational time of Hospital



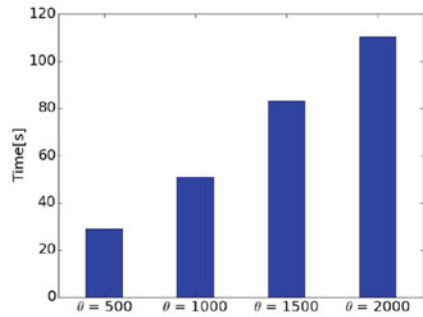
Diffusion of Primary School



Computational time of Primary School



Diffusion of High School 2013



Computational time of High School 2013

Fig. 8 Diffusion and computational time of different θ in Dynamic RIS. Left: diffusion does not change much when θ changes. Right: higher value of θ results in the increase of computational time

5.4.3 Diffusion and Computational Time of Different d in Dynamic RIS

In Dynamic RIS, d is a parameter for the number of time steps for looking back. Figure 9 shows diffusion and executing time when parameter d changes to

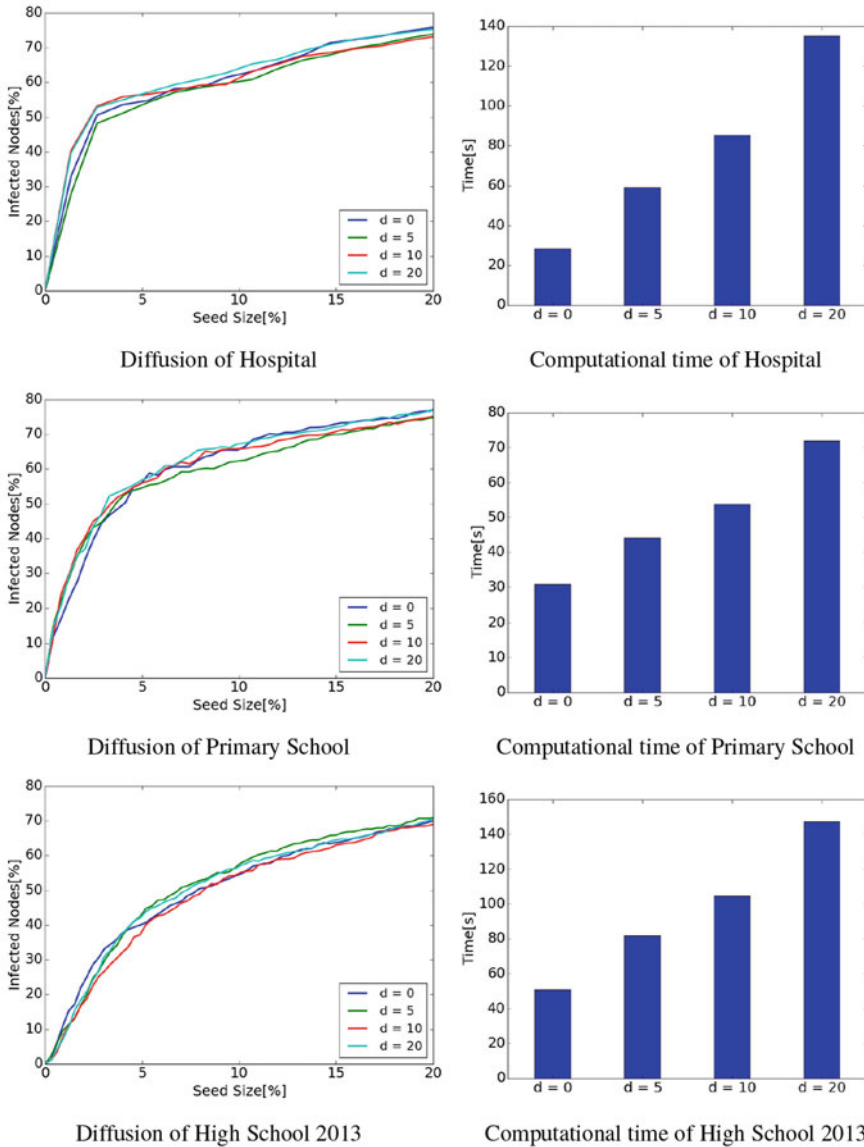


Fig. 9 Diffusion and computational time of different d in Dynamic RIS. Left: there is almost no difference in diffusion when d changes. Right: computational time increases as the value of d becomes bigger

0, 5, 10, 20. Left line graphs show that there is almost no difference in diffusion when d changes, while right bar graphs show that computational time increases as the value of d becomes bigger. The scale of diffusion does not change even if the value of d becomes bigger in our experiments.

6 Discussion

6.1 Analysis Focused on Diffusion of Each Node

In the experiments when susceptibility changes in Sect. 5.2, the difference between the proposed methods and Osawa was small when $\lambda = 0.001$ compared with the experiments with other values of λ . When $\lambda = 0.05$, Osawa outperforms proposed methods only in Primary School. This section discusses these two points.

Figure 10 shows the distribution of diffusion $\sigma(\{v\})$ of each node v when Monte-Carlo simulation is used. X-axis shows the percentage of diffusion from node v to the whole network ($\sigma(\{v\})$), and Y-axis shows the frequency of the nodes with each of the percentage in X-axis. When $\lambda = 0.001$, almost all nodes are less than 5% of diffusion in all networks. This means that there is no big difference of the diffusion from different seed nodes. This is the reason why the difference between proposed methods and Osawa is small in the experiment in Sect. 5.2. On the contrary, there are many nodes with more than 60% of diffusion in Primary School when $\lambda = 0.05$ compared with other two networks. In this case, large scale diffusion is easy to be achieved even if the most appropriate seed nodes are not selected. This is the reason why Osawa outperforms proposed method in Primary School in Sect. 5.2.

6.2 Advantages and Disadvantages of Each of Proposed Methods

Advantages and disadvantages of each of proposed methods are discussed in this section. An advantage of Dynamic Degree Discount is that it contains no parameter, so there is no need to adjust parameter. Its disadvantage is that it is only for SI model, so the method cannot be used for other models. This is because Dynamic Degree Discount is an extension of Chen's Degree Discount which is for SI model. There are other information propagation models such as LT model and Triggering models proposed by Kempe et al. Dynamic Degree Discount cannot be applied to such models.

An advantage of Dynamic CI is that it can be applied to many information propagation models in contrast to Dynamic Degree Discount because Dynamic CI uses only degree information when it calculates seed nodes. Its disadvantage is

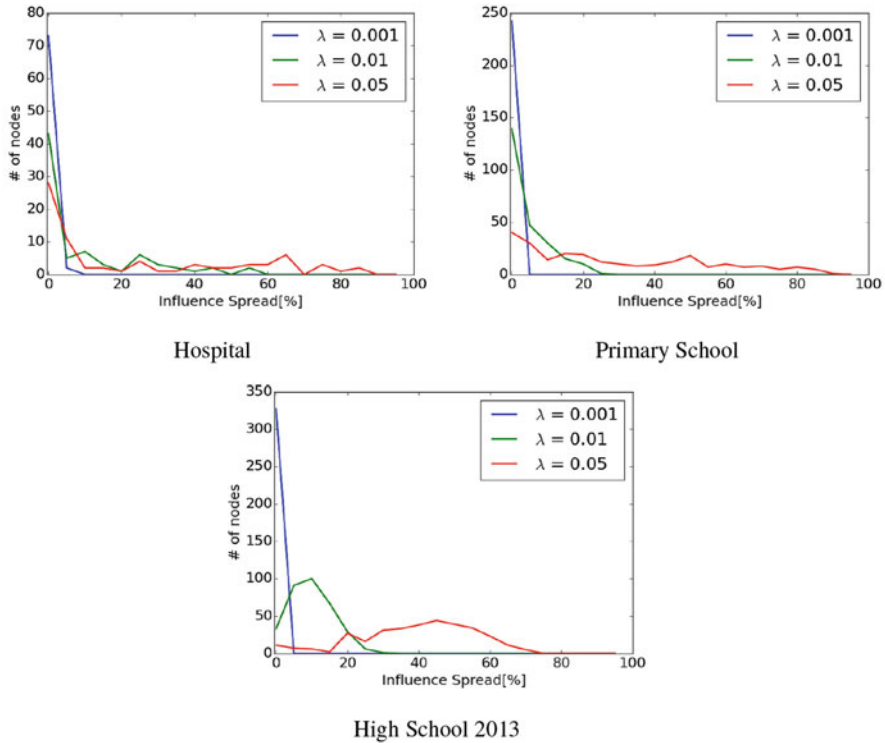


Fig. 10 Distribution of diffusion $\sigma(v)$ of each node v . In the case of Primary School when $\lambda = 0.05$, there are many nodes of high diffusion. Therefore, large scale diffusion is easy to be achieved even if the most appropriate seed nodes are not selected. This is the reason Osawa outperforms proposed method in Primary School

that the ability of diffusion depends on the value of parameter l as mentioned in section 5.4.1. It is necessary to search for appropriate values of l for Dynamic CI. The parameter l takes the value within the range $1 < l < T$, so the search takes time in general.

An advantage of Dynamic RIS is that its computational time is short. As shown in the experimental results, its computational time is shorter than other methods in all networks except Hospital. As the method can be applied to large networks due to its short computational time, this is a big advantage. Disadvantage of Dynamic RIS is that it needs to adjust parameters θ and d . As mentioned in the previous section, computational time becomes bigger as the parameter θ becomes bigger, and the scale of diffusion becomes smaller for too small θ . Therefore, it is necessary to set appropriate value for θ . However, parameter sensitivity of θ and d is not so much compared with the sensitivity of l in Dynamic CI.

7 Conclusion

We propose three new methods for influence maximization problem in temporal networks which are the extensions of the methods for static networks. As the result of experiments for comparing with previous methods, MC Greedy and Osawa, our three proposed methods are better than previous methods in the following sense. Although the performance of MC greedy is better than these three methods, it is computationally expensive and intractable for large scale networks. The computational time of our proposed methods are more than 10 times faster than MC greedy, so they can be computed in realistic time even for large scale temporal networks. As the comparison with Osawa, the performances of these three methods are almost the same as Osawa, but they are approximately 7.8 times faster than Osawa. Based on these facts, the proposed methods are suitable for influence maximization in temporal networks.

The comparison of Dynamic Degree Discount, Dynamic CI and Dynamic RIS is as follows. The choice of the methods should be done based on the following pros and cons.

Dynamic Degree Discount

- It requires no parameter.
- It is applicable to SI model only.

Dynamic CI

- It is applicable to other information propagation models.
- The performance heavily depend on parameter l .

Dynamic RIS

- It is relatively fast among these three methods.
- It requires two parameters to be adjusted (θ and d).

Finding the strategies of choosing suitable method for given temporal network is practically important. It is a challenging open question and is left for our future work. The problem of adjusting the parameters for Dynamic CI and and Dynamic RIS is also left for our future work.

Acknowledgement This work was supported by JSPS Grant-in-Aid for Scientific Research(B) (Grant Number 17H01785).

References

1. Afshar, M., Asadpour, M.: Opinion formation by informed agents. *J. Artif. Soc. Soc. Simul.* **13**(4), 1–5 (2010)
2. Babaei, M., Mirzasoleiman, B., Jalili, M., Safari, M.A.: Revenue maximization in social networks through discounting. *Soc. Netw. Anal. Min.* **3**(4), 1249–1262 (2013)

3. Borgs, C., Brautbar, M., Chayes, J., Lucier, B.: Maximizing social influence in nearly optimal time. In: Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 946–957. (2014)
4. Braha, D., Bar-Yam, Y.: From centrality to temporary fame: Dynamic centrality in complex networks. *Complexity* **12**(2), 59–63 (2006)
5. Braha, D., Bar-Yam, Y.: Time-dependent complex networks: Dynamic centrality, dynamic motifs, and cycles of social interactions. In: Adaptive Networks: Theory, Models and Applications, pp. 39–50. Springer, Berlin (2009)
6. Chen, W., Lu, W., Zhang, N.: Time-Critical Influence Maximization in Social Networks with Time-Delayed Diffusion Process. In: Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, pp. 592–598 (2012)
7. Chen, W., Wang, C., Wang, Y.: Scalable influence maximization for prevalent viral marketing in large-scale social networks. In: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '10, pp. 1029–1038 (2010)
8. Chen, W., Wang, Y., Yang, S.: Efficient influence maximization in social networks. In: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '09, pp. 199–207 (2009)
9. Feng, S., Chen, X., Cong, G., Yifeng, Z., Yeow, Meng, C., Yanping, X.: Influence Maximization with Novelty Decay in Social Networks. In: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, pp. 37–43 (2014)
10. Garimella, K., Morales, G.D.F., Mathioudakis, M., Gionis, A.: Polarization on social media. *Web Conf 2018 Tutorial* **1**(1), 1–191 (2018)
11. Gemmetto, V., Barrat, A., Cattuto, C.: Mitigation of infectious disease at school: targeted class closure vs school closure. *BMC Infect. Dis.* **14**(1), 1 (2014)
12. Habiba, Yu, Y., Berger-Wolf, T.Y., Saia, J.: Finding spread blockers in dynamic networks. In: Advances in Social Network Mining and Analysis, vol. 5498, pp. 55–76 (2010)
13. Hill, S.A., Braha, D.: Dynamic model of time-dependent complex networks. *Phys. Rev. E* **82**(046105), 1–7 (2010)
14. Holme, P.: Modern temporal network theory: a colloquium. *Eur. Phys. J. B* **88**(234), 1–30 (2015)
15. Holme, P., Saramäki, J.: Temporal networks. *Phys. Rep.* **519**(3), 97–125 (2012)
16. Jalili, M.: Effects of leaders and social power on opinion formation in complex networks. *Simulation* **89**(5), 578–588 (2012)
17. Jalili, M.: Social power and opinion formation in complex networks. *Physica A* **392**(4), 959–966 (2013)
18. Jalili, M., Perc, M.: Information cascades in complex networks. *J. Complex Networks* **5**(5), 665–693 (2017)
19. Jiang, Q., Song, G., Cong, G., Wang, Y., Si, W., Xie, K.: Simulated Annealing Based Influence Maximization in Social Networks. In: Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, pp. 127–132 (2011)
20. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '03, pp. 137–146 (2003)
21. Leskovec, J., Krause, A., Guestrin, C., Faloutsos, C., VanBrienen, J., Glance, N.: Cost-effective Outbreak Detection in Networks. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining- KDD '07, pp. 420–429 (2007)
22. Mastrandrea, R., Fournet, J., Barrat, A.: Contact patterns in a high school: a comparison between data collected using wearable sensors, contact diaries and friendship surveys. *PLoS one* **10**(9), e0136497 (2015)
23. Mihara, S., Tsugawa, S., Ohsaki, H.: Influence Maximization Problem for Unknown Social Networks. In: Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015 - ASONAM '15, pp. 1539–1546 (2015)
24. Morone, F., Makse, H.A.: Influence maximization in complex networks through optimal percolation. *Nature* **524**(7563), 65–68 (2015)

25. Murata, T., Koga, H.: Methods for Influence Maximization in Dynamic Networks. In: Proceedings of the 6th International Conference on Complex Networks and Their Applications (Complex Networks 2017), Studies in Computational Intelligence, pp. 955–966. Springer, Berlin (2017)
26. Murata, T., Koga, H.: Extended Methods for Influence Maximization in Dynamic Networks. *Comput. Soc. Networks* **5**(8), 1–21 (2018)
27. Ohsaka, N., Akiba, T., Yoshida, Y., Kawarabayashi, K.i.: Fast and Accurate Influence Maximization on Large Networks with Pruned Monte-Carlo Simulations. In: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, pp. 138–144 (2014)
28. Osawa, S., Murata, T.: Selecting Seed Nodes for Influence Maximization in Dynamic Networks. In: Proceedings of the 6th Workshop on Complex Networks (CompleNet 2015), Studies in Computational Intelligence, pp. 91–98. Springer, Berlin (2015)
29. Stehlé, J., Voirin, N., Barrat, A., Cattuto, C., Isella, L., Pinton, J.F., Quaggiotto, M., den Broeck, W., Régis, C., Lina, B., et al.: High-resolution measurements of face-to-face contact patterns in a primary school. *PloS one* **6**(8), e23176 (2011)
30. Tang, Y., Xiao, X., Shi, Y.: Influence maximization: Near-optimal time complexity meets practical efficiency. In: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, pp. 75–86 (2014)
31. Vanhems, P., Barrat, A., Cattuto, C., Pinton, J.F., Khanafer, N., Régis, C., Kim, B.a., Comte, B., Voirin, N.: Estimating potential infection transmission routes in hospital wards using wearable proximity sensors. *PloS one* **8**(9), e73970 (2013)