Influence Diffusion in Social Networks under Time Window Constraints*

Luisa Gargano[1], Pavol Hell[2], Joseph Peters[2], and Ugo Vaccaro[1]

[1] Dipartimento di Informatica, University of Salerno, Italy
[2] School of Computing Science, Simon Fraser University, Canada

Abstract. We study a combinatorial model of the spread of influence in networks that generalizes existing schemata recently proposed in the literature. In our model agents change behaviors/opinions on the basis of information collected from their neighbors in a time interval of *bounded size* whereas agents are assumed to have *unbounded memory* in previously studied scenarios. In our mathematical framework, one is given a network $G = (V, E)$, an integer value $t(v)$ for each node $v \in V$, and a time window size λ. The goal is to determine a small set of nodes (*target set*) that influences the whole graph. The spread of influence proceeds in rounds as follows: initially all nodes in the target set are influenced; subsequently, in each round, any uninfluenced node v becomes influenced if the number of its neighbors that have been influenced in the previous λ rounds is greater than or equal to $t(v)$. We prove that the problem of finding a minimum cardinality target set that influences the whole network G is hard to approximate within a polylogarithmic factor. On the positive side, we design exact polynomial time algorithms for paths, rings, trees, and complete graphs.

1 Introduction

Many phenomena can be represented by dynamical processes on networks. Examples include cascading failures in physical infrastructure networks [21], information cascades in social and economic systems [8], spreads of infectious diseases [2], and the spreading of ideas, fashions, or behaviors among people [12, 40]. Therefore, it comes as no surprise that the study of dynamical processes on complex networks is an active area of research, crossing a variety of different disciplines. Epidemiologists, social scientists, physicists, and computer scientists have studied diffusion phenomena using very similar models to describe the spreading of diseases, knowledge, behaviors, and innovations among individuals of a population (see [4, 9, 24] for surveys of the area).

A particularly important diffusion process is that of *viral marketing* [30], which refers to the spread of information about products and behaviors and their adoption by people. Recently, it has also become an important tool in the communication strategies of politicians [31, 39]. Although there are many similarities

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-3-319-03758-9_29

* This research was supported in part by the Ebco Eppich Endowment Fund at Simon Fraser University and by NSERC of Canada.

T. Moscibroda and A.A. Rescigno (Eds.): SIROCCO 2013, LNCS 8179, pp. 141–152, 2013.

between social and epidemiological contagion [23], social contagion is usually an intentional act on the part of the transmitter and/or the adopter, unlike a pathogen contagion. The spread of ideas requires extra mechanisms in addition to mere exposure, e.g., some kind of "social pressure". More importantly, in the marketing scenario one is interested in *maximizing* the spread of information [22], while this is not likely to happen in the spread of pathogenic viruses. The intent of maximizing the spread of viral information across a network naturally suggests many optimization problems. Some of them were first articulated in the seminal papers [27, 28], under various adoption paradigms. In the next section, we will explain and motivate our model of information diffusion, state the problem that we are investigating, describe our results, and discuss how they relate to the existing literature in the area.

Due to space constraints, proofs are omitted from this extended abstract.

2 The Model, the Context, and the Results

The network is represented by a pair (G, t), where $G = (V, E)$ is an undirected graph and $t : V \longrightarrow \mathbb{N} = \{1, 2, \ldots, \}$ is a function assigning integer thresholds to nodes. We assume that $1 \leq t(v) \leq \deg(v)$ for each $v \in V$, where $deg(v)$ is the degree of v. For a given set $S \subseteq V$ and a time window size $\lambda \in \mathbb{N}$, we consider a dynamical process of influence diffusion in G defined by two sequences of node subsets, $\texttt{Influenced}[S, r]$ and $\texttt{Active}[S, r]$, $r = 0, 1, \ldots,$ where

$\texttt{Influenced}[S, 0] = S$, $\texttt{Active}[S, 0] = \emptyset$, and for any $r \geq 1$ it holds that

$$\texttt{Influenced}[S, r] = \texttt{Influenced}[S, r-1] \cup \left\{v : \left|N(v) \cap \texttt{Active}[S, r]\right| \geq t(v)\right\} \tag{1}$$

$$\texttt{Active}[S, r] = \begin{cases} \texttt{Influenced}[S, r-1] & \text{if } r \leq \lambda \\ \texttt{Influenced}[S, r-1] \setminus \texttt{Influenced}[S, r-1-\lambda] & \text{if } r > \lambda \end{cases}$$

Intuitively, the set S might represent a group of people who are initially influenced/convinced to adopt a product or an idea. Then the cascade proceeds in rounds. In each round r, the set of influenced nodes is augmented by including each node v that has a number of influenced and *still active* neighbors greater than or equal to its threshold $t(v)$. A node is active for λ rounds after it becomes influenced and then it becomes inactive.

Our model is based on the models in [20, 33] which assume that people can be divided into three classes at any time instant. *Ignorants* are those not aware of a rumor/not yet influenced, *spreaders* are those who are spreading it, and *stiflers* are those who know the rumor/have been influenced but have ceased to spread the rumor/influence.[1] Several rules have been proposed to govern the transition from ignorants to spreaders and from spreaders to stiflers, and many papers have studied the dynamics of these systems, mostly in stochastic scenarios

[1] The reader will notice an analogy with the SIR model of mathematical epidemiology [2], in which individuals can be classified as Susceptible, Infected, and Recovered.

(see [7, 34] and references quoted therein). Here, we posit that any ignorant node becomes a spreader if the number of its neighbors who are spreaders is above a certain threshold (i.e., the node is subject to a large enough amount of "social pressure"), and any spreader becomes a stifler after λ rounds (because the spreader loses interest in the rumor, for instance). Other papers have studied information diffusion under similar assumptions [19, 26].

Our model also captures another important characteristic of influence diffusion. It is well known (e.g. [3]) that people are more inclined to react to pieces of information cumulatively heard during a "short" time interval than to information heard during a considerably longer period of time. In other words, one is more likely to be convinced of an opinion heard from a certain number of friends during the last few days than by an opinion heard sporadically during the last year from the *same* number of people. Therefore, it seems reasonable to study diffusion processes in which people have *bounded memory*, and only the number of spreaders heard during the last λ rounds may contribute to the change of status of an ignorant node.[2] Formally, one has a dynamical process of influence diffusion on G described by the sequence of node subsets $\texttt{Influenced}'[S,r]$, $r = 0, 1, \ldots$, where $\texttt{Influenced}'[S,0] = S$, and for any $r \geq 1$ it holds that

$$\begin{aligned}\texttt{Influenced}'[S,r] &= \\ &= \texttt{Influenced}[S,r-1] \cup \left\{v : \left|N(v) \cap \texttt{Influenced}'[S,r-1]\right| \geq t(v)\right\} \qquad (2)\end{aligned}$$

if $r \leq \lambda$, and

$$\begin{aligned}\texttt{Influenced}'[S,r] &= \texttt{Influenced}'[S,r-1] \qquad (3)\\ &\cup \left\{v : \left|N(v) \cap (\texttt{Influenced}'[S,r-1] \setminus \texttt{Influenced}'[S,r-1-\lambda])\right| \geq t(v)\right\}\end{aligned}$$

if $r > \lambda$. It is immediate that (2) and (3) are an equivalent way to write (1) and (2): for any $S \subseteq V$ and $r \geq 1$, $\texttt{Influenced}'[S,r] = \texttt{Influenced}[S,r]$, so we get that the spreading process with "stiflers" also describes the spreading process with "bounded memory" governed by (2) and (3).

Summarizing, the problem that we shall study in this paper is the following:

Time Window Constrained Target Set Selection (TWC–TSS)
Input: A graph $G = (V, E)$, a threshold function $t : V \longrightarrow \mathbb{N}$, and a time window size λ.
Output: A minimum size $S \subseteq V$ s.t. $\texttt{Influenced}[S,r] = V$, for some $r \geq 0$.

When λ is large enough, for instance equal to the number n of nodes, our Time Window Constrained Target Set Selection problem is equivalent to the classical Target Set Selection problem studied in [1, 5, 6, 10, 13–18, 37, 41]. In terms of our second formulation of the TWC–TSS problem, the classical Target Set Selection problem can be viewed as an extreme case in which it is assumed that people have unbounded memory. In general, the TWC–TSS and the TSS problems are quite different. One of the main difficulties of the new TWC–TSS problem is that the sequence of sets $\texttt{Active}[S,r]$, $r = 0, 1, \ldots$ is not necessarily monotonically non-decreasing: it is possible that $\texttt{Active}[S,r]$

[2] Another model in which individuals carry a memory of the "amount of influence" received during a bounded time interval has been studied in [23].

is larger than $\texttt{Active}[S, r+1]$ for some values of r. When $\lambda = n$, we have $\texttt{Active}[S, r] = \texttt{Influenced}[S, r-1]$ for any r, and monotonicity is restored. At the other extreme, when $\lambda = 1$, a node v becomes influenced at time r only if at least $t(v)$ of its neighbors become influenced at *exactly* time $r-1$. This sort of synchronization in the propagation of influence poses new challenges, both in the assessment of the computational complexity of the TWC–TSS problem and, especially, in the design of algorithms for its solution. The example in which the graph G is a path is particularly illuminating. As we shall see in Section 4.1, the Target Set Selection problem is trivial to solve on a path; it is far from being so when there is a fixed time window size λ.

Our Results. In Section 3, we prove a polylogarithmic inapproximability result for the TWC–TSS problem under a plausible computational complexity assumption. The result is obtained by a modification of the very clever proof of the inapproximability of TSS by Chen [13]. In view of the strong inapproximability of the TWC–TSS problem, we then turn our attention to special cases of the problem. In Section 4 we present the main results of the paper: exact polynomial time algorithms for paths, rings, complete graphs, and trees. The algorithms for paths and rings are based on dynamic programming, and the algorithm for complete graphs is greedy. The algorithm for trees is also based on dynamic programming and requires the solution of polynomially many integer linear programs. The polynomial time solvability of each integer linear program is guaranteed by the unimodularity of the associated matrix of coefficients.

3 Hardness of TWC–TSS

In general, our optimization problem TWC–TSS is unlikely to be efficiently approximable, as the following result shows.

Theorem 1. *For any fixed value of the time window size λ, the TWC–TSS problem cannot be approximated within a ratio of $O(2^{\log^{1-\epsilon} n})$ for any fixed $\epsilon > 0$, unless $NP \subseteq DTIME(n^{polylog(n)})$.*

Theorem 1 is a generalization of a similar inapproximability result given in [13] for the Target Set Selection problem that, as said before, corresponds to our Time Window Constrained Target Set Selection problem when the time window size λ is unbounded. Our result holds for any fixed value of λ. The proof details are presented in the Appendix; here we sketch the main idea. We prove Theorem 1 by a polynomial time reduction from the same `MIN REP` problem used in [13].

Let $H = (V_A \cup V_B, E)$ be a bipartite graph, where $V_A \cap V_B = \emptyset$ and $E \subseteq V_A \times V_B$. Let $\mathcal{A}$ be a family of subsets of V_A that partitions V_A into $|\mathcal{A}|$ equally sized subsets, and analogously let the family $\mathcal{B}$ be a partition of V_B into $|\mathcal{B}|$ equally sized subsets. Given graph H and partitions $\mathcal{A}$, $\mathcal{B}$, the `MIN REP` problem asks for a subset $U \subseteq V$ of minimum size such that for each $A \in \mathcal{A}$ and $B \in \mathcal{B}$

$$E \cap (A \times B) \neq \emptyset \text{ implies } [E \cap (A \times B)] \cap (U \times U) \neq \emptyset. \tag{4}$$

Theorem 2. [13] *The* MIN REP *problem cannot be approximated within a ratio of* $O(2^{\log^{1-\epsilon} n})$ *for any fixed* $\epsilon > 0$*, unless* $NP \subseteq DTIME(n^{polylog(n)})$.

Given an instance of MIN REP consisting of the bipartite graph $H = (V_A \cup V_B, E)$ and the pair of partitions $(\mathcal{A}, \mathcal{B})$, we construct an instance $\mathcal{I}$ for the TWC–TSS problem. More precisely, for the instance $\mathcal{I}$ we will only specify a suitable graph $G = (V, E)$ and threshold function $t : V \longrightarrow \mathbb{N} = \{1, 2, \ldots, \}$, since our aim is to prove inapproximability for *any* value of λ. We denote by Γ_ℓ the gadget shown in Figure 1(a), which consists of ℓ paths of length 2 connecting the same pair of nodes. If $\lambda \leq 8$, we need another gadget Γ_ℓ^λ shown in Figure 1(b); it consists of ℓ paths, each having length $11 - \lambda$ and connecting the same pair of extremal nodes. All internal nodes of the gadgets have threshold 1.

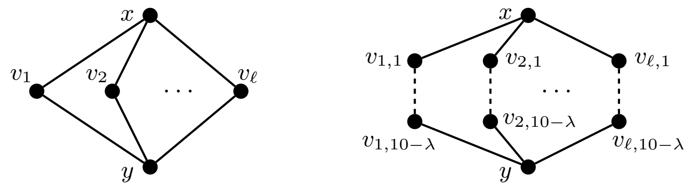

Fig. 1. (a) The gadget Γ_ℓ consisting of ℓ paths of length 2 sharing the extremal nodes. (b) The gadget Γ_ℓ^λ consisting of ℓ paths of length $11 - \lambda$ sharing the extremal nodes.

Let $N = |V| + |E|$. The graph G has node set $V_1 \cup V_2 \cup V_3 \cup V_4$ where

- $V_1 = V$ and each node has threshold N^2,
- $V_2 = \{x_{(a,b)} : (a, b) \in E\}$; each node $x_{(a,b)} \in V_2$ has threshold $2N^5$. The node $x_{(a,b)}$ is connected to both $a \in V_1$ and $b \in V_1$ by a gadget Γ_{N^5}; moreover, if $\lambda \leq 8$ then $x_{(a,b)}$ is also connected to both a and b by a gadget $\Gamma_{N^5}^\lambda$.
- $V_3 = \{y_{A,B} : (A \times B) \cap E \neq \emptyset\}$; each node $y_{A,B} \in V_3$ has threshold N^4 and is connected by a gadget Γ_{N^4} to each $x_{(a,b)} \in V_2$ with $a \in A$ and $b \in B$, and
- $V_4 = \{z_1, \ldots, z_N\}$; each node $z \in V_4$ has threshold $|V_3| \times N^2$ and is connected by a gadget Γ_{N^2} to each node in V_3 and by a gadget Γ_N to each node in V_1.

Theorem 1 follows by showing that any optimal solution U of the MIN REP instance gives rise to a solution $U \subseteq V_1$ to the TWC–TSS problem with input instance (G, t, λ). Vice versa, if S is a solution to the TWC–TSS istance, then in polynomial time one can construct a MIN REP solution of size at most $2|S|$.

4 Polynomially Solvable Cases of TWC–TSS

We now present exact polynomial time algorithms to solve the TWC–TSS problem in several classes of graphs.

4.1 Paths

Let $L^n = (V, E)$ be a path on n nodes, with $V = \{0, \ldots, n-1\}$ and $E = \{(v, v+1) : 0 \leq v \leq n-2\}$. Since the threshold of each node cannot exceed its degree, we have that $t(0) = t(n-1) = 1$ and $t(v) \in \{1, 2\}$, for each $v = 1, \ldots, n-2$.

The TWC–TSS problem is trivial to solve in case λ is unbounded. Letting $\{v_{i_1}, v_{i_2}, \ldots, v_{i_k}\}$ be the nodes of L^n having threshold equal to 2, one can see that $\{v_{i_1}, v_{i_3}, \ldots, v_{i_{k-2}}, v_{i_k}\}$ is an optimal solution when k is odd, whereas the subset $\{v_{i_1}, v_{i_3}, \ldots, v_{i_{k-1}}, v_{i_k}\}$ is optimal when k is even. In case λ has some fixed value, the situation is much more complicated. Indeed, because of the time window constraint, one must judiciously choose the initial target set in such a way that, for every node with threshold 2 that does not belong to the initial target set, its two neighbors become influenced at the correct times.

To avoid trivialities, we assume that L^n has at least two nodes with threshold equal to 2. Should it be otherwise, for instance all nodes have threshold 1, then any subset S of V with $|S| = 1$ is an optimal solution. If exactly one node, say v, has threshold 2, then $\{v\}$ is an optimal solution.

Lemma 1. *If $\ell = \min\{v \in V : t(v) = 2\}$ and $s = \max\{v \in V : t(v) = 2\}$, then there exists an optimal solution S such that*
i) $S \cap \{0, \ldots \ell - 1\} = \emptyset = S \cap \{s + 1, \ldots n - 1\}$;
ii) $\ell, s \in S$.

Lemma 1 implies that we can ignore all nodes in L^n that are to the left of the lowest numbered node with threshold 2, and to the right of the highest numbered node with threshold 2. Equivalently, from now on we can assume that $t(0) = t(n-1) = 2$. Define the array $D[0 \ldots (n-1)]$, where $D[n-1] = n-1$ and

$$D[i] = \min\{j \ : \ i < j \leq n-1 \text{ and } t(j) = 2\}, \tag{5}$$

for $i = 0, \ldots, n-2$. Since $t(n-1) = 2$, value $D[i]$ is always well defined. One can check that the following algorithm computes an array D satisfying (5).

Algorithm ARRAY(L^n) *[Input: A path L^n with threshold function $t(\cdot)$]*
1. Set $D[n-1] = n-1$ and $j = n-1$
2. for $i = (n-2)$ **down to** 0 **do**
3. set $D[i] = j$
4. **if** $t(i) = 2$ **then** set $j = i$

For each $i = 0, \ldots, n-1$, let L^n_i denote the sub-path consisting of the last $n-i$ nodes $\{i, i+1, \ldots, n-1\}$ of L^n. We denote by $\mathtt{s}(i)$ the minimum size of a TWC target set for L^n_i that contains both the extreme nodes, that is, i and $n-1$. Our first goal is to compute $\mathtt{s}(0)$, the size of an optimal solution for $L^n_0 = L^n$.

Lemma 2. *Fix the time window size λ and consider the family of all TWC target sets for L^n_i that include both i and $n-1$. If $i < n-1$, such a family contains a minimum size TWC target set whose second smallest element is in*

$$\{D[i]\} \cup \{x : \max\{D[i]+1,\ 2D[i]-i-\lambda+1\} \leq x \leq \min\{2D[i]-i+\lambda-1,\ D[D[i]]\}\}. \tag{6}$$

From Lemma 2, we have $\mathtt{s}(n-1) = 1$ and, for each $i = 0, \ldots, n-2$,

$$\mathtt{s}(i) = 1 + \min\left\{\mathtt{s}(D[i]),\ \min_j \mathtt{s}(j)\right\} \tag{7}$$

where j satisfies $\max\{D[i]+1,\, 2D[i]-i-\lambda+1\} \le j \le \min\{2D[i]-i+\lambda-1,\, D[D[i]]\}$. The size of an optimal target set for L_n can be computed as $\mathtt{s}(0)$.

The actual TWC target set of optimal size $\mathtt{s}(0)$ can be constructed using standard backtracking techniques.

Theorem 3. *For any time window size λ, an optimal TWC target set for the path L^n can be computed in time $O(n)$.*

4.2 Rings

We can use Theorem 3 above to design an algorithm for the TWC–TSS problem on rings. Let R^n denote the ring on n nodes $\{0,\dots,n-1\}$ with edges $(i,(i+1) \bmod n)$ and thresholds $t(i)$, for $i=0,\dots,n-1$.

We first notice that if all nodes have threshold 2, then an optimal TWC target set for R^n trivially has size $\lceil n/2\rceil$, so let us now assume that there exists a node j that has threshold $t(j)=1$. Either j is in an optimal TWC target set for R^n or it is not. Consider the path $R^n_{j,2}$ obtained by "breaking" the ring R^n at node j, duplicating node j into j and j', and assigning threshold 2 to both j and j' (regardless of the original threshold value $t(j)=1$ in R^n). Therefore, the edges of $R^n_{j,2}$ are $(j,j{+}1),(j{+}1,j{+}2),\dots,(n{-}2,n{-}1),(n{-}1,0),\dots(j{-}2,j{-}1),(j{-}1,j')$. The thresholds of $R^n_{j,2}$ are

$$t_{j,2}(i) = \begin{cases} t(i) & \text{if } 0 \le i \le n-1 \text{ and } i \ne j \\ 2 & \text{if } i=j \text{ or } i=j'. \end{cases}$$

We can use the algorithm of Section 4.1 to compute the size of an optimal TWC target set $S_{j,2}$ for the path $R^n_{j,2}$. Notice that both j and j' must be in $S_{j,2}$, so $S_{j,2}-\{j'\}$ is a TWC target set for the ring R^n, optimal among all TWC target sets that include node j.

Now we want to compute a TWC target set for the ring R^n that is optimal among all TWC target sets that do not include node j. To do this, consider the path $R^n_{j,1}$ that has the same nodes and edges as $R^n_{j,2}$ but has thresholds

$$t_{j,1}(i) = \begin{cases} t(i) & \text{if } 0 \le i \le n-1 \text{ and } i \ne j \\ 1 & \text{if } i=j \text{ or } i=j'. \end{cases}$$

In particular, the endpoints of $R^n_{j,1}$ have thresholds $t_{j,1}(j)=t_{j,1}(j')=1$. First, we apply Lemma 1 to $R^n_{j,1}$ and then we use the algorithm of Section 4.1 to compute (the size of) an optimal TWC target set $S_{j,1}$. Since $j,j' \notin S_{j,1}$, we have that $S_{j,1}$ is a TWC target set for the ring R^n, optimal among all TWC target sets that do not include node j.

An optimal solution for the ring R^n is then obtained by choosing the smaller of $S_{j,2}-\{j'\}$ and $S_{j,1}$. In conclusion we have the following result.

Theorem 4. *For any value of the time window size λ, an optimal TWC target set for the ring R^n can be computed in time $O(n)$.*

4.3 Trees

Let $T=(V,E)$ be a tree with threshold function $t: V \longrightarrow \mathbb{N}$, and let $\lambda \ge 1$ be a fixed value of the time window size. We consider T to be rooted at some arbitrary node $p \in V$. For each node $v \in V$, we denote by $T_v = (V_v, E_v)$ the subtree of T rooted at v. Moreover, we denote by $\texttt{Ch}(v)$ the set of all children of node v in T_v.

Definition 1. *Given node $v \in V$ and integers t, r, with $t \in \{t(v), t(v)-1\}$ and $r \geq 0$, we denote by $\texttt{s}(v,t,r)$ the minimum size of a TWC target set $S \subseteq V_v$ for subtree T_v that influences node v in round r (that is, $v \in \texttt{Influenced}[S,r] \setminus \texttt{Influenced}[S,r-1]$), under the assumption that v has threshold t in T_v. The threshold of each other node $w \neq v$ in T_v is the original one $t(w)$.*

The size of an optimal TWC target set for the tree T can be computed as

$$\min_r \texttt{s}(p, t(p), r), \tag{8}$$

where r ranges between 0 and the maximum possible number of rounds needed to complete the influence diffusion process. The number of rounds is always upper bounded by the number of nodes in the graph (since at least one new node must be influenced in each round before the diffusion process stops). However, for a tree T, this value is upper bounded by the length of the longest path in T. In other words, the parameter r in Definition 1 is bounded by the diameter $diam(T)$ of T.

We use a dynamic programming approach to compute the value in (8). Then, the corresponding optimal TWC target set S can be built using standard back-tracking techniques. In our dynamic programming algorithm we compute all of the values

$$\texttt{s}(v,t,r) \text{ for each } v \in V,\ t \in \{t(v), t(v)-1\} \text{ and } r = 0, \ldots, diam(T),$$

and the computation is performed according to a breadth-first search (BFS) reverse ordering of the nodes of T, so that each node v is considered only when all of the values $\texttt{s}(\cdot,\cdot,\cdot)$ for all of its children are known. The rationale behind the computation of both $\texttt{s}(v,t(v),r)$ and $\texttt{s}(v,t(v)-1,r)$ is the following:

i) $\texttt{s}(v,t(v),r)$ corresponds to the case of a target set S for tree T such that
- $v \in \texttt{Influenced}[S,r] \setminus \texttt{Influenced}[S,r-1]$ and
- at least $t(v)$ of v's children belong to $\texttt{Active}[S \cap V_v, r] \subseteq \texttt{Active}[S,r]$;

ii) $\texttt{s}(v,t(v)-1,r)$ is the size of an optimal target set S for T satisfying
- $v \in \texttt{Influenced}[S,r] \setminus \texttt{Influenced}[S,r-1]$,
- $\texttt{Active}[S,r]$ contains v's parent in T, and
- at least $t(v)-1$ of v's children belong to $\texttt{Active}[S \cap V_v, r] \subseteq \texttt{Active}[S,r]$.

In the following, we show how to compute the above values $\texttt{s}(\cdot,\cdot,\cdot)$. The procedure is summarized in algorithm TREE.

First, consider the computation of $\texttt{s}(v,t,r)$ when v is a leaf of T. In this case we have $t(v) = deg(v) = 1$.
– If $r = 0$, v trivially must belong to the target set since v needs to be active at time 0; hence $\texttt{s}(v,t,0) = 1$.
– If $r > 0$ and $t = t(v) = 1$, we observe that any TWC target set that influences leaf v at time *exactly* r cannot contain v and, therefore, must influence v's parent at time $r-1$. To do so, we set $\texttt{s}(v,1,r) = \infty$ in the algorithm; this forces the minimum at line **14** or **18** to be reached with threshold $t(v)-1 = 0$, thus forcing

v's parent to be active in round $r-1$.
– If $r > 0$ and $t = t(v) - 1 = 0$ then, trivially, $\mathtt{s}(v, 0, r) = 0$.

Algorithm TREE(T, p, λ, t)

[`Input`*: Tree T rooted at p, time window size λ, threshold function t.]*

For each $v \in T$ in reverse order to a BFS of T
  [We compute $\mathtt{s}(v,t,r)$ for each $t \in \{t(v), t(v)-1\}$ and $0 \leq r \leq$ $diam(T)$]
If v is a leaf **then** *[here $t(v) = 1$]*
  For $r = 0, \ldots, diam(T)$
    Set $\mathtt{s}(v, 0, r) = 0$ and $\mathtt{s}(v, 1, r) = \begin{cases} 1 & \text{if } r = 0 \\ \infty & \text{otherwise} \end{cases}$
If v is NOT a leaf in T **then**
  For each $\big(\ r = 0, \ldots, diam(T)$ AND $t \in \{t(v), t(v)-1\}$ (only $t = t(v)$ if $v = p$) $\big)$
    If $r = 0$ **then**
      Set $\mathtt{s}(v,t,0) = 1 + \sum_{w \in \mathtt{Ch}(v)} \min\{\min_{1 \leq j \leq \lambda} \mathtt{s}(w, t(w)-1, j), \min_{j \geq 0} \mathtt{s}(w, t(w), j)\}$
    If $r \geq 1$ and $t = 0$ **then**
      Set $\mathtt{s}(v, 0, r) = \sum_{w \in \mathtt{Ch}(v)} \min\{\min_{r+1 \leq j \leq r+\lambda} \mathtt{s}(w, t(w)-1, j), \min_{j \geq r-1} \mathtt{s}(w, t(w), j)\}$
    If $r \geq 1$ and $t = 1$ **then**
      For each $w \in Ch(v)$
        Compute $m(w) = \min\{\min_{r+1 \leq j \leq r+\lambda} \mathtt{s}(w, t(w)-1, j), \min_{j \geq r-1} \mathtt{s}(w, t(w), j)\}$
      Set $z = \mathrm{argmin}_{w \in \mathtt{Ch}(v)} \{\mathtt{s}(w, t(w), r-1) - m(w)\}$
      Set $\mathtt{s}(v, 1, r) = \sum_{w \in \mathtt{Ch}(v) \setminus \{z\}} m(w) + \mathtt{s}(z, t(z), r-1)$
    If $r \geq 1$ and $t > 1$ **then**
      Set $\mathtt{s}(v,t,r) = \min \sum_{w \in \mathtt{Ch}(v)} m(w)$, where
      $m(w) \in \{\mathtt{s}(w,t,j)\ :\ (t = t(w)$ AND $j \geq 0)$ OR $(t = t(w)-1$ AND $r < j \leq r+\lambda)\}$
      $|\{w\ :\ m(w) = \mathtt{s}(w, t(w), j),\ r - \lambda \leq j \leq r-1\}| \geq t$
      $|\{w\ :\ m(w) = \mathtt{s}(w, t(w), j),\ \ell - \lambda \leq j \leq \ell - 1\}| < t, \quad \forall \ell = 1, \ldots, r-1$

Now consider an arbitrary internal node v. Since we process nodes in a BFS reverse order, each child of v has already been processed when the algorithm processes v. If $r = 0$, then v must necessarily be in the target set and any $w \in \mathtt{Ch}(v)$ can benefit from this. Therefore, the size $\mathtt{s}(v,t,0)$ of an optimal solution for the subtree T_v is equal to

$$\mathtt{s}(v,t,0) = 1 + \sum_{w \in \mathtt{Ch}(v)} \min\left\{ \min_{1 \leq j \leq \lambda} \mathtt{s}(w, t(w)-1, j), \min_{0 \leq j \leq diam(T)} \mathtt{s}(w, t(w), j) \right\}.$$

Notice that we have constrained j to be in the range $1, \ldots, \lambda$ in the formula above when w's threshold is $t(w) - 1$. This is correct since v is active and able to influence w only in rounds $j = 1, \ldots, \lambda$.
Now, let us consider the computation of $\mathtt{s}(v,t,r)$ with $r \geq 1$, that is, when v is not part of the target set and v is influenced at time r by t of its children (plus its parent if $t = t(v) - 1$). To determine the optimal solution, we need to know the best among the values $\mathtt{s}(w, \tau, j)$ for each $w \in \mathtt{Ch}(v)$ and for all possible values of parameters τ and j, subject to the following two constraints:

1) if $\tau = t(w) - 1$, then $r + 1 \leq j \leq r + \lambda$ (indeed v is active and can influence w only during the λ rounds after it has become influenced, that is, in rounds $j = r+1, \ldots, r+\lambda$),

2) at least t nodes in $\texttt{Ch}(v)$ are active in round r but at most $t-1$ are active in any previous round $j \leq r-1$ (otherwise v would become influenced before the required round r).

The special case $t = 0$ can hold only if $t(v) = 1$ and $t = t(v) - 1$; hence node v must be influenced by its parent at round r and none of its children can be active before round r.

Lemma 3. *The computation at lines 18–21 of the algorithm TREE(T, p, λ) can be done in polynomial time.*

Theorem 5. *For any tree T, the optimal TWC target set can be computed in polynomial time.*

4.4 Complete Graphs

Let $K_n = (V, E)$ denote the complete graph on n nodes. The following observation was made in [35] for target set selection without a time window constraint; it is easy to see that it also holds in our scenario.

Lemma 4. [35] *If the optimal TWC target set for K_n has size k, then there exists an optimal TWC target set consisting of k nodes with the largest thresholds.*

Lemma 4 follows from the observation that in any target set S for K_n, if there exist $v \in S$ and $u \in V - S$ with $t(v) < t(u)$, then $S \setminus \{v\} \cup \{u\}$ is also a target set for K_n. Lemma 4 implies that we only need to determine the size of an optimal TWC target set. The following algorithm $MAX(n, k)$ determines the largest number of nodes that can be influenced using a TWC target set of k nodes. The algorithm assumes that the thresholds have been sorted in non-decreasing order. Moreover, it assumes the precomputation of the integer vector $A[1..n-1]$ such that $A[\ell] = |\{v \in V \mid t(v) \leq \ell\}|$, for $\ell = 1, \ldots, n-1$. Notice that both sorting the thresholds, by counting sort, and computing A can be done in linear time.

Algorithm $MAX(n, k)$ *[Input: vector A[1..n-1], parameters λ and k]*

Set $\ell = k$
If $A[\ell] > 0$ **then** *[at least one node outside the target set can be influenced]*
  For $j = 0, \ldots, \lambda - 2$
    Set $X[j] = -k$
  Set $X[\lambda - 1] = 0$, Set $j = 0$;
  Repeat
    Set $y = A[\ell]$, $\ell = A[\ell] - X[j]$, $X[j] = y$, $j = (j+1) \bmod \lambda$
  Until $(A[\ell] - X[j] \leq \ell$ OR $A[\ell] + k \geq n)$
Output $\min\{n, k + A[\ell]\}$

We can show that the algorithm $MAX(n, k)$ requires $O(n)$ time to compute the largest number of nodes that can be influenced in K_n using a TWC target set of size k. Using a binary search for the optimal value of k we obtain the following result.

Theorem 6. *The optimal TWC target set in a complete graph K_n can be computed in time $O(n \log n)$.*

References

1. Ackerman, E., Ben-Zwi, O., Wolfovitz, G.: Combinatorial model and bounds for target set selection. Theoretical Computer Science 411, 4017–4022 (2010)
2. Anderson, R.M., May, R.M.: Infectious Diseases of Humans: Dynamics and Control. Oxford University Press (1991)
3. Banos, R.A., Borge-Holthoefer, J., Moreno, Y.: The role of hidden influentials in the diffusion of online information cascades. arXiv:1303.4629v1 [physics.soc-ph] (2013)
4. Barrat, A., Barthelemy, M., Vespignani, A.: Dynamical Processes on Complex Networks. Cambridge University Press (2012)
5. Bazgan, C., Chopin, M., Nichterlein, A., Sikora, F.: Parameterized Approximability of Maximizing the Spread of Influence in Networks. arXiv:1303.6907 [cs.DS]
6. Ben-Zwi, O., Hermelin, D., Lokshtanov, D., Newman, I.: Treewidth governs the complexity of target set selection. Discrete Optimization 8, 87–96 (2011)
7. Bettencourt, L.M.A., Cintron-Arias, A., Kaiser, D.I., Castillo-Chavez, C.: The power of a good idea: Quantitative modeling of the spread of ideas from epidemiological models. Physica A 364, 513–536 (2006)
8. Bikhchandani, S., Hirshleifer, D., Welch, I.: A theory of fads, customs, and cultural change as informational cascades. Journal of Political Economy 100, 992–1026 (1992)
9. Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., Hwang, D.-U.: Complex Networks: Structure and Dynamics. Physics Reports 472, 175–308 (2006)
10. Centeno, C.C., Dourado, M.C., Penso, L.D., Rautenbach, D., Szwarcfiter, J.L.: Irreversible conversion of graphs. Theoretical Computer Science 412, 3693–3700 (2011)
11. Centola, D.: The spread of behavior in an online social network experiment. Science 329, 1194–1197 (2010)
12. Centola, D., Macy, M.: Complex contagions and the weakness of long ties. American Journal of Sociology 113, 702–734 (2007)
13. Chen, N.: On the approximability of influence in social networks. SIAM J. Discrete Math. 23, 1400–1415 (2009)
14. Chiang, C.Y., Huang, L.H., Huang, W.T., Yeh, H.G.: The Target Set Selection Problem on Cycle Permutation Graphs, Generalized Petersen Graphs and Torus Cordalis. arXiv:1112.1313 (2011)
15. Chopin, M., Nichterlein, A., Niedermeier, R., Weller, M.: Constant Thresholds Can Make Target Set Selection Tractable. In: Even, G., Rawitz, D. (eds.) MedAlg 2012. LNCS, vol. 7659, pp. 120–133. Springer, Heidelberg (2012)
16. Chiang, C.-Y., Huang, L.-H., Li, B.-J., Wu, J., Yeh, H.-G.: Some results on the target set selection problem. Journal of Combinatorial Optimization (to appear)
17. Chiang, C.-Y., Huang, L.-H., Yeh, H.-G.: Target Set Selection Problem for Honeycomb Networks. SIAM J. Discrete Math. 27(1), 310–328 (2013)
18. Cicalese, F., Cordasco, G., Gargano, L., Milanič, M., Vaccaro, U.: Latency-Bounded Target Set Selection in Social Networks. In: Bonizzoni, P., Brattka, V., Löwe, B. (eds.) CiE 2013. LNCS, vol. 7921, pp. 65–77. Springer, Heidelberg (2013)
19. Comellas, F., Mitjana, M., Peters, J.G.: Broadcasting in Small-world Communication Networks. In: SIROCCO. Proceedings in Informatics, Carleton Scientific, vol. 13, pp. 73–85 (2002)
20. Daley, D.J., Kendall, D.G.: Epidemics and rumours. Nature 204, 1118 (1964)

21. Duenas-Osorio, L., Vemuru, S.M.: Cascading failures in complex infrastructure systems. Structural Safety 31(2), 157–167 (2009)
22. Domingos, P., Richardson, M.: Mining the network value of customers. In: Proc. of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 57–66 (2001)
23. Dodds, P.S., Watts, D.J.: A generalized model of social and biological contagion. J. Theoretical Biology 232, 587–604 (2005)
24. Easley, D., Kleinberg, J.: Networks, Crowds, and Markets: Reasoning About a Highly Connected World. Cambridge University Press (2010)
25. Flocchini, P., Královic, R., Ruzicka, P., Roncato, A., Santoro, N.: On time versus size for monotone dynamic monopolies in regular topologies. J. Discrete Algorithms 1, 129–150 (2003)
26. Karimi, F., Holme, P.: Threshold model of cascades in temporal networks. arXiv:1207.1206v2 [physics.soc-ph] (2012)
27. Kempe, D., Kleinberg, J.M., Tardos, E.: Maximizing the spread of influence through a social network. In: Proc. of the Ninth ACM SIGKDD International Conference on Knowledge, Discovery and Data Mining, pp. 137–146 (2003)
28. Kempe, D., Kleinberg, J.M., Tardos, É.: Influential Nodes in a Diffusion Model for Social Networks. In: Caires, L., Italiano, G.F., Monteiro, L., Palamidessi, C., Yung, M. (eds.) ICALP 2005. LNCS, vol. 3580, pp. 1127–1138. Springer, Heidelberg (2005)
29. Kephart, J.O., White, S.R., Chess, D.M.: Computers and epidemiology. IEEE Spectrum 30, 20–26 (1993)
30. Leskovic, H., Adamic, L.A., Huberman, B.A.: The dynamic of viral marketing. ACM Transactions on the WEB 1 (2007)
31. Leppaniemi, M., Karjaluoto, H., Lehto, H., Goman, A.: Targeting Young Voters in a Political Campaign: Empirical Insights into an Interactive Digital Marketing Campaign in the 2007 Finnish General Election. Journal of Nonprofit & Public Sector Marketing 22, 14–37 (2010)
32. Lopez-Pintado, D.: Diffusion in complex social networks. Games and Economic Behavior 62, 573–590 (2008)
33. Maki, D.P., Thompson, M.: Mathematical Models and Applications, with Emphasis on the Social, Life and Management Sciences. Prentice Hall (1973)
34. Nekovee, M., Moreno, Y., Bianconi, G., Marsili, M.: Theory of rumour spreading in complex social networks. Physica A: Statistical Mechanics and its Applications 374, 467–470 (2007)
35. Nichterlein, A., Niedermeier, R., Uhlmann, J., Weller, M.: On Tractable Cases of Target Set Selection. Social Network Analysis and Mining, 1–24 (2012)
36. Peleg, D.: Local majorities, coalitions and monopolies in graphs: a review. Theoretical Computer Science 282, 231–257 (2002)
37. Reddy, T.V.T., Rangan, C.P.: Variants of spreading messages. J. Graph Algorithms Appl. 15(5), 683–699 (2011)
38. Schrijver, A.: Combinatorial Optimization: Polyhedra and Efficiency. Springer (2004)
39. Tumulty, K.: Obama's Viral Marketing Campaign. TIME Magazine (July 5, 2007)
40. Ugander, J., Backstrom, L., Marlow, C., Kleinberg, J.: Structural Diversity in Social Contagion. Proc. of the Nat'l Academy of Sciences (PNAS) 109, 5962–5966 (2012)
41. Zaker, M.: On dynamic monopolies of graphs with general thresholds. Discrete Mathematics 312(6), 1136–1143 (2012)