Social Networks with Competing Products*

Krzysztof R. Apt
CWI, Amsterdam, the Netherlands and University of Amsterdam

Evangelos Markakis
Athens University of Economics and Business, Athens, Greece

Abstract

We introduce a new threshold model of social networks, in which the nodes influenced by their neighbours can adopt one out of several alternatives. We characterize social networks for which adoption of a product by the whole network is possible (respectively necessary) and the ones for which a unique outcome is guaranteed. These characterizations directly yield polynomial time algorithms that allow us to determine whether a given social network satisfies one of the above properties.

We also study algorithmic questions for networks without unique outcomes. We show that the problem of determining whether a final network exists in which all nodes adopted some product is NP-complete. In turn, the problems of determining whether a given node adopts some (respectively, a given) product in some (respectively, all) network(s) are either co-NP complete or can be solved in polynomial time.

Further, we show that the problem of computing the minimum possible spread of a product is NP-hard to approximate with an approximation ratio better than $\Omega(n)$, in contrast to the maximum spread, which is efficiently computable. Finally, we clarify that some of the above problems can be solved in polynomial time when there are only two products.

1 Introduction

1.1 Background

Social networks have become a huge interdisciplinary research area with important links to sociology, economics, epidemiology, computer science, and mathematics. A flurry of numerous articles, notably the influential [19], and recent books, see

*A preliminary version of this paper appeared as [2].

[6, 12, 24, 15, 9] shows the growing relevance of this field. It deals with such diverse topics as epidemics, spread of certain patterns of social behaviour, effects of advertising, and emergence of ‘bubbles’ in financial markets.

A large part of research on social networks focusses on the problem of *diffusion*, that is the spread of a certain event or information over the network, for example becoming infected or adopting a given product. In the remainder of the paper, we will use as a running example the adoption of a product, which is being marketed over a social network.

Two prevalent models have been considered for capturing diffusion: the *threshold models* introduced in [13] and [21] and the *independent cascade models* studied in [10]. In the threshold models, each node i has a threshold $\theta(i) \in (0,1]$ and it decides to adopt a product when the total weight of incoming edges from nodes that have already adopted a product exceeds the threshold. In a special case a node decides to adopt a new product if at least the fraction $\theta(i)$ of its neighbours has done so. In some cases the threshold may also depend on the specific product under consideration. In the cascade models, each node that adopts a product can activate each of his neighbours with a certain probability and each node has only one chance of activating a neighbour[1].

Most of research has focussed on the situation in which the players face the choice of adopting a specific product or not. The algorithmic problem of choosing an initial set of nodes so as to maximize the adoption of a given product were studied initially in [8] and [16]. Certain variants and generalizations of this problem were also studied in several publications that followed, e.g., [7, 11, 20].

When studying social networks from the point of view of adopting new products it is natural to lift the restriction of one product. One natural example of such a situation is when users can adopt one out of several competing products (for example providers of mobile telephones). Then, because of lower subscription costs, each owner of a mobile telephone naturally prefers that his friends choose the same provider. Another example is when children have to choose a secondary school. Here, again, children prefer to choose a school which their friends will choose, as well. Also, in discussions preceding voting in a small institution, for instance for the position of a chairman of a club, preferences announced by some club members may influence the votes cast by their friends.

What is common in these situations is that the number of choices is small in comparison with the number of agents and the outcome of the adoption process does not need to be unique. Indeed, individuals with a low ’threshold’ can adopt

[1]For the case of a single product, and when thresholds are assumed to be random variables, the two models have been proved to be equivalent in the sense that they produce the same distribution on outcomes [16].

any product a small group of their friends adopts. As a result this model leads to different considerations than the ones mentioned above.

Social networks in the presence of multiple products have been studied in a number of recent papers. In the presence of multiple products, diffusion has been investigated recently for the cascade model in [3, 5, 17]. In [17] a special case of the cascade model is studied and NP-hardness results are obtained on finding the best set of influential nodes in the presence of another competing product. In [3] the authors also study a generalization to a cascade model with multiple products and provide approximation algorithms for the problem of maximizing the influence of a product given the initial adopters of the other products. Finally, in [5], the authors provide approximation algorithms for certain variants of the problem with two products.

For threshold models, an extension to two products has been recently proposed in [4], where the authors examine whether the algorithmic approach of [16] can be extended. Algorithms and hardness of approximation results are provided for certain variants of the diffusion process. In line with [16], the authors of [4] also assume that the threshold of each node is a random variable and the goal is to maximize the expected spread.

Game theoretic aspects have also been considered in the case of two products. In particular, the behavior of best response dynamics in infinite graphs is studied in [19], when each node has to choose between two different products. An extension of this model is studied in [14] with a focus on notions of compatibility and bilinguality, i.e., having the option to adopt both products at an extra cost so as to be compatible with all your neighbours.

1.2 Contributions

We study a new model of a social network in which nodes (agents) can choose out of *several* alternatives *and* in which various outcomes of the adoption process are possible. Our model combines a number of features present in various models of networks.

It is a threshold model and we assume that the threshold of a node is a fixed number as in [7] (and unlike [16, 4], where they are random variables). This is in contrast to Hebb's model of learning in networks of neurons, the focus of which is on learning, leading to strengthening of the connections (here thresholds). In our context, the threshold should be viewed as a fixed 'resistance level' of a node to adopt a product. Contrary to the SIR model, see, e.g., [15], in which a node can be in only two states, in our model each node can choose out of several states (products). We also allow that not all nodes have exactly the same set of products to choose from, e.g. due to geographic or income restrictions some products may be available only to a subset of the nodes. If a node changes its state from the initial one, the new state (that corresponds to the adopted product) is final, as is the case with most of the related literature.

Our work consists of two parts. In the first part (Sections 3, 4, 5) we study three basic problems concerning this model. In particular, we find necessary and sufficient conditions for determining whether

- a specific product will possibly be adopted by all nodes.
- a specific product will necessarily be adopted by all nodes.
- the adoption process of the products will yield a unique outcome.

For each of these questions, we obtain a characterization with respect to properties of the underlying graph. Furthermore, our characterizations yield efficient algorithms for solving each problem. We also identify a natural class of social networks that yield a unique outcome.

In the second part (Section 6) we investigate the complexity of various other algorithmic problems concerning the adoption process. We start with the problem of determining whether, given an initial network, a final network exists in which all nodes adopted a product. Then we move on to questions regarding the behaviour of a given node in terms of adopting a given product or some product in some (respectively, all) network(s). We also study the problems of computing the minimum (respectively, maximum) possible spread of a product.

We resolve the complexity of all these problems. Some of them turn out to be efficiently solvable, whereas the remaining ones are either co-NP-complete or have strong inapproximability properties. We also show that some, but not all, of these problems can be solved in polynomial time when there are only two products.

Finally, in Section 7 we explain how one can transform social networks into ones that are in some sense simpler, at the cost of addition of new nodes. These transformations clarify the conciseness hidden in the initial definition and relate it to the one used in [2], in which the threshold functions were product independent.

2 Preliminaries

Assume a fixed weighted directed graph $G = (V, E, w)$ (with no parallel edges and no self-loops), with $n = |V|$ and $w_{ij} \in [0,1]$ being the weight of edge (i, j). In our proposed algorithms we shall assume that we are given the adjacency matrix representation of the graph. Some of our algorithms use the adjacency lists representation, which can be easily obtained from the adjacency matrix in time $O(n^2)$.

Given a node i of G we denote by $N(i)$ the set of nodes from which there is an incoming edge to i. We call each $j \in N(i)$ a ***neighbour*** of i in G. We assume that for each node i such that $N(i) \neq \emptyset$, $\sum_{j \in N(i)} w_{ji} \leq 1$.

Let P be a finite set of alternatives, that we call from now on ***products***. By a ***social network*** (from now on, just ***network***) we mean a tuple (G,P,p,θ), where p assigns to each agent i a non-empty set of products $p(i) \subseteq P$ from which it can make a choice. For $i \in V$ and $t \in p(i)$ the ***threshold function*** θ yields a value $\theta(i,t) \in (0,1]$. The threshold $\theta(i,t)$ should be viewed as agent i's resistance level to adopt product t. In some cases, the threshold function may not depend on the product t. We will call such functions ***product independent*** and we will then use $\theta(i)$ instead of $\theta(i,t)$ to denote the resistance of agent i.

The idea is that each node i is offered a non-empty set $p(i)$ of products from which it can make its choice. If $p(i)$ is a singleton, say $p(i) = \{t\}$, the node adopted the product t. Otherwise it can adopt a product t if the total weight of incoming edges from neighbours that have already adopted t is at least equal to the threshold $\theta(i,t)$. To formalize the problems that we want to study, we need first to introduce a number of notions. Since G,P and θ are fixed, we often identify each network with the function p.

Consider a binary relation $\rightarrow$ on networks. Denote by $\rightarrow^*$ the reflexive, transitive closure of $\rightarrow$. We call a reduction sequence $p \rightarrow^* p'$ ***maximal*** if for no p'' we have $p' \rightarrow p''$. In that case we will say that p' is a ***final*** network, given the initial network p.

Definition 2.1 Assume an initial network p and a network p'. We say that

- p' is ***reachable*** (from p) if $p \rightarrow^* p'$,
- p' is ***unavoidable*** (from p) if for all maximal sequences of reductions $p \rightarrow^* p''$ we have $p' = p''$,
- p admits a ***unique outcome*** if some network is unavoidable from p. □

So a network is reachable if it can be reached by some sequence of $\rightarrow$ reductions that starts with p, and it is unavoidable if it is reachable by a maximal sequence of reductions and a unique outcome of the initial network p exists.

From now on we specialize the relation $\rightarrow$. Given a social network p when $N(i) \neq \emptyset$ we use the abbreviation $A(t,i)$ (for 'adoption condition for product t by node i') for

$$\sum_{j \in N(i) | p(j)=\{t\}} w_{ji} \geq \theta(i,t)$$

and stipulate that $A(t,i)$ holds when $N(i) = \emptyset$.

Definition 2.2

- We write $p_1 \to p_2$ if $p_2 \neq p_1$ and for all nodes i, if $p_2(i) \neq p_1(i)$, then $|p_1(i)| \geq 2$ and for some $t \in p_1(i)$

$$p_2(i) = \{t\} \text{ and } A(t,i) \text{ holds in } p_1.$$

- We say that node i in a network p
  - ***adopted product*** t if $p(i) = \{t\}$,
  - ***can adopt product*** t if

$$t \in p(i) \wedge |p(i)| \geq 2 \wedge A(t,i).$$

$\square$

In particular, a node with no neighbours and more than one available product can adopt any product that is a possible choice for it. Also, we assume that an adoption decision is final. Once a node decides to adopt a product, it cannot cancel its decision or switch later to another product.

So $p_1 \to p_2$ holds if

- any node that adopted a product in p_2 either adopted it in p_1 or could adopt it in p_1,
- at least one node could adopt a product in p_1 and adopted it in p_2,
- the nodes that did not adopt a product in p_2 did not change their product sets.

Note that each modification of the function p results in assigning to a node i a singleton set. So if $p_1 \to^* p_2$, then for all nodes i either $p_2(i) = p_1(i)$ or $p_2(i)$ is a singleton set.

One of the questions we are interested in is whether a product t can spread to the whole network. We will denote this final network by $[t]$, where $[t]$ denotes the constant function p such that $p(i) = \{t\}$ for all nodes i.

Below, given a network (G, P, p, θ) and a product $t \in P$ we denote by $G_{p,t}$ the weighted directed graph obtained from G by removing from it all edges to nodes i with $p(i) = \{t\}$. So in $G_{p,t}$ for all such nodes i we have $N(i) = \emptyset$ and for all other nodes the set of neighbours in $G_{p,t}$ and G is the same.

If each weight w_{ji} in the considered graph equals $\frac{1}{|N(i)|}$, then we call the corresponding network ***equitable***. So in equitable networks the adoption condition $A(t,i)$ holds if at least the fraction $\theta(i,t)$ of the neighbours of i adopted in p product t.

Example 2.3 As an example consider the equitable networks in Figure 1, where $P = \{t_1, t_2\}$ and where we mention next to each node the set of products available to it. We assume in this example that the threshold function does not depend on the product argument, hence we omit it.

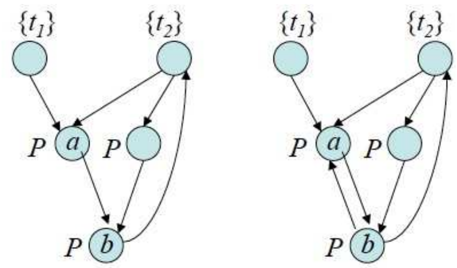

Figure 1: Two examples of social networks

In the first network, if $\theta(a) \leq \frac{1}{2}$, then the network in which each node apart from the one on the top left adopts product t_2 is reachable, though not unavoidable. It is no longer a reachable network if $\theta(a) > \frac{1}{2}$. In that case the initial network admits a unique outcome. In this unique outcome, node b adopts product t_2 if and only if $\theta(b) \leq \frac{1}{2}$.

For the second network the following more elaborate case distinction lists the possible values of p in the final reachable networks.

$$
\begin{array}{ll}
\theta(a) \leq \frac{1}{3} \wedge \theta(b) \leq \frac{1}{2} & : (p(a) = \{t_1\} \vee p(a) = \{t_2\}) \wedge \\
 & \ \ (p(b) = \{t_1\} \vee p(b) = \{t_2\}) \\
\theta(a) \leq \frac{1}{3} \wedge \theta(b) > \frac{1}{2} & : (p(a) = \{t_1\} \wedge p(b) = P) \vee \\
 & \ \ (p(a) = p(b) = \{t_2\}) \\
\frac{1}{3} < \theta(a) \leq \frac{2}{3} \wedge \theta(b) \leq \frac{1}{2} & : p(a) = p(b) = \{t_2\} \\
\frac{1}{3} < \theta(a) \wedge \theta(b) > \frac{1}{2} & : p(a) = p(b) = P \\
\frac{2}{3} < \theta(a) \wedge \theta(b) \leq \frac{1}{2} & : p(a) = P \wedge p(b) = \{t_2\}
\end{array}
$$

In particular, when $\frac{1}{3} < \theta(a) \leq \frac{2}{3}$ and $\theta(b) \leq \frac{1}{2}$, node a adopts the product t_2 only *after* node b adopts it. $\square$

3 Reachable outcomes

We start with providing necessary and sufficient conditions for a product to be reachable by all nodes. This is achieved by a structural characterization of graphs that allow products to spread to the whole graph, given the threshold function θ. In particular, we shall need the following notion.

Definition 3.1 Given a weighted directed graph G, a threshold function θ and a product t, we will say that G is ***(θ,t)-well-structured*** if for some function $level$ that maps nodes to natural numbers, we have that for all nodes i such that $N(i) \neq \emptyset$

$$\sum_{j \in N(i) | level(j) < level(i)} w_{ji} \geq \theta(i,t). \tag{1}$$

In other words, a weighted directed graph is (θ,t)-well-structured if levels can be assigned to its nodes in such a way that for each node i such that $N(i) \neq \emptyset$, the sum of the weights of the incoming edges from lower levels is at least $\theta(i,t)$. We will often refer to the function $level$ as a ***certificate*** for the graph being (θ,t)-well-structured. Note that there can be many certificates for a given graph. Note also that (θ,t)-well structured graphs can have cycles. For instance, it is easy to check that for every product $t \in P$ the second network in Figure 1 is (θ,t)-well structured when $\theta(i) \leq \frac{1}{3}$ for every node i.

We have the following characterization.

Theorem 3.2 *Assume a network (G,P,p,θ) and a product $top \in P$. The network $(G,P,[top],\theta)$ is reachable from (G,P,p,θ) iff*

- *for all i, $top \in p(i)$,*
- *$G_{p,top}$ is (θ,top)-well-structured.*

Proof.
$(\Rightarrow)$ If for some node i we have $top \notin p(i)$, then i cannot adopt product top and $[top]$ is not reachable.

To establish the second condition consider a reduction sequence

$$p_1 \to p_2 \to \ldots \to p_m$$

starting in p and such that $p_m = [top]$.

Assign now to each node i the minimal k such that $p_{k+1}(i) = \{top\}$. We claim that this definition of the $level$ function shows that $G_{p,top}$ is (θ,top)-well-structured. Consider a node i.

Case 1. $level(i) = 0$.

Then $p(i) = \{top\}$, so by the definition of $G_{p,top}$ we have $N(i) = \emptyset$ in $G_{p,top}$. Hence we do not need to argue about these nodes since we only need to ensure condition (1) for nodes with $N(i) \neq \emptyset$.

Case 2. $level(i) > 0$.

Suppose that $N(i) \neq \emptyset$ and that $level(i) = k$. By the definition of the reduction $\rightarrow$ the adoption condition $A(top, i)$ holds in p_k, i.e.,

$$\sum_{j \in N(i) | p_k(j) = \{top\}} w_{ji} \geq \theta(i, top).$$

But for each $j \in N(i)$ such that $p_k(j) = \{top\}$ we have by definition $level(j) < level(i)$. So (1) holds.

$(\Leftarrow)$ Consider a certificate function *level* showing that $G_{p,top}$ is (θ, top)-well-structured. Without loss of generality we can assume that the nodes in $G_{p,top}$ such that $N(i) = \emptyset$ are exactly the nodes of level 0. We construct by induction on the level m a reduction sequence

$$p \rightarrow^* p'',$$

such that for all nodes i we have $top \in p''(i)$ and for all nodes i of level $\leq m$ we have $p''(i) = \{top\}$.

Consider level 0. By definition of $G_{p,top}$, a node i is of level 0 iff it has no neighbours in G or $p(i) = \{top\}$. In the former case, by the first condition, $top \in p(i)$. So $p \rightarrow^* p''$, where the function p'' is defined by

$$p''(i) := \begin{cases} \{top\} & \text{if } level(i) = 0 \\ p(i) & \text{otherwise} \end{cases}$$

This establishes the induction basis.

Suppose the claim holds for some level m. So we have $p \rightarrow^* p'$, where for all nodes i we have $top \in p'(i)$ and for all nodes i of level $\leq m$ we have $p'(i) = \{top\}$.

Consider the nodes of level $m+1$. For each such node i we have $top \in p'(i)$, $N(i) \neq \emptyset$ and

$$\sum_{j \in N(i) | level(j) < level(i)} w_{ji} \geq \theta(i, top).$$

By the definition of $G_{p,top}$ the sets of neighbours of i in G and $G_{p,top}$ are the same. By the induction hypothesis for all nodes j such that $level(j) < level(i)$ we have $p'(j) = \{top\}$.

So either such a node i adopted product top in p' or can adopt product top in p'.

Hence $p' \rightarrow^* p''$, where the function p'' is defined by

$$p''(i) := \begin{cases} \{top\} & \text{if } level(i) = m+1 \\ p'(i) & \text{otherwise} \end{cases}$$

Consequently $p \to^* p''$, which establishes the induction step.

By induction we conclude $p \to^* [top]$. $\square$

Next we show that testing if a graph is (θ,t)-well-structured can be efficiently solved.

Theorem 3.3 *Given a weighted directed graph G, a threshold function θ and a product t, we can decide whether G is (θ,t)-well-structured in time $O(n^2)$.*

Proof. We claim that the following simple algorithm achieves this:

- Given a weighted directed graph G, first assign level 0 to all nodes with $N(i) = \emptyset$. If no such node exists, output that the graph is not (θ,t)-well-structured.
- Inductively, at step i, assign level i to each node for which condition (1) from Definition 3.1 is satisfied when considering only its neighbours that have been assigned levels $0,\ldots,i-1$.
- If by iterating this all nodes are assigned a level, then output that the graph is (θ,t)-well-structured. Otherwise, output that G is not (θ,t)-well-structured.

The above algorithm can be implemented in time $O(n^2 + |E|) = O(n^2)$. We can first create the adjacency list representation so that for each node we have a list with its outgoing edges. Given this representation, we can implement the steps of the algorithm in $O(|E|)$ time. The idea is that each edge of the graph is processed only once and only a constant number of operations is needed for every edge. Indeed, one can keep a counter for every node that sums up the weight from nodes that have already been assigned a level. For every node that was assigned a level at the previous round, one can go through its outgoing edges and update the corresponding counters accordingly (only counters of nodes that have not yet been assigned a level are updated). Hence we can assign a level number to any node whose counter has been updated at the current round and has exceeded the threshold.

To prove the correctness of the algorithm, note that if the input graph is not (θ,t)-well-structured, then the algorithm will output No, as otherwise, at termination it would have constructed a level function for a non-(θ,t)-well-structured graph. Hence it remains to prove that if a graph is (θ,t)-well-structured, the algorithm will output Yes.

Suppose a graph G is (θ,t)-well-structured. We will use a certificate function, l_G, in which all nodes are assigned the minimum possible level. For each node i, let l^i be a certificate function where node i has the minimum possible level. Then define $l_G(i) := \min_j l^j(i) = l^i(i)$.

First note that l_G is a certificate function because a minimum of certificate functions is also a certificate for G. By the definition of l_G, the level of each node i cannot be lowered below $l_G(i)$, i.e., for all nodes i

$$l_G(i) = \min\{k : \sum_{j \in N(i) | l_G(j) < k} w_{ji} \geq \theta(i,t)\}. \tag{2}$$

We now prove that every node is assigned a level by the algorithm and in particular that l_G is the level function *level* generated by the algorithm, hence the algorithm outputs Yes. For level 0, note that by the minimality of l_G and since $\theta(i,t) > 0$ for every i, the only nodes for which l_G assigns 0 are all nodes i such that $N(i) = \emptyset$. But these are precisely the nodes that are assigned level 0 by the algorithm as well.

Suppose by induction that l_G and *level* coincide on all nodes considered by the algorithm in steps $1,\ldots,k-1$, where k is a level used by l_G. Then by the construction of the algorithm and by (2), the algorithm assigns level k to *all* nodes i such that $l_G(i) = k$. Moreover, since k is used by l_G, some new nodes are assigned a level at step k.

Hence, l_G and *level* coincide. Consequently the algorithm assigns a level to all nodes and hence outputs Yes. $\square$

Note 3.4 *The above algorithm can run in time $O(|E|)$ when we are given directly the representation of the graph in terms of adjacency lists of outgoing edges for each node instead of the adjacency matrix.*

Finally, we end this section by observing that the algorithmic question of determining whether a network $[top]$ is reachable can be solved efficiently.

Theorem 3.5 *Assume a network (G,P,p,θ) and a product $top \in P$. There is an algorithm running in time $O(n^2)$ that determines whether the network $(G,P,[top],\theta)$ is reachable.*

Proof. The proof follows either by using Theorem 3.2 and Theorem 3.3 for $G_{p,top}$ or by simply start performing adoptions only of product top until no further reduction is possible. $\square$

4 Unavoidable outcomes

Next, we focus on the notion of unavoidable outcomes. We establish the following characterization.

Theorem 4.1 *Assume a network (G,P,p,θ) and a product $top \in P$. A network $(G,P,[top],\theta)$ is unavoidable iff*

- *for all i, if $N(i) = \emptyset$, then $p(i) = \{top\}$,*
- *for all i, $top \in p(i)$,*
- *$G_{p,top}$ is (θ, top)-well-structured.*

To prove this, we shall need first a few lemmas.

Lemma 4.2 *Suppose that $p \to^* p'$ and for some node i we have $p'(i) = \{t\}$. Then for some node j such that $N(j) = \emptyset$ or $p(j)$ is a singleton, we have $t \in p(j)$.*

Intuitively, this means that each product eventually adopted can also be initially adopted (by a possibly different node).

Proof. Let $p \to^* p'$ be of the form

$$p_1 \to p_2 \to \ldots \to p_m.$$

Let l' be the smallest index l such that for some node j we have $p_l(j) = \{t\}$. If $l' = 1$, then $p(j) = \{t\}$ and we are done. If $l' > 1$ then by the choice of l' we have $N(j) = \emptyset$, as otherwise for some node k we would have $p_{l'-1}(k) = \{t\}$. Moreover, $t \in p_{l'-1}(j)$ implies $t \in p_1(j)$, that is $t \in p(j)$. $\Box$

Lemma 4.3 *Assume a network (G,P,p,θ) and a product $top \in P$. Suppose that*

- *for all i, if $N(i) = \emptyset$ or $p(i)$ is a singleton, then $p(i) = \{top\}$.*

Then (G,P,p,θ) admits a unique outcome.

Intuitively, this means that if initially only one product can be adopted, then a unique outcome of the network exists.

Proof. Consider two maximal sequences of reductions $p \to^* p'$ and $p \to^* p''$. Let $p \to^* p'$ be of the form

$$p_1 \to p_2 \to \ldots \to p_m.$$

We prove by induction on k that for all nodes i and products t if $p_k(i) = \{t\}$, then $p''(i) = \{t\}$. If $p(i) = \{t\}$, then also $p''(i) = \{t\}$. This takes care of the induction basis.

Assume the claim holds for some k and suppose $p_{k+1}(i) = \{t\}$. If $p_k(i) = \{t\}$, then by the induction hypothesis $p''(i) = \{t\}$. Otherwise by the definition of the $\to$ relation $t \in p_k(i)$, $N(i) \neq \emptyset$ and $A(t,i)$ holds in p_k.

By the assumption and Lemma 4.2 $t = top$. By the induction hypothesis $A(t,i)$ holds in p''.

Moreover, $p_{k+1}(i) = \{top\}$ implies $top \in p(i)$. By the assumption and Lemma 4.2 for no $t' \neq top$ we have $p''(i) = \{t'\}$. Hence $p'' \rightarrow^* p^\#$, where $p^\#(i) = \{top\}$. But $p \rightarrow^* p''$ is a maximal sequence of reductions, so $p^\# = p''$ and consequently $p''(i) = \{top\}$.

We conclude by induction that for all nodes i if $p'(i) = \{t\}$, then $p''(i) = \{t\}$. By symmetry the reverse implication holds. By the definition of the $\rightarrow$ relation this implies that $p' = p''$. □

Proof of Theorem 4.1

$(\Rightarrow)$ If $[top]$ is unavoidable, then it is reachable from p, hence, thanks to Theorem 3.2 we only need to establish the first condition. But if for some node i such that $N(i) = \emptyset$ we have $p(i) \neq \{top\}$, then i can adopt a different product than top and $[top]$ cannot be unavoidable.

$(\Leftarrow)$ By Theorem 3.2 $[top]$ is reachable, so we only need to show that it is a unique outcome. But this is guaranteed by Lemma 4.3. □

In analogy to Theorem 3.5, we also have the following simple fact.

Theorem 4.4 *Assume a network (G, P, p, θ) and a product $top \in P$. There is an algorithm, running in time $O(n^2)$, that determines whether the network $(G, P, [top], \theta)$ is unavoidable.*

5 Unique outcomes

Finally, we consider the question of when a network admits a unique outcome. To answer this, we introduce the following definitions.

Definition 5.1 Given networks p, p' based on the same graph we say that

- node i ***can switch in*** p' ***given*** p if i adopted in p' a product t and for some $t' \neq t$

  $$t' \in p(i) \wedge A(t', i) \text{ holds in } p',$$

- p' is ***ambivalent given*** p if it contains a node that either can adopt more than one product or can switch in p' given p,

- the reduction $p \to p'$ is ***fast*** if for each node i, if i can adopt a product in p then i adopted a product in p'. Intuitively, $p \to p'$ is then a 'maximal' one-step reduction of p. $\square$

Definition 5.2 By the ***contraction sequence*** of a network we mean the unique reduction sequence $p \to^* p'$ such that

- each of its reduction steps is fast,
- either $p \to^* p'$ is maximal or p' is the first network in the sequence $p \to^* p'$ that is ambivalent given p. $\square$

We now formulate a characterization of networks that admit a unique outcome.

Theorem 5.3 *A network admits a unique outcome iff its contraction sequence ends in a non-ambivalent network.*

Proof.
$(\Rightarrow)$ Suppose that a network p admits a unique outcome and assume by contradiction that the contraction sequence χ of p ends in an ambivalent network p'. If a node in p' can adopt two different products, then we get a contradiction. Otherwise a node i' in p' can switch from a product t to a product $t' \neq t$.

Given a reduction sequence ξ that starts in p and a node i that adopted in it a product t, but not initially (so not in p), we define a modified reduction sequence in which this node can adopt a product but did not adopt any. This is done so as to cancel all adoptions that led i' to adopt t. To this end we set $p''(j) := p(j)$ for every node j that adopted product t and every network p'' from ξ and subsequently remove from the resulting sequence the duplicate networks.

Since node i' can switch from t to t', we have $\{t, t'\} \subseteq p(i')$, so on χ node i' did not adopt the product t initially. So the corresponding modification of χ w.r.t. node i' results in a reduction sequence that starts in p and in which node i' can adopt product t'. So p admits two outcomes which yields a contradiction.

$(\Leftarrow)$ First, given a maximal reduction sequence $\xi := p \to^* p'$ we define its ***fast run*** inductively by its length as follows. If $p = p'$, then p is the fast run of $p \to^* p'$. Otherwise, $\xi = p \to p_1 \to^* p'$ for some network p_1. Define a social network p_2 as follows:

$$p_2(i) := \begin{cases} \{t\} & \text{if } i \text{ can adopt } t \text{ in } p \text{ and } p'(i) = \{t\} \\ p(i) & \text{otherwise} \end{cases}$$

We have then $p \to p_2$ and $p_2 \to^* p'$. We define then the fast run of $p \to^* p'$ as the concatenation of $p \to p_2$ and the fast run of $p_2 \to^* p'$.

Intuitively, a fast run of a maximal reduction sequence $p \to^* p'$ yields the same final result, p', but by maximizing at each reduction step the number of nodes that adopt a product.

Suppose now that the contraction sequence of a network p ends in a non-ambivalent network and assume by contradiction that p admits two outcomes. So two sequences of reductions ξ and ξ' exist that both start in p, are maximal, and their final elements differ.

Let $fr(\xi)$ and $fr(\xi')$ be the respective fast runs of ξ and ξ'. By assumption at least one of these two fast runs, say $fr(\xi)$, differs from the contraction sequence χ of p. Let p'_1 be the first network in the sequence χ in which a difference with $fr(\xi)$ arises.

By assumption p'_1 is non-ambivalent, so some fast reduction $p'_1 \to p'$ is part of χ and a reduction $p'_1 \to p''$ with $p' \neq p''$ is part of $fr(\xi)$. Since $p'_1 \to p'$ is a fast reduction and $fr(\xi)$ is a fast run, the difference between p' and p'' arises due to the fact that some node i adopted in p' one product and in p'' a different product. But this means that p'_1 is ambivalent, which is a contradiction. $\square$

It would be interesting to find a structural characterization of networks that admit a unique outcome, as Theorem 5.3 only provides such a characterization in terms of the contraction sequences. At this stage we only have the following result.

Corollary 5.4 *Assume a network (G,P,p,θ) such that*

- *for all nodes i and products t we have $\theta(i,t) > \frac{1}{2}$,*
- *for all i, if $N(i) = \emptyset$, then $p(i)$ is a singleton.*

Then (G,P,p,θ) admits a unique outcome.

Proof. It suffices to note that if $p \to^* p'$, then p' is not ambivalent given p. So the result is a direct consequence of Theorem 5.3. $\square$

It is easy to see that in the above corollary the condition that for all nodes and products i we have $\theta(i,t) > \frac{1}{2}$ cannot be dropped. Indeed, consider the equitable network depicted in Figure 2 and assume that for all products t, $\theta(c,t) \leq \frac{1}{2}$. Then node c can adopt both product t_1 and product t_2, so no unique outcome exists.

On the other hand, the above corollary can be strengthened by assuming that the network is such that if for some product t we have $\theta(i,t) \leq \frac{1}{2}$, then $|N(i)| < 2$ or $|p(i)| = 1$. The reason is that the nodes for which $|N(i)| < 2$ or $|p(i)| = 1$ cannot introduce an ambivalence.

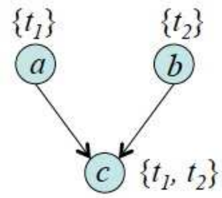

Figure 2: Social network with two different outcomes

When for some node i and product t, $\theta(i,t) \leq \frac{1}{2}$ holds and neither $|N(i)| < 2$ nor $|p(i)| = 1$, the equitable network still may admit a unique outcome but it does not have to. For instance the second network in Figure 1 admits a unique outcome for the last three alternatives (explained in Example 2.3), while for the first two it does not.

Theorem 5.3 does yield a simple algorithm for testing whether a network has a unique outcome.

Theorem 5.5 *There exists a polynomial time algorithm, running in time $O(n^2 + n|P|)$, that determines whether a network admits a unique outcome.*

For all practical purposes we have $|P| << n$, so the running time is in practice $O(n^2)$.

Proof of Theorem 5.5.

By Theorem 5.3 it suffices to determine whether the contraction sequence of a network p ends in a non-ambivalent social network. This can be tested using the algorithm presented in Figure 3. The algorithm keeps performing fast reductions until we realize that either a node can adopt two different products or can switch from one product to another. If none of these happens then we can safely conclude given Theorem 5.3 that the network has a unique outcome.

Given a network (G, P, p, θ), the algorithm uses for each node j and each product $t \in p(j)$ a counter $S_{j,t}$. The counter measures the accumulated weight from incoming edges that have already adopted a product t.

Regarding the complexity of the algorithm, the initial part of producing the required representation in Line 1 may take time up to $O(n^2)$ if we are given the matrix representation or any other of the standard ways of representing a graph. The initialization of the counters $S_{j,t}$ requires in the worst case $O(n|P|)$. As for the remaining part, the variable L maintains the set of nodes that adopted a product in the last round (Lines 11 and 28). Each edge (i, j) is examined exactly once, just after i adopts a product.

The number of operations that we need to perform for every edge is $O(1)$ because we only need to update the appropriate counter $S_{j,t}$ and add j to the list R (Lines 16-17). Furthermore, we also need to check for each such j whether it can adopt more than one product. This can also be done while we update each counter $S_{j,t}$ by having another counter that increases by one for every $S_{j,t}$ that exceeds the threshold $\theta(j,t)$. In total, we do not need more than $O(1)$ operations per edge and therefore the total running time is $O(n^2 + n|P| + |E|) = O(n^2 + n|P|)$. $\square$

Finally, we note that for the class of networks of Corollary 5.4 we can have an even simpler algorithm, removing the dependency on $|P|$.

Theorem 5.6 *There exists an algorithm, running in time $O(n^2)$, that determines whether a network, such that for all nodes i and products t we have $\theta(i,t) > \frac{1}{2}$, admits a unique outcome.*

6 Product adoption

In this section we study a number of questions concerning adoption of the products by the nodes of a given network, focusing on complexity matters. Recall that given an initial network p, a final network is one that has been obtained from p by a maximal sequence of reductions. We first clarify the complexity of the following problem.

FINAL: Given an initial network determine whether a final network exists in which all nodes adopted some product.

Theorem 6.1 ***FINAL** is NP-complete, even for 2 products and product independent thresholds.*

Proof. First we prove that **FINAL** is in NP. Given an initial network, the certificate can consist of a final network in which every node adopted some product along with the series of reductions that led to this final network (there can be at most $O(n)$ such reductions). One can then check in polynomial time that this is a valid final network, given the initial network, and that indeed all nodes have adopted a product.

For NP-hardness, we give a reduction from the NP-complete PARTITION problem, which is: given n positive rational numbers $(a_1, \ldots, a_n)$, is there a set S such that $\sum_{i \in S} a_i = \sum_{i \notin S} a_i$? Consider an instance I of PARTITION. Without loss of generality, suppose we have normalized the numbers so that $\sum_{i=1}^{n} a_i = \frac{1}{2}$. Hence the question is to decide whether there is a set S such that $\sum_{i \in S} a_i = \sum_{i \notin S} a_i = \frac{1}{4}$.

Produce the representation with a list of outgoing edges for each node;
for $i \in V$ **do**
  set $p(i)$ to be the initial list of products available to node i
end for
for $j \in V, t \in p(j)$ **do**
  $S_{j,t} := 0$;// counts total weight to j from nodes that adopted t;
end for
if $\exists i \in V$ with $N(i) = \emptyset$ and $|p(i)| \geq 2$ **then**
  return ”No unique outcome”;
end if
$L := \{i \in V : |p(i)| = 1\}$;// initialize L to a list of nodes that already have adopted a product;
if $L = \emptyset$ **return** ”Unique outcome” **endif**;
while $L \neq \emptyset$ **do**
  $R := \emptyset$;
  for $i \in L$ and j such that $(i, j) \in E$ **do**
    if i has adopted t and $t \in p(j)$ **then** $S_{j,t} := S_{j,t} + w_{ij}$ **end if**;
    $R := R \cup \{j\}$; // nodes we need to check for ambivalence
  end for
  for $j \in R$ **do**
    Compute $|\{t : S_{j,t} \geq \theta(j,t)\}|$; //even for nodes that have already adopted a product
    if $|\{t : S_{j,t} \geq \theta(j,t)\}| \geq 2$ **return** ”No unique outcome” **endif**;
    if $|\{t : S_{j,t} \geq \theta(j,t)\}| = 1$ and j has not yet adopted t **then**
      node j adopts product t;
    else
      $R := R \setminus \{j\}$; // j does not adopt any product;
    end if
  end for
  $L := R$ // put in L all nodes that adopted a product in last round
end while
return ”Unique outcome” // No further reduction is possible

Figure 3: Pseudocode for the algorithm of Theorem 5.5

We build an instance of our problem with two products, namely $P = \{t_1, t_2\}$, and with the network shown in Figure 4. The threshold function does not depend on the product argument (that is omitted) and is given by: $\theta(a) = \theta(b) = \frac{3}{4}$. Finally, for each node $i \in \{1, \ldots, n\}$, we set $w_{ia} = w_{ib} = a_i$. The weights of the other two edges are $\frac{1}{2}$.

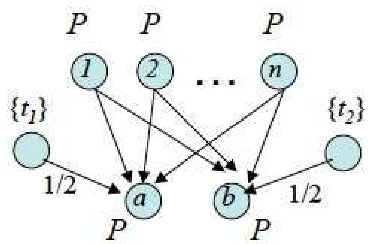

Figure 4: Social network related to the **FINAL** problem, with $P = \{t_1, t_2\}$.

Suppose there is a solution S to I. Then we can have the nodes corresponding to the set S adopt t_1 and the remaining nodes from $\{1, \ldots, n\}$ adopt t_2. By the choice of the weights w_{ia} and w_{ib} this implies that node a can adopt t_1 and node b can adopt t_2. Hence a final network exists in which all nodes adopted a product.

For the reverse direction, suppose that a final network exists in which all nodes adopted a product. Then node a adopted product t_1 and node b adopted product t_2, as it is not possible for node a to adopt t_2 and for node b to adopt t_1. Let S be the set of nodes $i \in \{1, \ldots, n\}$ that adopted product t_1. Then by the choice of the weights w_{ia} and w_{ib} and the thresholds of the nodes a and b, it holds that both $\sum_{i \in S} a_i \geq \frac{1}{4}$ and $\sum_{i \notin S} a_i \geq \frac{1}{4}$. But since $\sum_{i=1}^{n} a_i = \frac{1}{2}$, this implies that $\sum_{i \in S} a_i = \sum_{i \notin S} a_i = \frac{1}{4}$, i.e., there is a solution to the instance I of the PARTITION problem. $\square$

We now move on to a different class of problems, motivated by the results of Section 5, which reveal that many networks will not admit a unique outcome. Therefore the following questions concerning product adoption by a given node are of natural interest for such networks.

ADOPTION 1: (unavoidable adoption of some product)
Determine whether a given node has to adopt some product in all final networks.
ADOPTION 2: (unavoidable adoption of a given product)
Determine whether a given node has to adopt a given product in all final networks.
ADOPTION 3: (possible adoption of some product)
Determine whether a given node adopted some product in some final network.
ADOPTION 4: (possible adoption of a given product)

Determine whether a given node adopted a given product in some final network.

Below we resolve the complexity of all these problems.

Theorem 6.2 *The complexity of the above problems is as follows:*

(i) ***ADOPTION 1*** *is co-NP-complete, even for 2 products and product independent thresholds.*

(ii) ***ADOPTION 2*** *for 2 products can be solved in $O(n^2)$ time.*

(iii) ***ADOPTION 2*** *is co-NP-complete for at least* 3 *products, even with product independent thresholds.*

(iv) ***ADOPTION 3*** *can be solved in $O(n^2|P|)$ time.*

(v) ***ADOPTION 4*** *can be solved in $O(n^2)$ time.*

Proof.
(i) It suffices to prove NP-completeness of the complementary problem, which is: given an initial network determine if there is a final network in which a given node does not adopt any product. The argument for the membership in NP is very similar to the membership proof in Theorem 6.1.

To prove NP-hardness, we use again a reduction from the PARTITION problem but with a different normalization for the PARTITION instance. In particular, we assume an instance I with the numbers $a_1,...,a_n$ satisfying $\sum_{i=1}^{n} a_i = 1$. Hence the question is to decide whether there is a set S such that $\sum_{i \in S} a_i = \sum_{i \notin S} a_i = \frac{1}{2}$. We also use a slightly different network than the one depicted in Figure 4. Given the instance I we use the network presented in Figure 5. This network depicts an instance of our problem with $P = \{t_1, t_2\}$ and with node c as the "designated" node. The threshold function does not depend on the product argument (that is omitted) and is given by: $\theta(a) = \theta(b) = \frac{1}{2}$, and $\theta(c) = 1$. Finally, as in the proof of Theorem 6.1, for each node $i \in \{1,\ldots,n\}$, we set $w_{ia} = w_{ib} = a_i$. We also use the weights $w_{ac} = w_{bc} = \frac{1}{2}$.

Suppose now that there is a solution S to the PARTITION instance. Then we can have the nodes corresponding to the set S adopt t_1 and the remaining nodes from $\{1,\ldots,n\}$ adopt t_2. By the choice of the weights w_{ia} and w_{ib} and the thresholds of a and b, this implies that node a can adopt t_1 and node b can adopt t_2. In that case node c cannot adopt any product. Thus a final network exists in which node c does not adopt any product.

Suppose now that in a final network node c did not adopt any product. Then it cannot be the case that nodes a and b adopted the same product since then node c would have adopted it as well. Note also that in all final networks nodes a and b

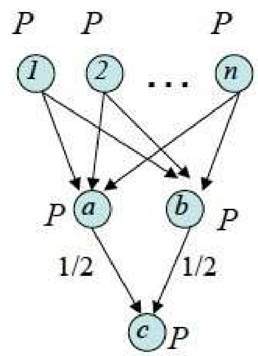

Figure 5: Social network related to the **ADOPTION 1** problem. Here $P = \{t_1, t_2\}$.

have adopted some product. Suppose without loss of generality that node a adopted t_1 and node b adopted t_2. Let S be the set of nodes $i \in \{1,\ldots,n\}$ that adopted t_1. Then the nodes $i \in \{1,\ldots,n\} \setminus S$ adopted t_2. By the choice of the weights we have both $\sum_{i\in S} a_i \geq \frac{1}{2}$ and $\sum_{i\notin S} a_i \geq \frac{1}{2}$. But since $\sum_{i=1}^{n} a_i = 1$, this implies that $\sum_{i\in S} a_i = \sum_{i\notin S} a_i = \frac{1}{2}$, i.e., there is a solution to the instance I of the PARTITION problem.

(ii) The algorithm resembles the one used in the proof of Theorem 5.5. Let $P = \{t_1, t_2\}$, and suppose the given product is t_1. We use the following observation. To determine whether a given node has to adopt t_1 in all final networks, it suffices to check this for the worst possible final network with respect to adoption of t_1. So we first perform fast reductions only for product t_2. Once no further adoption of t_2 is possible, we perform all possible adoptions of t_1 so as to reach a final network. If in this final network, the given node has not adopted t_1, the answer to **ADOPTION 2** is No and otherwise the answer is Yes.

(iii) We provide a reduction from PARTITION but with a slightly more involved network than in the proof of (i). Note that again, it suffices to prove NP-completeness of the complementary problem, which is: given an initial network determine if there is a final network in which a given node does not adopt the given product. The argument for the membership in NP is straighforward.

To prove NP-hardness, we start again with a PARTITION instance I with the numbers $a_1, \ldots, a_n$ satisfying $\sum_{i=1}^{n} a_i = 1$. From this we construct the network shown in Figure 6. This network has 3 products, $P = \{t_1, t_2, t_3\}$, the designated node is e and the designated product is t_3. The weights in the first layer of the graph are as in (i). The rest of the weights are shown in the figure. The product independent threshold function is given by $\theta(a) = \theta(b) = \theta(c) = \theta(d) = \frac{1}{2}$, $\theta(e) = 1/2 + \varepsilon$, for some $\varepsilon > 0$.

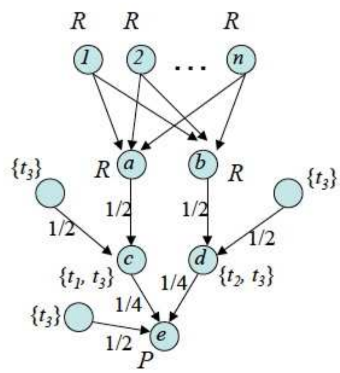

Figure 6: Social network related to the **ADOPTION 2** problem. We fix $R = \{t_1, t_2\}$.

Suppose there is a solution S to I. Then we can have the nodes corresponding to the set S adopt t_1 and the remaining nodes from $\{1, \ldots, n\}$ adopt t_2. Then node a can adopt t_1 and node b can adopt t_2. Subsequently, node c can adopt t_1 and node d can adopt t_2. This yields a final network in which node e does not adopt product t_3.

Conversely, suppose that in a final network node e did not adopt product t_3. Then neither node c nor node d adopted t_3. Hence node c adopted t_1 and node d adopted t_2 and consequently node a adopted t_1 and node b adopted t_2. As in the proof of part (i) this implies that there is a solution to the instance I of the PARTITION problem.

(v) The algorithm resembles the one used in the proof of Theorem 5.5. Given a product, say t, it suffices to start with the nodes that have already adopted t, perform fast reductions but only with respect to t until no further adoption of t is possible, and check if the given node has adopted t.

(iv) Run the algorithm used in (v) for each product. $\square$

It is interesting to observe the separation between **ADOPTION 1** and **ADOPTION 2** for $|P| = 2$. While for $|P| \geq 3$ both problems are co-NP-complete and the proofs are based on similar arguments, in the case that $|P| = 2$, **ADOPTION 2** becomes efficiently solvable but **ADOPTION 1** remains co-NP-complete.

We conclude our study by the following two optimization problems. Suppose that a given product top is neither reachable by all nodes nor unavoidable for all nodes. We would like then to estimate what is the worst and best-case scenario for the spread of this product. That is, starting from a given initial network p, what is the minimum (resp. maximum) number of nodes that will adopt this product in a final network. Hence, the following two problems are of interest.

MIN-ADOPTION: Given an initial network and a product top, what is the minimum number of nodes that adopted top in a final network.
MAX-ADOPTION: Given an initial network and a product top, what is the maximum number of nodes that adopted top in a final network.

We show that these two problems are substantially different, the first being essentially inapproximable, while the second being efficiently solvable.

Theorem 6.3 *Suppose n is the number of nodes of a network.*

(i) ***MAX-ADOPTION*** *can be solved in* $O(n^2)$ *time.*

(ii) ***MIN-ADOPTION*** *for* 2 *products can be solved in* $O(n^2)$ *time.*

(iii) *For at least* 3 *products and even with product independent thresholds, it is NP-hard to approximate* ***MIN-ADOPTION*** *with an approximation ratio better than* $\Omega(n)$.

Proof.
(i) The algorithm is analogous to the one used when analyzing the **ADOPTION 4** problem in the proof of Theorem 6.2. Given a product t, we start with the nodes that have already adopted the product and perform fast reductions but only with respect to t until no further adoption of t is possible.

(ii) Suppose $P = \{t_1, t_2\}$ and that t_1 is the designated product. We first solve the **MAX-ADOPTION** problem for product t_2 and then perform any necessary adoptions of t_1 to reach a final network. This yields a final network with the minimum number of adoptions for product t_1.

(iii) We again give a reduction from PARTITION, though the appropriate network is now more involved. Consider an instance I of PARTITION problem, so n positive rational numbers $(a_1, \ldots, a_n)$ such that $\sum_{i=1}^{n} a_i = 1$. We build an instance of our problem with 3 products, namely $P = \{t_1, t_2, t_3\}$, and with the network shown in Figure 7. Note that this is derived by adding to the network of Figure 6 a chain of M nodes starting from node e. We take M to be $n^{O(1)}$ so that the reduction is of polynomial time. The weight of each edge in the chain is set to 1.

We consider t_3 as the designated product. The threshold function does not depend on the product argument (that is omitted) and is given by: $\theta(a) = \theta(b) = \theta(c) = \theta(d) = \frac{1}{2}$, $\theta(e) = 1/2 + \varepsilon$, for some $\varepsilon > 0$ and for the nodes to the right of node e we can set the thresholds to an arbitrary positive number in $(0,1]$. Finally, for each node $i \in \{1,\ldots,n\}$, we set $w_{ia} = w_{ib} = a_i$. The weights of the other edges can be seen in the figure.

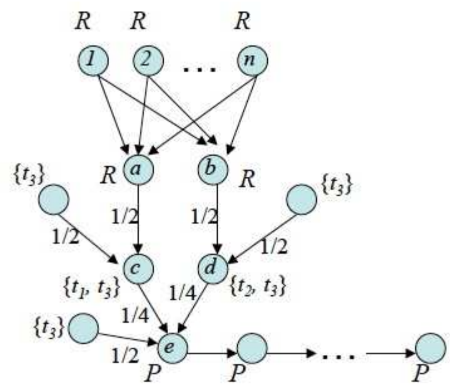

Figure 7: The graph of the reduction with $P = \{t_1,t_2,t_3\}$ and $R = \{t_1,t_2\}$.

We claim that if there exists a solution to the instance I, then a final network exists with the number of nodes that adopted t_3 equal to 3, and otherwise in all final networks the number of nodes that adopted t_3 equals $M+5$. This claim directly yields the desired result, since $M = \Omega(|V|)$.

Suppose there is a solution S to I. As in the proof of Theorem 6.2(iii) it follows that there exists a final network in which node e did not adopt product t_3. Hence a final network exists in which only 3 nodes adopted t_3.

For the reverse direction, suppose there is no solution to the PARTITION problem. This means that no matter how we partition the nodes $\{1,\ldots,n\}$, into two sets S,S', it will always be that for one of them, say S, we have $\sum_{i\in S} a_i > \frac{1}{2}$, whereas for the other we have $\sum_{i\in S'} a_i < \frac{1}{2}$. Thus in each final network, no matter which nodes from $\{1,\ldots,n\}$ adopted t_1 or t_2, the nodes a and b adopted the same product. Suppose for example that nodes a and b both adopted t_1 (the same reasoning applies if they both adopted t_2). This in turn implies that node c adopted t_1 and node d did not adopt t_2. Thus, the only possibility for node d is to adopt t_3. But then the only choice for node e is to adopt t_3 and this propagates along the chain starting from node e. This completes the proof. $\square$

7 Structural results

In [2] we used a slightly more restricted model of a social network in that the threshold functions were product independent. We clarify here the relation between these two models by presenting two transformations of the social networks here considered to social networks with product independent threshold functions and by explaining in which sense they are related.

The first transformation takes as input an arbitrary social network $\mathscr{S} := (G, P, p, \theta)$ and produces an equitable social network with threshold functions that do not depend on the product argument.

First we add a new product t_0 to P. Then for each node i such that $N(i) \neq \emptyset$ and $|p(i)| \geq 2$ we remove the edges $j \to i$ for each node $j \in N(i)$ and perform the following steps for each product $t \in p(i)$ and each minimal subset S of $N(i)$ such that $\sum_{j \in S | p(j) = \{t\}} w_{ji} \geq \theta(i,t)$:

- add a new node $a_{S,t,i}$,
- put $p(a_{S,t,i}) := \{t, t_0\}$ and $\theta(a_{S,t,i}, t) := 1$,
- add the edges $j \to a_{S,t,i}$ for each node $j \in S$,
- add the edge $a_{S,t,i} \to i$,
- put $\theta(i,t) := \frac{1}{2^{|N(i)|}|p(i)|}$.

Call the resulting equitable network $\mathscr{S}'$. The following result relates the networks $\mathscr{S}$ and $\mathscr{S}'$.

Theorem 7.1 *Consider the networks $\mathscr{S}$ and $\mathscr{S}'$.*

(i) If $\mathscr{S} \to^* p_0$ for a final network p_0 given $\mathscr{S}$, then for an extension p'_0 of p_0 we have $\mathscr{S}' \to^* p'_0$, where p'_0 is a final network given $\mathscr{S}'$.

(ii) If $\mathscr{S}' \to^* p'_0$ for a final network p'_0 given $\mathscr{S}'$, then for a restriction p_0 of p'_0 we have $\mathscr{S} \to^* p_0$, where p_0 is a final network given $\mathscr{S}$.

Proof. (i) Let i be the first node belonging to $\mathscr{S}$ and such that $N(i) \neq \emptyset$, $|p(i)| \geq 2$ and i adopted a product t in the reduction sequence $\mathscr{S} \to^* p_0$. So for some subset S of $N(i)$ we have $\sum_{j \in S | p(j) = \{t\}} w_{ji} \geq \theta(i,t)$. Choose a minimal subset S with this property. Then by the definition of the threshold functions in the network $\mathscr{S}'$ node $a_{S,t,i}$ can adopt product t and subsequently node i can adopt t, as well. Repeating this procedure we obtain the desired extension p'_0 of p_0.

(ii) Let i be the first node belonging to $\mathscr{S}$ such that $N(i) \neq \emptyset$, $|p(i)| \geq 2$ and i adopted a product t in the reduction sequence $\mathscr{S}' \to^* p'_0$. So product t was first adopted in $\mathscr{S}'$ by some node $a_{S,t,i}$ and then by i. By the definition of the thresholds functions in $\mathscr{S}'$ node i can adopt product t in the network $\mathscr{S}$. Repeating this procedure we obtain the desired restriction p_0 of p'_0. □

A disadvantage of this transformation is that it yields an exponential blow up in the number of nodes. Indeed, $\mathscr{S}'$ has in the worst case $n + n2^n|P|$ nodes, where n is the number of nodes in $\mathscr{S}$.

A smaller increase can be achieved by the second transformation that takes as input an equitable network $\mathscr{S} := (G, P, p, \theta)$. First we add a new product t_0 to P. Then for each node i such that $N(i) \neq \emptyset$ and $|p(i)| \geq 2$ we remove the edges $j \to i$ for each node $j \in N(i)$ and perform the following steps for each product $t \in p(i)$:

- add a new node $a_{t,i}$,
- put $p(a_{t,i}) := \{t, t_0\}$ and $\theta(a_{t,i}) := \theta(i, t)$,
- add the edges $j \to a_{t,i}$ for each node $j \in N(i)$,
- add the edge $a_{t,i} \to i$,
- put $\theta(i) := \frac{1}{|p(i)|}$.

Call the resulting network $\mathscr{S}'$. Note that $\mathscr{S}'$ has $\leq n(|P|+1)$ nodes, where n is the number of nodes in $\mathscr{S}$. The following result relates the networks $\mathscr{S}$ and $\mathscr{S}'$.

Theorem 7.2 *Consider the equitable networks $\mathscr{S}$ and $\mathscr{S}'$.*

(i) If $\mathscr{S} \to^* p_0$ for a final network p_0 given $\mathscr{S}$, then for some extension p'_0 of p_0 we have $\mathscr{S}' \to^* p'_0$, where p'_0 is a final network given $\mathscr{S}'$.

(ii) If $\mathscr{S}' \to^* p'_0$ for a final network p'_0 given $\mathscr{S}'$, then for a restriction p_0 of p'_0 we have $\mathscr{S} \to^* p_0$, where p_0 is a final network given $\mathscr{S}$.

Proof. The proof is analogous to that of Theorem 7.1 and omitted. □

8 Conclusions and future work

We have introduced a diffusion model in the presence of multiple competing products and studied some basic questions. We have provided characterizations of the underlying graph structure for determining whether a product can spread or will necessarily spread to the whole network, and of the networks that admit a unique outcome. We also studied the complexity of various problems that are of interest for networks that do not admit a unique outcome, such as the problems of computing the minimum or maximum number of nodes that will adopt a given product in a final network, or the problem of determining whether a given node has to adopt some (resp. a given) product in all final networks.

Regarding the results of Section 6, it would be interesting to see if the negative results can be alleviated by studying special cases of networks. One example is to find classes of graphs for which we can have efficient constant factor approximation algorithms for the MIN-ADOPTION problem. We are also not yet aware if the same hardness results hold for equitable networks. Finally, it would be interesting to study such problems for graphs that resemble real networks with respect to degree distribution or other graph theoretic properties.

Below we outline some further topics for future work (and some partial answers based on previous works).

Optimizing the spread of a product Given a diffusion model, one important problem, especially in the context of viral marketing is: given a network (G, P, p, θ), a product $t \in P$, and $k \geq 0$, we wish to find the optimal set S of nodes, under the restrictions that $|S| \leq k$ and $t \in p(i)$ for $i \in S$, such that if we give the product t to the members of S, optimal spread is achieved.

The parameter k indicates a bound on the budget for the company's advertising campaign. This problem was initially studied for the case of a single product, and when the thresholds are random variables in [16] (as noted in [16], when thresholds are fixed numbers strong inapproximability results hold). Some extensions for the case of two products have recently appeared in [4] where various options on how nodes decide when choosing between two products have been proposed. In most cases however, the techniques of [16] cannot be applied and algorithmic results are still elusive. It would be interesting to make further progress on this for multiple products.

Game theoretic analysis A game theoretic analysis for players choosing between two products has been presented in [19]. An extension with the additional option of adopting both products has also been considered in [14] (e.g. choosing to have two operating systems in your PC, instead of just one).

Recently [22] used the model introduced here to study consequences of adopting products by the nodes forming a social network. This leads to a study of strategic games in which the nodes decide which product to choose (or decide not to adopt any). In particular, deciding whether a game in this class has a pure Nash equilibrium is NP-complete. Pure Nash equilibria always exist when the underlying graph is a DAG or has no source nodes.

We plan to study in the proposed model other game theoretic aspects, by considering a strategic game between the producers who decide to offer their products for free to some selected nodes. A limited case for two players was studied in [1] (see also [23]) in a simpler model in which no thresholds for the adoption of the product exist. We are particularly interested in analyzing the set of Nash equilibria in the presence of multiple products, as well as in introducing weights and thresholds in the model of [1].

Introducing new products When a new product is introduced in a market, it is natural to assume that this takes place when various customers have already adopted some other product. The issue is then whether some nodes would switch to the new product. The present model does not allow us to study such a problem since the input network for such an analysis is already a final network and we stipulate that the choices of the nodes are final. Allowing a switching by a node to a new product can result in an 'illegal' network, in which choices of some nodes are not anymore justified and have to be reconsidered.

In contrast, in the framework of [22] such a study is possible, since the input is simply a strategy profile that is an assignment of products to nodes (with a special 'no-choice' strategy allowed). Currently, we initiated in this setting a study of the consequences of introducing new products by means of improvement paths in the sense of [18], the special case of which is the best-response dynamics. We also plan to study the consequences of adding new products or new links to a social network.

Acknowledgements

The **MIN-ADOPTION** and **MAX-ADOPTION** problems from Section 6 were suggested to us by Berthold Vöcking. The second author was partially supported by the Basic Research Funding Program (BRFP) of the Athens University of Economics and Business.

References

[1] N. Alon, M. Feldman, A. D. Procaccia, and M. Tennenholtz. A note on competitive diffusion through social networks. *Inf. Process. Lett.*, 110(6):221–225, 2010.

[2] K. R. Apt and E. Markakis. Diffusion in social networks with competing products. In *Proc. 4th International Symposium on Algorithmic Game Theory (SAGT11)*, volume 6982 of *Lecture Notes in Computer Science*, pages 212–223. Springer, 2011.

[3] S. Bharathi, D. Kempe, and M. Salek. Competitive influence maximization in social networks. In *Proc. 3rd International Workshop on Internet and Network Economics (WINE)*, pages 306–311, 2007.

[4] A. Borodin, Y. Filmus, and J. Oren. Threshold models for competitive influence in social networks. In *Proc. 6th International Workshop on Internet and Network Economics (WINE 2010)*, pages 539–550, 2010.

[5] T. Carnes, C. Nagarajan, S. M. Wild, and A. van Zuylen. Maximizing influence in a competitive social network: A follower's perspective. In *Proc. 9th International Conference on Electronic Commerce (ICEC)*, pages 351–360, 2007.

[6] C. P. Chamley. *Rational herds: Economic models of social learning*. Cambridge University Press, 2004.

[7] N. Chen. On the approximability of influence in social networks. *SIAM J. Discrete Math.*, 23(3):1400–1415, 2009.

[8] P. Domingos and M. Richardson. Mining the network value of customers. In *KDD*, pages 57–66, 2001.

[9] D. Easley and J. Kleinberg. *Networks, Crowds, and Markets*. Cambridge University Press, 2010.

[10] J. Goldenberg, B. Libai, and E. Muller. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing Letters*, 12(3):211–223, 2001.

[11] D. Golovin and A. Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *J. Artif. Intell. Res. (JAIR)*, 42:427–486, 2011.

[12] S. Goyal. *Connections: An introduction to the economics of networks*. Princeton University Press, 2007.

[13] M. Granovetter. Threshold models of collective behavior. *American Journal of Sociology*, 83(6):1420–1443, 1978.

[14] N. Immorlica, J. M. Kleinberg, M. Mahdian, and T. Wexler. The role of compatibility in the diffusion of technologies through social networks. In J. K. MacKie-Mason, D. C. Parkes, and P. Resnick, editors, *ACM Conference on Electronic Commerce*, pages 75–83. ACM, 2007.

[15] M. Jackson. *Social and Economic Networks*. Princeton University Press, Princeton, 2008.

[16] D. Kempe, J. M. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In L. Getoor, T. E. Senator, P. Domingos, and C. Faloutsos, editors, *KDD*, pages 137–146. ACM, 2003.

[17] J. Kostka, Y. A. Oswald, and R. Wattenhofer. Word of mouth: Rumor dissemination in social networks. In *SIROCCO*, pages 185–196, 2008.

[18] D. Monderer and L. S. Shapley. Potential games. *Games and Economic Behaviour*, 14:124–143, 1996.

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

[20] E. Mossel and S. Roch. On the submodularity of influence in social networks. In *STOC*, pages 128–134, 2007.

[21] T. Schelling. *Micromotives and Macrobehavior*. Norton, 1978.

[22] S. Simon and K. R. Apt. Choosing products in social networks. Manuscript, CWI, Amsterdam, The Netherlands, 2012. Computing Research Repository (CoRR), `http://arxiv.org/abs/1202.2209`.

[23] R. Takehara, M. Hachimori, and M. Shigeno. A comment on pure-strategy Nash equilibria in competitive diffusion games. *Inf. Process. Lett.*, 112(3):59–60, 2012.

[24] F. Vega-Redondo. *Complex Social Networks*. Cambridge University Press, 2007.