Welfare Maximization with Friends-of-Friends Network Externalities

Sayan Bhattacharya¹ · Wolfgang Dvořák² ·
Monika Henzinger² · Martin Starnberger²

Published online: 10 April 2017

© The Author(s) 2017. This article is published with open access at Springerlink.com

Abstract Online social networks allow the collection of large amounts of data about the influence between users connected by a friendship-like relationship. When distributing items among agents forming a social network, this information allows us to exploit network externalities that each agent receives from his neighbors that get the same item. In this paper we consider Friends-of-Friends (2-hop) network externalities, i.e., externalities that not only depend on the neighbors that get the same item but also on neighbors of neighbors. For these externalities we study a setting where multiple different items are assigned to unit-demand agents. Specifically, we study the problem of welfare maximization under different types of externality functions. Let n be the number of agents and m be the number of items. Our contributions are the following: (1) We show that welfare maximization is APX-hard; we show that even for step functions with 2-hop (and also with 1-hop) externalities it is NP-hard to approximate social welfare better than $(1 - 1/e)$. (2) On the positive side we present (i) an $O(\sqrt{n})$ -approximation algorithm for general concave externality functions, (ii) an $O(\log m)$ -approximation algorithm for linear externality functions, and (iii) a $\frac{5}{18}(1 - 1/e)$ -approximation algorithm for 2-hop step function externalities. We also improve the result from [7] for 1-hop step function externalities by giving a $\frac{1}{2}(1 - 1/e)$ -approximation algorithm.

✉ Wolfgang Dvořák
wolfgang.dvorak@univie.ac.at

Monika Henzinger
monika.henzinger@univie.ac.at

Martin Starnberger
martin.starnberger@univie.ac.at

¹ Institute of Mathematical Sciences, Chennai, India

² Universität Wien, Fakultät für Informatik, Währingerstraße 29, 1090 Vienna, Austria

Keywords Network externalities · Welfare maximization · Approximation algorithms · Social networks

1 Introduction

Assume you have to form a committee and need to decide whom to choose as a member. It seems like a good strategy to select members from your network that are well-connected to the whole field so that not only the knowledge of the actual members but also of their whole network can be called upon when needed. Along the same vein assume you want to play a multiplayer online game but you do not have enough friends who are willing to play with you. Then it is a good idea to ask these friends to contact their friends whether they are willing to play as well. Both these settings can be modeled by a social network graph and in both settings not the *direct* (or *1-hop*) neighbors alone, but instead the 1-hop neighbors in combination with the *neighbors of neighbors* (or *2-hop neighbors*) are the decisive factor. Note that the 2-hop neighborhoods cannot be modeled by 1-hop neighborhoods through the insertion of an additional edge (to the neighbor of the neighbor) as we require that every *participating* neighbor of a neighbor is adjacent to a *participating* neighbor. In the above example, we can only get the opinion of a contact of a contact if we asked the contact before. In the same way, the participation of a friend of a friend will only be possible if there is a participating friend that invites him.

There has been a large body of work by social scientists and, in the last decade, also by computer scientists (see e.g., the influential paper by Kempe, Kleinberg, and Tardos [26] and its citations) to model and analyze the effect of 1-hop neighborhoods. The study of 2-hop neighborhoods has received much less attention (see e.g., [15, 25]). This is surprising as a recent study [20] of the Facebook network shows that the median Facebook user has 31k people as “friends of friends” and due to some users with very large friend lists, the average number of friends-of-friends reaches even 156k. Thus, even if each individual friend of a friend has only a small influence on a Facebook user, in aggregate the influence of the friends-of-friends might be large and should not be ignored.

We, therefore, initiate the study of the influence of 2-hop neighborhoods in the popular *assignment* setting, where items are assigned to users whose values for the item depend on who else in their neighborhood has the item. There is a large body of work on mechanisms and pricing strategies for this problem with a single [1, 3, 5, 6, 9, 10, 16, 19, 21, 28] or multiple items [2, 7, 12, 18, 24, 30–32] when the valuation function of a user depends *solely* on the 1-hop neighborhood of a user and the user itself. All this work assumes that there is an infinite supply of items (of each type if there are different items) and the users have unit-demand, that is, they want to buy only *one* item. This is frequently the case, for example, if the items model competing products or if the user has to make a binary decision between participating or not participating. In the above examples, this requirement would model that each user can only be in one committee or play one game at a time.

Thus, we study the allocation of items to users in a setting with 2-hop network externalities, where the valuation that a user derives from the products depends on

herself, her 1-hop, and her 2-hop neighborhood with the goal of maximizing the social welfare of the allocation. The prior work that is most closely related to our work is the work by Bhalgat et al. [7], where they study the multi-item setting with 1-hop externality functions and give approximation algorithms based on LP-relaxations and randomized rounding for different classes of externality functions. For linear externalities they give a $1/64$ -approximation algorithm and for step function externalities they get an approximation ratio of $(1 - 1/e)/16 \approx 0.04$. Additionally they present a $2^{O(d)}$ -approximation algorithm for convex externalities that are bounded by polynomials of degree d and a polylogarithmic approximation algorithm for submodular externalities.

1.1 Our Results

The Model Let $G = (V, E)$ be an undirected graph modeling the social network. Consider any agent $j \in V$ who receives item $i \in I$, and let $S_{ij} \subseteq V \setminus \{j\}$ denote the (2-hop) *support* of agent j for item i : this is the set of agents who contribute towards the valuation of j . Specifically, an agent $j' \in V \setminus \{j\}$ belongs to the set S_{ij} iff j' gets item i and the following condition holds: either j' is a neighbor of j (i.e., $(j, j') \in E$), or j and j' have a common neighbor j'' who also gets item i . The valuation received by agent j is equal to $\lambda_{ij} \cdot f_{ij}(|S_{ij}|)$, where λ_{ij} is the agent's *intrinsic valuation* and $f_{ij}(|S_{ij}|)$ is her 2-hop *externality* for item i . The goal is to compute an assignment of items to the agents that maximizes the social welfare, which is defined as the sum of the valuations obtained by the agents.

We study three types of 2-hop externality functions, namely concave, linear and step function externalities.

Step-Function Externalities Consider a game requiring a minimal or fixed number of players (larger than two), e.g., Bridge or Canasta, then the externality is a step function. For step functions (Theorem 9) we show that it is NP-hard to approximate the social welfare within a factor of $(1 - 1/e)$. The result holds for 1-hop and 2-hop externalities. We also show that the problem remains APX-hard when the number of items is restricted to 2 (Theorem 10). Then we give a $5/18 \cdot (1 - 1/e) \approx 0.17$ -approximation algorithm for 2-hop step function externalities (Theorem 6). Our technique also leads to a *combinatorial* $(1 - 1/e)/2 \approx 0.3$ -approximation algorithm for 1-hop step function externalities (Theorem 7), improving the approximation ratio of the LP-based algorithm in [7].

Linear Externalities We show that social welfare maximization for linear 2-hop externality functions is APX-hard¹ (Theorem 5) and give an $O(\log n)$ -approximation algorithm (Theorem 3). Moreover, for linear externality functions we can relax the unit-demand requirement. Specifically, we can handle the setting where each user j

¹Theorem 3.1 in [7] claims that the welfare maximization problem for linear 1-hop externality functions in complete graphs is MaxSNP-hard, which would imply our result, but, as we show in Appendix C, this claim is not true.

can be assigned up to c_j different items (still a user cannot be assigned the same item twice), where c_j is a parameter given in the input.²

Concave Externalities We give an $O(\sqrt{n})$ -approximation algorithm when the externality functions $f_{ij}(\cdot)$ are concave and monotone (Theorem 1) and show that the hardness results for step-functions extend to this setting (Theorem 2). That is, it is NP-hard to approximate the social welfare within a factor of $(1 - 1/e)$, and the problem remains APX-hard when only two different items are considered.

Extensions Our algorithms for linear and concave externalities can be further generalized to allow a weighting of 2-hop neighbors so that 2-hop neighbors have a lower weight than 1-hop neighbors. This can be useful if it is important that the influence of 2-hop neighbors does not completely dominate the influence of the 1-hop neighbors.

Techniques The main challenge in dealing with 2-hop externalities is as follows. Fix an agent j who gets an item i , and let $V_i \subseteq V$ denote the set of all agents who get item i . Recall that agent j 's externality is given by $f_{ij}(|S_{ij}|)$, where the set S_{ij} is called the support of agent j . The problem is that $|S_{ij}|$, as a function of $V_i \setminus \{j\}$, is not submodular. This is in sharp contrast with the 1-hop setting, where the support for the agent's externality comes only from the set of her 1-hop neighbors who receive item i .

All the mechanisms in [7] use the same basic approach: First solve a suitable LP-relaxation and then round its values independently for each item i . In the 2-hop setting, however, the lack of submodularity of the support size (as described above) leads to many dependencies in the rounding step. Nevertheless, we show how to extend the technique in [7] to achieve the approximation algorithm for linear 2-hop externality functions, using a novel LP. We further give a simple combinatorial algorithm with an approximation guarantee of $O(\sqrt{n})$ for 2-hop concave externalities. For this, we show that either an $\Omega(1/\sqrt{n})$ -fraction of the optimal social welfare comes from a single item, or we can reduce our problem to a setting with 1-hop step function externalities by losing a $(1 - \Omega(1/\sqrt{n}))$ -fraction of the objective.

Our approach for 2-hop step functions is different. We use a novel decomposition of the graph into a maximal set of disjoint connected sets of size 3, 2, and 1. We say an assignment is *consistent* if it assigns all the nodes (i.e., users) in the same connected set the same item. We show first that restricting to consistent assignments reduces the maximum welfare by at most a factor of 5/18. Finally, we show that finding the optimal consistent assignment is equal to maximizing social welfare in a scenario where agents are not unit-demand, do not influence each other, and have valuation functions that are fractionally subadditive in the items they get assigned. For the latter we use the $(1 - 1/e)$ -approximation algorithm by Feige [14].

²This is also true for the results in [7]. In both results, the assumption is that the valuation functions are additive over the items.

1.2 Related Work

As discussed earlier the prior work that is most closely related to our work is the work by Bhargat et al. [7], that studies a similar setting where multiple different items are assigned to unit-demand agents with 1-hop externalities.

Beside that there is a bunch of work on pricing strategies for selling an item in social networks that exploit 1-hop network externalities to maximize the revenue of a seller [1, 5, 10, 16, 21, 28], most prominent the work of Hartline et al. [21]. In these settings one typically first sells products for a discount to some customers to later exploit their positive influence on the other customers to maximize revenue. As the revenue maximization problems in the different models are computational hard one goes for approximations that are based on randomized selection of customers and/or submodular function maximization for selecting an initial set of customers who get the product for free.

The work of Anshelevich et al. [4] studies friends-of-friends benefits in a network formation game setting, where each node of a social network has to decide how to spend its resources to cultivate relationships with his friends. The strength of a relationship is then determined by the resources the two nodes put into it. The utility of a node is calculated by the strength of the relationships to its friends and the strength of the relationships between its friends and friends-of-friends. Anshelevich et al. consider two concrete functions for calculated the strength of relations, i.e. Sum and Min, and study equilibria and the price of anarchy.

Seeman and Singer [29] consider friends-of-friends for adaptive seeding in social networks. The problem they study is to select some nodes of the graph as seeds, e.g. as early adopters of a new technology that then promote it to the other nodes. As not always all nodes are available as early adopters Seeman and Singer promote a 2-phase strategy for selecting seed. In the first phase a part of the budget is spend to select some of the initial available as seeds and each of their neighbors becomes available as early adopter in the second phase (with some probability). In the second phase the remaining budget is then spend among these friends-of-friends such that information diffusion through the word-of-mouth processes is maximized.

Dobzinski et al. [11] gave an incentive-compatible combinatorial auction for complement-free Bidders that achieves an $O(\sqrt{m})$ -approximation of the optimal item allocation. They distinguish whether most of the social welfare comes from bidders that get more than $\sqrt{m}$ items or by bidders who get less than $\sqrt{m}$ items and their auction computes two assignment one which is good in the first case and one that is good in the later one and uses the one with higher social welfare. This idea is somehow similar to our $O(\sqrt{n})$ -approximation algorithm for concave externalities which computes two assignments one that is an $O(\sqrt{n})$ -approximation if in an optimal assignment most of the social welfare comes from users with large support and another that is an $O(\sqrt{n})$ -approximation if in an optimal assignment most of the social welfare comes from users with small support. However, our sub-procedures for computing the two assignments are not directly related to the sub-procedures used in [11].

2 Notations and Preliminaries

We are given a simple undirected graph $G = (V, E)$ with $|V| = n$ nodes. Each node $j \in V$ in this graph is an agent, and there is an edge $(j, j') \in E$ iff the agents j and j' are friends with each other. There is a set of m items $I = \{1, \dots, m\}$. Each item is available in unlimited supply, and the agents have *unit-demand*, i.e., each agent wants to get at most one item. An *assignment* $\mathcal{A} : V \rightarrow I$ specifies the item received by every agent, and under this assignment, $u_j(\mathcal{A}, G)$ gives the *valuation* of an agent $j \in V$. Our goal is to find an assignment that maximizes the *social welfare* $\sum_{j \in V} u_j(\mathcal{A}, G)$, i.e., the sum of the valuations of the agents.

Let $F_j^1(G)$, $F_j^2(G)$ and $F_j^{\leq 2}(G)$ be the *1-hop neighborhood*, *2-hop neighborhood*, and *2-neighborhood*, of a node $j \in V$.

$$\begin{aligned} F_j^1(G) &= \{j' \in V \mid (j, j') \in E\} \\ F_j^2(G) &= \bigcup_{j' \in F_j^1(G)} F_{j'}^1(G) \setminus (F_j^1(G) \cup \{j\}) \\ F_j^{\leq 2}(G) &= F_j^1(G) \cup F_j^2(G) \end{aligned}$$

Define $V_i(\mathcal{A}, G) = \{j \in V : \mathcal{A}(j) = i\}$ to be the set of agents who receive item $i \in I$ under the assignment $\mathcal{A}$. Let

$$N_j^1(i, \mathcal{A}, G) = F_j^1(G) \cap V_i(\mathcal{A}, G)$$

denote the 1-hop neighbors of j who receive item i under the assignment $\mathcal{A}$. Further, let

$$N_j^2(i, \mathcal{A}, G) = F_j^2(G) \cap V_i(\mathcal{A}, G) \cap \left(\bigcup_{j'' \in N_j^1(i, \mathcal{A}, G)} F_{j''}^1(G) \right)$$

denote the 2-hop neighbors of j who receive item i under the assignment $\mathcal{A}$ and are adjacent to some node in $N_j^1(i, \mathcal{A}, G)$.

The *support* of an agent $j \in V$ for item $i \in I$ is defined as $S_{ij}(\mathcal{A}, G) = N_j^1(i, \mathcal{A}, G) \cup N_j^2(i, \mathcal{A}, G)$. This is the set of agents contributing towards the valuation of j for item i . Let λ_{ij} be the *intrinsic valuation* of agent j for item i , and let $f_{ij}(|S_{ij}(\mathcal{A}, G)|)$ be the *externality* of the agent for the same item. The agent's valuation from the assignment $\mathcal{A}$ is given by the following equality.

$$u_j(\mathcal{A}, G) = \lambda_{\mathcal{A}(j), j} \cdot f_{\mathcal{A}(j), j}(|S_{\mathcal{A}(j), j}(\mathcal{A}, G)|).$$

We consider three types of externalities in this paper.

Definition 1 In *concave externality* it holds that $f_{ij}(t)$ is a monotone and concave function of t , with $f_{ij}(0) = 0$, for every item $i \in I$ and agent $j \in V$.

Definition 2 In *linear externality* it holds that for all $j \in V$, $i \in I$ and every nonnegative integer t , we have $f_{ij}(t) = t$.

We extend the step function definition of [7] as follows to 2-hop neighborhoods. In Appendix D we generalize it further to p -hop neighborhoods for $p \geq 2$.

Definition 3 For integer $s \geq 1$, in *s-step function externality* it holds that for all $j \in V$, $i \in I$ and every nonnegative integer t , we have $f_{ij}(t)$ is 1 if $t \geq s$ and 0 otherwise.

We omit the symbol G from these notations if the underlying graph is clear from the context. All the missing proofs from Sections 4 and 5 appear in the Appendix.

3 An $O(\sqrt{n})$ -Approximation for Concave Externalities

For the rest of this section, we fix the underlying graph G , and assume that the agents have concave externalities as per Definition 1. We also fix the intrinsic valuations λ_{ij} and the externality functions $f_{ij}(\cdot)$.

- Let $\mathcal{A}^* \in \arg \max_{\mathcal{A}} \left(\sum_{j \in V} u_j(\mathcal{A}) \right)$ be an assignment that maximizes the social welfare, and let $\text{OPT} = \sum_{j \in V} u_j(\mathcal{A}^*)$ be the optimal social welfare.
- Let $X^* = \{j \in V : |S_{\mathcal{A}^*(j), j}(\mathcal{A}^*)| \geq \sqrt{n}\}$ be the set of agents with support size at least $\sqrt{n}$ under the assignment $\mathcal{A}^*$, and
- let $Y^* = V \setminus X^*$ be the agents with support size less than $\sqrt{n}$ under the assignment $\mathcal{A}^*$.

Since X^* and Y^* partition the set of agents V , half of the social welfare under $\mathcal{A}^*$ is coming either from (1) the agents in X^* , or from (2) the agents in Y^* . Lemma 1 shows that in the former case there is a *uniform assignment*, where every agent gets the same item, that retrieves a $1/(2\sqrt{n})$ -fraction of the optimal social welfare. We consider the latter case in Lemma 2, and reduce it to a problem with 1-hop externalities.

The main idea behind Lemma 1 is that in $\mathcal{A}^*$ there can be at most $\sqrt{n}$ many different items that are assigned to $\sqrt{n}$ many nodes. Thus picking among these items the one contributing the most to the social welfare and assign it to all nodes gives a $\sqrt{n}$ approximation for the social welfare of X^* .

Lemma 1 If $\sum_{j \in X^*} u_j(\mathcal{A}^*) \geq \text{OPT}/2$, then there is an item $i \in I$ such that $\sum_{j \in V} u_j(\mathcal{A}^i) \geq \text{OPT}/(2\sqrt{n})$, where $\mathcal{A}^i$ is the assignment that gives item i to every agent in V , that is, $\mathcal{A}^i(j) = i$ for all $j \in V$.

Proof Define the set of items $I(X^*) = \bigcup_{j \in X^*} (\mathcal{A}^*(j))$.

We claim that $|I(X^*)| \leq \sqrt{n}$. To see why the claim holds, let $V_i^* = \{j \in V : \mathcal{A}^*(j) = i\}$ be the set of agents who receive item i under $\mathcal{A}^*$. Now, fix any item $i \in I(X^*)$, and note that, by definition, there is an agent $j \in X^*$ with $\mathcal{A}^*(j) = i$.

Thus, we have $|V_i^*| \geq |S_{ij}(\mathcal{A}^*)| \geq \sqrt{n}$. We conclude that $|V_i^*| \geq \sqrt{n}$ for every item $i \in I(X^*)$. Since $\sum_{i \in I(X^*)} |V_i^*| \leq |V| = n$, it follows that $|I(X^*)| \leq \sqrt{n}$. To conclude the proof of the lemma, we now make the following observations.

$$\begin{aligned} \sum_{j \in X^*} u_j(\mathcal{A}^*) &= \sum_{i \in I(X^*)} \sum_{\substack{j \in X^*: \\ \mathcal{A}^*(j)=i}} u_j(\mathcal{A}^*) \leq |I(X^*)| \cdot \max_{i \in I(X^*)} \left(\sum_{\substack{j \in X^*: \\ \mathcal{A}^*(j)=i}} u_j(\mathcal{A}^*) \right) \\ &\leq \sqrt{n} \cdot \max_{i \in I(X^*)} \left(\sum_{\substack{j \in X^*: \\ \mathcal{A}^*(j)=i}} u_j(\mathcal{A}^i) \right) \leq \sqrt{n} \cdot \max_{i \in I(X^*)} \left(\sum_{j \in V} u_j(\mathcal{A}^i) \right) \end{aligned}$$

The lemma holds since $\text{OPT}/(2\sqrt{n}) \leq \sum_{j \in X^*} u_j(\mathcal{A}^*)/\sqrt{n} \leq \max_{i \in I(X^*)} \left(\sum_{j \in V} u_j(\mathcal{A}^i) \right)$. $\square$

For every item $i \in I$ and agent $j \in V$, we now define the externality function $g_{ij}(t)$ and the valuation function $\hat{u}_j(\mathcal{A})$.

$$g_{ij}(t) = \begin{cases} f_{ij}(1) & \text{if } t \geq 1; \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

$$\hat{u}_j(\mathcal{A}) = \lambda_{\mathcal{A}(j),j} \cdot g_{ij}(|N_j^1(i, \mathcal{A})|) \quad (2)$$

Clearly, for every assignment $\mathcal{A} : V \rightarrow I$, we have $0 \leq \sum_{j \in V} \hat{u}_j(\mathcal{A}) \leq \sum_{j \in V} u_j(\mathcal{A})$. Also note that the valuation function $\hat{u}_j(\cdot)$ depends only on the 1-hop neighborhood of the agent j . Specifically, if an agent j gets an item i , then her valuation $\hat{u}_j(\mathcal{A})$ is $\lambda_{ij} \cdot f_{ij}(1)$ if at least one of her 1-hop neighbors also gets the same item i , and zero otherwise. Bhalgat et al. [7] gave an LP-based $O(1)$ -approximation for finding an assignment $\mathcal{A} : V \rightarrow I$ that maximizes the social welfare in this setting (also see Section 5 for a combinatorial algorithm). In the lemma below, we show that if the agents in Y^* contribute sufficiently towards OPT under the assignment $\mathcal{A}^*$, then by losing an $O(\sqrt{n})$ -factor in the objective, we can reduce our original problem to the one where the externalities are $g_{ij}(\cdot)$ and the valuations are $\hat{u}_j(\cdot)$.

The intuition behind Lemma 2 is that when considering nodes with a support smaller than $\sqrt{n}$ then by the concavity of f_{ij} the externality $g_{ij}(t) = f_{ij}(1)$ gives a $\sqrt{n}$ approximation of $f_{ij}(t)$.

Lemma 2 *If $\sum_{j \in Y^*} u_j(\mathcal{A}^*) \geq \text{OPT}/2$, then $\sum_{j \in V} \hat{u}_j(\mathcal{A}^*) \geq \text{OPT}/(2\sqrt{n})$.*

Proof Consider a node $j \in Y^*$ that makes nonzero contribution towards the objective (i.e., $u_j(\mathcal{A}^*) > 0$) and suppose that it gets item i (i.e., $\mathcal{A}^*(j) = i$). Since $u_j(\mathcal{A}^*) > 0$, we have $S_{ij}(\mathcal{A}^*) = N_j^1(i, \mathcal{A}^*) \cup N_j^2(i, \mathcal{A}^*) \neq \emptyset$, which in turn implies that $N_j^1(i, \mathcal{A}^*) \neq \emptyset$. Thus, we have $\hat{u}_j(\mathcal{A}^*) = \lambda_{ij} \cdot f_{ij}(1)$. Since $|S_{ij}(\mathcal{A}^*)| \leq \sqrt{n}$ and $f_{ij}(\cdot)$ is a concave function, we have $f_{ij}(1) \geq f_{ij}(|S_{ij}(\mathcal{A}^*)|)/|S_{ij}(\mathcal{A}^*)| \geq f_{ij}(|S_{ij}(\mathcal{A}^*)|)/\sqrt{n}$. Multiplying both sides of this

inequality by λ_{ij} , we conclude that $\hat{u}_j(\mathcal{A}^*) \geq u_j(\mathcal{A}^*)/\sqrt{n}$ for all agents $j \in Y^*$ with $u_j(\mathcal{A}^*) > 0$. In contrast, if $u_j(\mathcal{A}^*) = 0$, then the inequality $\hat{u}_j(\mathcal{A}^*) \geq u_j(\mathcal{A}^*)/\sqrt{n}$ is trivially true. Thus, summing over all $j \in Y^*$, we infer that $\sum_{j \in Y^*} \hat{u}_j(\mathcal{A}^*, G) \geq \sum_{j \in Y^*} u_j(\mathcal{A}^*, G)/\sqrt{n} \geq \text{OPT}/(2\sqrt{n})$. The lemma now follows since $\sum_{j \in V} \hat{u}_j(\mathcal{A}^*, G) \geq \sum_{j \in Y^*} \hat{u}_j(\mathcal{A}^*, G)$. $\square$

The Algorithm for Concave Externalities Algorithm 1 computes two assignments, the first in accordance with Lemma 1 to give an $O(\sqrt{n})$ -approximation if most of the social welfare is by X^* , the second in accordance with Lemma 2 to give an $O(\sqrt{n})$ -approximation if most of the social welfare is by Y^* , and then returns the assignment that has higher social welfare and thus achieves an $O(\sqrt{n})$ -approximation in any case.

Algorithm 1 Algorithm for concave externalities.

1. $\mathcal{A}' \leftarrow \arg \max_{i \in I} \left(\sum_{j \in V} u_j(\mathcal{A}^i) \right)$,
 where $\mathcal{A}^i(j) = i$ for all $i \in I$ and $j \in V$.
 2. $\mathcal{A}'' \leftarrow$ Solve the problem for valuations $\hat{u}_j(\cdot)$
 (using one of the 1-step function $O(1)$ -approx. algorithms)
 3. RETURN $\arg \max_{\mathcal{A} \in \{\mathcal{A}', \mathcal{A}''\}} \left(\sum_{j \in V} u_j(\mathcal{A}) \right)$
-

Theorem 1 *Algorithm 1 gives an $O(\sqrt{n})$ -approximation for social welfare under 2-hop, concave externalities.*

Proof Recall the notations introduced in the beginning of Section 3 and let $\mathcal{A}$ be the assignment returned by Algorithm 1. Since the set of agents V is partitioned into $X^* \subseteq V$ and $Y^* = V \setminus X^*$, either $\sum_{j \in X^*} u_j(\mathcal{A}^*) \geq \text{OPT}/2$ or $\sum_{j \in Y^*} u_j(\mathcal{A}^*) \geq \text{OPT}/2$. In the former case, Lemma 1 guarantees that $\sum_{j \in \mathcal{A}} u_j(\mathcal{A}) \geq \sum_{j \in \mathcal{A}'} u_j(\mathcal{A}') \geq \text{OPT}/(2\sqrt{n})$. In the latter case, by Lemma 2 we have $\sum_{j \in \mathcal{A}} u_j(\mathcal{A}) \geq \sum_{j \in \mathcal{A}''} u_j(\mathcal{A}'') \geq \sum_{j \in \mathcal{A}''} \hat{u}_j(\mathcal{A}'') \geq \sum_{j \in \mathcal{A}^*} \hat{u}_j(\mathcal{A}^*)/\alpha \geq \text{OPT}/(2\alpha\sqrt{n})$, where α is the approximation ratio of the algorithm use for solving the 1-step setting. We conclude that the social welfare returned by our algorithm is always within an $O(\sqrt{n})$ -factor of the optimal social welfare. $\square$

Finally, we discuss APX-hardness results for concave externalities. Notice that 1-step function externalities are indeed concave externalities and thus each hardness result for 1-step function externalities is also a hardness result for concave functions. The following theorem is immediate by the corresponding results for 1-step function externalities in Appendix E.

Theorem 2 *The problem of maximizing social welfare under concave externalities is APX-hard, that is there is no polynomial-time $1 - \frac{1}{e} + \epsilon$ -approximation algorithm (unless $P = NP$). It remains APX-hard even for the case with only two items, i.e., it is NP-hard to approximate better than a factor of $\frac{23}{24}$.*

4 An $O(\log m)$ -Approximation for Linear Externalities

In this section, we give an $O(\log m)$ -approximation for linear externalities by an LP relaxation and a randomized rounding schema.

To this end consider the LP below. Here, the variable $\alpha(i, j, k)$ indicates if both the agents $j \in V$ and $k \in F_j^1$ received item $i \in I$. If this variable is set to one, then agent j gets one unit of externality from agent k . Similarly, the variable $\beta(i, j, l)$ indicates if both the agents $j \in V, l \in V \cap F_j^2$ received item $i \in I$ and there is at least one agent $k \in F_j^1 \cap F_l^1$ who also received the same item. If this variable is set to one, then agent j gets one unit of externality from agent l . The variables $y(i, j)$ indicate if an agent $j \in V$ receives item $i \in I$. If this variable is set to one, then agent j receives item i . Clearly, the total valuation of agent j for item i is given by $\sum_{k \in V \cap F_j^1} \lambda_{ij} \cdot \alpha(i, j, k) + \sum_{l \in V \cap F_j^2} \lambda_{ij} \cdot \beta(i, j, l)$. Summing over all the items and all the agents in V , we see that the LP-objective encodes the social welfare of the set V .

$$\text{Maximize:} \quad \sum_{j \in V} \sum_{i \in I} \lambda_{ij} \cdot \left(\sum_{k \in F_j^1} \alpha(i, j, k) + \sum_{l \in F_j^2} \beta(i, j, l) \right) \quad (3)$$

$$\beta(i, j, l) \leq \min\{y(i, j), y(i, l)\} \quad \forall i \in I, j \in V, l \in V \cap F_j^2 \quad (4)$$

$$\beta(i, j, l) \leq \sum_{k \in F_j^1 \cap F_l^1} y(i, k) \quad \forall i \in I, j \in V, l \in F_j^2 \quad (5)$$

$$\alpha(i, j, k) \leq \min\{y(i, j), y(i, k)\} \quad \forall i \in I, j \in V, k \in F_j^1 \quad (6)$$

$$\sum_i y(i, j) \leq 1 \quad \forall j \quad (7)$$

$$0 \leq y(i, j), \alpha(i, j, k), \beta(i, j, l) \quad \forall i, j, k, l \quad (8)$$

Constraint 7 states that an agent can get at most one item. Constraint 6 says that if $\alpha(i, j, k) = 1$, then both $y(i, j)$ and $y(i, k)$ must also be equal to one. Constraint 4 states that if $\beta(i, j, l) = 1$, then both $y(i, j)$ and $y(i, l)$ must also be equal to one. Finally, note that if an agent $l \in V$ contributes one unit of 2-hop externality to an agent $j \in V$ for an item $i \in I$, then there must be some agent $k \in F_j^1 \cap F_l^1$ who received item i . This condition is encoded in constraint 5. Thus, we have the following lemma.

Lemma 3 *The LP is a valid relaxation for social welfare under 2-hop, linear externalities.*

Before proceeding towards the rounding scheme, we perform a preprocessing step as described in the next lemma.

Lemma 4 *In polynomial time, we can get a feasible solution to the LP that gives an $O(\log m)$ approximation to the optimal objective, and ensures that each $\alpha(i, j, k), \beta(i, j, l) \in \{0, \gamma\}$ for some real number $\gamma \in [0, 1]$, and that each $y(i, j) \leq \gamma$.*

Proof We compute the optimal solution of the LP, and partition the $\alpha(i, j, k)$ and $\beta(i, j, l)$ variables into two groups (*large* and *small*) depending on whether they are greater than or less than $1/m^2$. By losing at most a $1/m$ fraction of the objective, we can set all the small variables to zero. To see this, suppose that the claim is false, i.e., the contribution of these small variables exceeds $1/m$ fraction of the total objective. Then we can scale up all these small variables by a factor of m , set all the large $\alpha(i, j, k)$, $\beta(i, j, l)$'s to zero, and set every $y(i, j)$ to $1/m$. This will satisfy all the constraints, and the total contribution towards the objective by the erstwhile small variables will get multiplied by m , which, in turn, will imply that their new contribution actually exceeds the optimal objective. Thus, we reach a contradiction.

We discretize the range $[1/m^2, 1]$ in powers of two, thereby creating $O(\log m)$ intervals, and accordingly, we partition the large variables into $O(\log m)$ groups. The variables in the same group are within a factor 2 of each other. By losing an $O(\log m)$ factor in the approximation ratio, we select the group that contributes the most towards the LP-objective. Let all the variables in this group lie in the range $[\gamma, 2\gamma]$. We now make the following transformation. All the $\alpha(i, j, k)$, $\beta(i, j, l)$'s in this group are set to γ . This way we lose another factor of at most 2 in the LP-objective. All the remaining $\alpha(i, j, k)$, $\beta(i, j, l)$'s are set to zero. At this stage, we have ensured that each $\alpha(i, j, k)$, $\beta(i, j, l) \in \{0, \gamma\}$. Finally, we set $y(i, j) \leftarrow \min(y(i, j), \gamma)$ for each $i \in I, j \in V$. Since each $\alpha(i, j, k)$, $\beta(i, j, l) \in \{0, \gamma\}$, this transformation does not violate the feasibility of the solution. $\square$

Note that after the processing according to Lemma 4 there might exist i, j, k , and l such that $y(i, k)$ and $\beta(i, j, l)$ both equal γ , but $\alpha(i, j, k)$ equals 0. In this case we could increase the value of the solution by setting $\alpha(i, j, k)$ to γ but this is not necessary for our rounding procedure as it ignores both the α as well as the β values.

We now present the rounding scheme for the LP (see Algorithm 2). Here, the set W_i denotes the set of agents that have not yet been assigned any item when the rounding scheme enters the FOR loop for item i (see Step 3). Note that the sets T_i might overlap, but these conflicts are resolved in Line 8 by intersecting T_i with W_i , which is disjoint with all previous $T_j, j < i$.

Algorithm 2 Rounding Scheme for LP

1. In accordance with Lemma 4, compute a feasible solution to the LP.
 2. Set $T_0 \leftarrow \emptyset$, and $W_0 \leftarrow V$.
 3. FOR all items $i \in I = \{1, \dots, m\}$:
 4. Set $W_i \leftarrow W_{i-1} \setminus T_{i-1}$, and $T_i \leftarrow \emptyset$.
 5. Pick a value η_i uniformly at random from $[0, 1]$.
 6. IF $\eta_i \leq \gamma/6$:
 7. FOR all nodes $j \in V$ with probability $y(i, j)/\gamma$, set $T_i \leftarrow T_i \cup \{j\}$.
 8. Assign item i to all nodes in $W_i \cap T_i$, i.e., set $\mathcal{A}(t) \leftarrow i$ for all $t \in W_i \cap T_i$.
 9. RETURN the (random) assignment $\mathcal{A}$.
-

Lemma 5 For all $t \in V$ and all $i \in I$, we have $\mathbf{P}[t \in W_i] \geq 5/6$. Thus, $\mathbf{P}[\{t_1, t_2, t_3\} \subseteq W_i] \geq 1/2$ for all $t_1, t_2, t_3 \in V$.

Proof Fix any node $t \in V$ and any item $i \in I$, and consider an indicator random variable $\Gamma_{i't}$ that is set to one iff $t \in T_{i'}$. It is easy to verify that $\mathbf{E}[\Gamma_{i't}] = y(i', t)/6$ for all items $i' \in I$. By constraint 7 and linearity of expectation, we thus have: $\mathbf{E}[\sum_{i' < i} \Gamma_{i't}] = \sum_{i' < i} y(i', t)/6 \leq 1/6$. Applying Markov's inequality, we get $\mathbf{P}[\sum_{i' < i} \Gamma_{i't} = 0] \geq 5/6$. In other words, with probability at least $5/6$, we have that $t \notin T_{i'}$ for all $i' < i$. Under this event, we must have $t \in W_i$.

We have $\mathbf{P}[t \notin W_i] \leq 1/6$ for all $t \in \{t_1, t_2, t_3\}$, and thus $\mathbf{P}[\{t_1, t_2, t_3\} \subseteq W_i] \geq 1/2$ follows from applying union-bound over these three events. $\square$

In the first step, when we find a feasible solution to the LP in accordance with Lemma 4, we lose a factor of $O(\log m)$ in the objective. Below, we will show that the remaining steps in the rounding scheme result in a loss of at most a constant factor in the approximation ratio.

For all items $i \in I$, nodes $j \in V$, and nodes $k \in F_j^1, l \in F_j^2$, we define the random variables $X(i, j, k)$ and $Y(i, j, l)$. Their values are determined by the outcome $\mathcal{A}$ of our randomized rounding.

$$X(i, j, k) = \begin{cases} 1 & \text{if both } j \text{ and } k \text{ receive item } i; \\ 0 & \text{otherwise.} \end{cases}$$

$$Y(i, j, l) = \begin{cases} 1 & \text{if both } j \text{ and } l \text{ receive item } i, \text{ and} \\ & \text{there is some node in } F_j^1 \cap F_l^1 \text{ that also received item } i; \\ 0 & \text{otherwise.} \end{cases}$$

Now, the valuation of any agent $j \in V$ from the (random) assignment $\mathcal{A}$ is:

$$u_j(\mathcal{A}) = \sum_{i \in I} \left(\sum_{k \in F_j^1} \lambda_{ij} \cdot X(i, j, k) + \sum_{l \in F_j^2} \lambda_{ij} \cdot Y(i, j, l) \right) \quad (9)$$

We will analyze the expected contribution of the rounding scheme to each term in the LP-objective. Towards this end, we prove the following lemmas.

Lemma 6 For all $i \in I, j \in V, k \in F_j^1$, we have $\mathbf{E}_{\mathcal{A}}[X(i, j, k)] \geq \delta \cdot \text{Alpha}(i, j, k)$, where $\delta > 0$ is a sufficiently small constant.

Proof Fix an item $i \in I$, a node $j \in V$ and a node $k \in F_j^1$. By Lemma 4 we have that $\alpha(i, j, k) \in \{0, \gamma\}$ and $y(i, j) \leq \gamma, y(i, k) \leq \gamma$. If $\alpha(i, j, k) = 0$ the lemma is trivially true. Otherwise suppose for the rest of the proof that $\alpha(i, j, k) = \gamma$. From constraint (6) it follows that also $y(i, j) = y(i, k) = \gamma$. Now, we have:

$$\begin{aligned} \mathbf{E}_{\mathcal{A}}[X(i, j, k)] &= \mathbf{P}[X(i, j, k) = 1] = \mathbf{P}[j, k \in W_i] \cdot \mathbf{P}[\eta_i] \\ &\leq \gamma/6 \mid j, k \in W_i] \cdot \mathbf{P}[j, k \in T_i \mid (\eta_i \leq \gamma/6) \wedge (j, k \in W_i)] \end{aligned}$$

Now we can use the following (in)equalities: $\mathbf{P}[j, k \in W_i] \geq 1/2$ by Lemma 5; $\mathbf{P}[\eta_i \leq \gamma/6 \mid j, k \in W_i] = \gamma/6$ as η_i is independent of W_i ; $\mathbf{P}[j, k \in T_i \mid (\eta_i \leq$

$\gamma/6) \wedge (j, k \in W_i)] = 1$ as $y(i, j) = y(i, k) = \gamma$ and thus in the event that $\eta_i \leq \gamma/6$ the algorithm adds j, k to T_i with probability 1.

$$\mathbf{E}_{\mathcal{A}}[X(i, j, k)] \geq 1/2 \cdot \gamma/6 \cdot 1 = \delta \cdot \gamma = \delta \cdot \alpha(i, j, k) \quad (\text{where } \delta = 1/12)$$

□

Lemma 7 For all $i \in I, j \in V, l \in F_j^2$, we have $\mathbf{E}_{\mathcal{A}}[Y(i, j, l)] \geq \delta \cdot \beta(i, j, l)$, where δ is a sufficiently small constant.

Proof Fix an item $i \in I$, a node $j \in V$ and a node $l \in F_j^2$. By Lemma 4 we have that $\beta(i, j, l) \in \{0, \gamma\}$ and $y(i, j) \leq \gamma, y(i, l) \leq \gamma$. If $\beta(i, j, l) = 0$ the lemma is trivially true. Otherwise suppose for the rest of the proof that $\beta(i, j, l) = \gamma$ which, given constraint 4, implies that also $y(i, j) = y(i, l) = \gamma$.

Let $\mathcal{E}_i$ be the event that $\eta_i \leq \gamma/6$ (see Steps 5 & 6 in Algorithm 2). Let $Z(i, k)$ be an indicator random variable that is set to one iff node $k \in F_j^1 \cap F_l^1$ is included in the set T_i by our rounding scheme (see Step 7 in Algorithm 2). We have:

$$\mathbf{P}[\mathcal{E}_i] = \gamma/6, \text{ and } \mathbf{P}[Z(i, k) = 1 \mid \mathcal{E}_i] = y(i, k)/\gamma \text{ for all } k \in F_j^1 \cap F_l^1 \quad (10)$$

Thus, conditioned on the event $\mathcal{E}_i$, the expected number of common neighbors of j and l who are included in the set T_i is given by

$$\mu_i := \mathbf{E} \left[\sum_{k \in F_j^1 \cap F_l^1} Z(i, k) \mid \mathcal{E}_i \right] = \sum_{k \in F_j^1 \cap F_l^1} y(i, k)/\gamma \geq 1 \quad (11)$$

The above inequality is by constraint (5) of the LP. Note that conditioned on the event $\mathcal{E}_i$, the random variables $Z(i, k)$ are mutually independent. Thus, applying Chernoff bound on (11), we infer that with constant probability, at least one common neighbor of j and l will be included in the set T_i . To be more precise, define $T_{i,j,l} = T_i \cap F_j^1 \cap F_l^1$. For some sufficiently small constant δ_1 , we have:

$$\mathbf{P}[T_{i,j,l} \neq \emptyset \mid \mathcal{E}_i] = \mathbf{P} \left[\sum_{k \in F_j^1 \cap F_l^1} Z(i, k) > 0 \mid \mathcal{E}_i \right] \geq 1 - e^{-1/2} = \delta_1 \quad (12)$$

Let $\mathcal{E}_{i,j,l}$ be the event that the following two conditions hold simultaneously: (a) $T_{i,j,l} \neq \emptyset$, and (b) the nodes j, l , and an arbitrary node from $T_{i,j,l}$ are included in W_i . Now, (12) and Lemma 5 imply that $\mathbf{P}[\mathcal{E}_{i,j,l} \mid \mathcal{E}_i] \geq \delta_2$ for $\delta_2 = \delta_1/2$. Putting all these observations together, we obtain that $\mathbf{P}[Y(i, j, l) = 1] = \mathbf{P}[\mathcal{E}_i] \cdot \mathbf{P}[\mathcal{E}_{i,j,l} \mid \mathcal{E}_i] \cdot \mathbf{P}[j, l \in T_i \mid \mathcal{E}_{i,j,l} \cap \mathcal{E}_i] \geq \gamma/6 \cdot \delta_2 \cdot 1 = \delta \cdot \gamma = \delta \cdot \beta(i, j, l)$ for $\delta = \delta_2/6$. □

Theorem 3 The rounding scheme in Algorithm 2 gives an $O(\log m)$ -approximation for social welfare under 2-hop, linear externalities.

Proof In the first step, when we find a feasible solution to the LP in accordance with Lemma 4, we lose a factor of $O(\log m)$ in the objective. At the end of the remaining steps, the expected valuation of an agent $j \in V$ is given by:

$$\begin{aligned} \mathbb{E}_{\mathcal{A}}[u_j(\mathcal{A})] &= \sum_{i \in I} \lambda_{ij} \cdot \left(\sum_{k \in F_j^1} \mathbb{E}_{\mathcal{A}}[X(i, j, k)] + \sum_{l \in F_j^2} \mathbb{E}_{\mathcal{A}}[Y(i, j, l)] \right) \\ &= \Theta \left(\sum_{i \in I} \lambda_{ij} \cdot \left(\sum_{k \in F_j^1} \alpha(i, j, k) + \sum_{l \in F_j^2} \beta(i, j, l) \right) \right) \end{aligned}$$

The first equality follows from linearity of expectation, while the second equality follows from Lemma 6 and Lemma 7. Thus, the expected valuation of any agent in V is within a constant factor of the fractional valuation of the same agent under the feasible solution to the LP obtained at the end of Step 1 (see Algorithm 2). Summing over all the agents in V , we get the theorem. $\square$

We can generalize the above approach to the following setting: Each user j is given an integer c_j and can be assigned up to c_j different items (each at most once). The valuation of an agent j for such an assignment $\mathcal{A}$ is then given by $u_j(\mathcal{A}, G) = \sum_{i \in \mathcal{A}(j)} \lambda_{i,j} \cdot f_{\mathcal{A}(j),j}(|S_{\mathcal{A}(j),j}(\mathcal{A}, G)|)$, with unchanged definitions of $\lambda_{i,j}$, $f_{i,j}$ and $S_{i,j}(\mathcal{A}, G)$. To generalize the LP we replace, for each item i and node j , the constraint $\sum_i y(i, j) \leq 1$ by the two constraints $\sum_i y(i, j) \leq c_j$ and $y(i, j) \leq 1$ and adapt the proof of Lemma 5. We give the modified proof in Appendix A.

Finally, we show NP-hardness for linear externalities, not only in the 2-hop setting but also for 1-hop. The proof is provided in Appendix C.³

Theorem 4 *Maximizing social welfare under linear externalities is NP-hard.*

Moreover, if we consider 2-hop externality functions we get APX-hardness for linear externalities. The proof is provided in Appendix E.

Theorem 5 *Maximizing social welfare with linear externalities in a 2-neighborhood is APX-hard even for the case with 2 items, i.e., it is NP-hard to approximate better than a factor of $\frac{143}{144}$.*

5 Constant Factor Approximation for Step Function Externalities

In this section, our goal is to maximize social welfare when agents have general step function externalities, i.e., an agent needs a certain number of 1- and 2-hop neighbors having the same product to receive externality. We will show that no constant

³As mentioned before Theorem 3.1 in [7] claims that the welfare maximization problem for linear 1-hop externality functions in complete graphs is MaxSNP-hard, which would imply our result, but, as we show in Appendix C, this claim is not true.

factor approximation is possible unless a bound on the number of 2-neighbors an agent needs for receiving externality is given. We consider the case of 2-step function externalities, where only two neighbors are needed (see Definition 3) and give a $\frac{5}{18} \cdot (1 - 1/e)$ -approximation algorithm for this problem. Moreover, as an upper bound we show that there is no polynomial time $1 - \frac{1}{e} + \epsilon$ -approximation algorithm (unless $P = NP$). Notice that if we consider step functions that just require one neighbor the problem reduces to the 1-hop step function scenario in [7]. However, our algorithm gives a $\frac{1}{2} \cdot (1 - 1/e)$ -approximation for this scenario, improving the result in [7].

In the following we assume 2-step function externalities. Let $G_{V'}$ denote the subgraph induced by $V' \subseteq V$. For the rest of this section, the term “triple” will refer to any (unordered) set of three nodes $T = \{j_1, j_2, j_3\}$ such that G_T is connected. Similarly, the term “pair” will refer to any (unordered) set of two nodes $\{j_1, j_2\}$ that are connected by an edge in E .

We first compute a maximal collection of mutually disjoint triples in the graph G . We denote this collection by $\mathcal{T}$, and let $V(\mathcal{T}) = \bigcup_{T \in \mathcal{T}} T \subseteq V$. The graph $G_{V \setminus V(\mathcal{T})}$, by definition, consists of a mutually disjoint collection of pairs (say $\mathcal{P}$) and a set of isolated nodes (say B). We thus have the following lemma.

Lemma 8 *In $G = (V, E)$, there is no edge that connects a node $j \in B$ with another node in B or with a node belonging to a pair in $\mathcal{P}$. Furthermore, there is no edge that connects two nodes j, j' belonging to two different pairs $P, P' \in \mathcal{P}$.*

Definition 4 An assignment $\mathcal{A}$ is *consistent* iff two agents get the same item whenever they belong to the same triple or the same pair. To be more specific, for all $j, j' \in V$, we have that $\mathcal{A}(j) = \mathcal{A}(j')$ if either (a) $j, j' \in T$ for some triple $T \in \mathcal{T}$ or (b) $\{j, j'\} \in \mathcal{P}$.

The next lemma shows that by losing a factor of $5/18$ in the approximation ratio, we can focus on maximizing the social welfare via a consistent assignment.

Lemma 9 *The social welfare from the optimal consistent assignment is at least $(5/18) \cdot \text{OPT}$, where OPT is the maximum social welfare over all assignments.*

Algorithm 3 Convert an assignment $\mathcal{A}^*$ into a consistent one

1. FOR each triple $\{j_1, j_2, j_3\} \in \mathcal{T}$
 2. Pick $i \in \{\mathcal{A}^*(j_1), \mathcal{A}^*(j_2), \mathcal{A}^*(j_3)\}$ uniformly at random.
 3. Set $\mathcal{A}(j_1) \leftarrow i$, $\mathcal{A}(j_2) \leftarrow i$, and $\mathcal{A}(j_3) \leftarrow i$.
 4. FOR each pair $P = \{j_1, j_2\} \in \mathcal{P}$ with $u_{j_1}(\mathcal{A}^*) \geq u_{j_2}(\mathcal{A}^*)$:
 5. IF there is a triple adjacent to P with $\mathcal{A}(T) = \mathcal{A}^*(j_1)$
 6. Set $\mathcal{A}(j_1) \leftarrow \mathcal{A}^*(j_1)$ and $\mathcal{A}(j_2) \leftarrow \mathcal{A}^*(j_1)$.
 7. ELSE
 8. Set $\mathcal{A}(j_1) \leftarrow \mathcal{A}^*(j_2)$ and $\mathcal{A}(j_2) \leftarrow \mathcal{A}^*(j_2)$.
 9. FOR each $B = \{j\}$:
 10. Set $\mathcal{A}(j) \leftarrow \mathcal{A}^*(j)$.
 11. RETURN the (random) assignment $\mathcal{A}$.
-

The proof of Lemma 9 exploits Algorithm 3 that converts an assignment $\mathcal{A}^*$ that gives maximum social welfare into a consistent assignment $\mathcal{A}$. First, for each triple $\{j_1, j_2, j_3\} \in \mathcal{T}$, the algorithm picks one of the items $\mathcal{A}^*(j_1), \mathcal{A}^*(j_2), \mathcal{A}^*(j_3)$ uniformly at random, and assigns that item to all the three agents j_1, j_2, j_3 . Second, for each pair $\{j_1, j_2\} \in \mathcal{P}$, we consider the items $\mathcal{A}^*(j_1), \mathcal{A}^*(j_2)$ and let j_1 be the agent with larger $u_j(\mathcal{A}^*)$. If there is an adjacent triple that was assigned item $\mathcal{A}^*(j_1)$, then we assign item $\mathcal{A}^*(j_1)$ to both agents. Otherwise we assign $\mathcal{A}^*(j_2)$ to both agents. Finally, the remaining agents (those who are in B) get the same items as in $\mathcal{A}^*$.

Proof (Lemma 9) Let $\mathcal{A}^*$ be an assignment (not necessarily consistent) that gives maximum social welfare. It is easy to see that Algorithm 3 returns a consistent assignment $\mathcal{A}$. In the following we show that also $\mathbf{E} \left[\sum_{j \in V} u_j(\mathcal{A}) \right] \geq 5/18 \cdot \sum_{j \in V} u_j(\mathcal{A}^*)$. To this end we will show that $\mathbf{E} \left[\sum_{j \in T} u_j(\mathcal{A}) \right] \geq 5/18 \cdot \sum_{j \in T} u_j(\mathcal{A}^*)$ for each tuple $T \in \mathcal{T} \cup \mathcal{P} \cup B$; we consider the following three cases.

Case 1 ($T \in B$): Let $T = \{j\}$ and $\mathcal{A}^*(j) = i$. Since $j \in B$, it always gets the same item under $\mathcal{A}$, i.e., $\mathcal{A}(j) = i$. Now, if $u_j(\mathcal{A}^*) = 0$, then the claim is trivially true. Otherwise it must be the case that $\mathcal{A}^*(j') = i$ for some neighbor j' of j . Since $j \in B$, this neighbor j' must be part of some triple $T' \in \mathcal{T}$ (see Lemma 8). With probability at least $1/3$ all the three nodes in T' are assigned item i under $\mathcal{A}$ and at least two nodes of T' are in the 2-hop neighborhood of j . In that event j gets the same valuation as in $\mathcal{A}^*$, and we have that $\mathbf{E}[u_j(\mathcal{A})] \geq (1/3) \cdot u_j(\mathcal{A}^*)$.

Case 2 ($T \in \mathcal{P}$): Consider the pair $T = \{j_1, j_2\}$ with $\mathcal{A}^*(j_1) = i$ and $\mathcal{A}^*(j_2) = i'$. If $u_{j_1}(\mathcal{A}^*) = 0$ or $u_{j_2}(\mathcal{A}^*) = 0$, we can exclude this node from our analysis and proceed as in case 1. Otherwise consider two cases.

- Assume $i = i'$. Then it must be the case that there exists a node $j' \notin T$ with $\mathcal{A}^*(j') = i$ such that j' is either a neighbor of j_1 or a neighbor of j_2 . Since $\{j_1, j_2\} \in \mathcal{P}$, this agent j' must be part of some triple $T' \in \mathcal{T}$ (see Lemma 8). Consider the event that all the three nodes in T' are assigned item i under $\mathcal{A}$; given this event we have $u_j(\mathcal{A}) = u_j(\mathcal{A}^*)$ for $j \in \{j_1, j_2\}$. It follows that $\mathbf{E}[u_j(\mathcal{A})] \geq (1/3) \cdot u_j(\mathcal{A}^*)$.
- Assume $i \neq i'$ and w.l.o.g. $u_{j_1}(\mathcal{A}^*) \geq u_{j_2}(\mathcal{A}^*)$. Then there exists a neighbor $j'_1 \notin T$ of j_1 with $\mathcal{A}^*(j'_1) = i$ and a neighbor $j'_2 \notin T$ of j_2 with $\mathcal{A}^*(j'_2) = i'$. Since $\{j_1, j_2\} \in \mathcal{P}$, these agents must be part of some triples $T_1, T_2 \in \mathcal{T}$ (see Lemma 8). Note that T_1 and T_2 can be identical. Let $\mathcal{E}_1$ be the event that all the three nodes in T_1 are assigned item i under $\mathcal{A}$, and $\mathcal{E}_2$ be the event that all the three nodes in T_2 are assigned item i' under $\mathcal{A}$. Let $\mathcal{E}_1^c$ be the complementary event of $\mathcal{E}_1$; then we have:

$$\mathbf{E} \left[\sum_{j \in T} u_j(\mathcal{A}) \right] \geq \mathbf{P}[\mathcal{E}_1] \cdot u_{j_1}(\mathcal{A}^*) + \mathbf{P}[\mathcal{E}_2 \cap \mathcal{E}_1^c] \cdot u_{j_2}(\mathcal{A}^*)$$

Notice that the events $\mathcal{E}_2$ and $\mathcal{E}_1^c$ are not necessarily independent (there might be a triple that contains agents j_1^*, j_2^* with $\mathcal{A}^*(j_1^*) = i$ and $\mathcal{A}^*(j_2^*) = i'$), but

can be only positively correlated. Thus, we can use $\mathbf{P}[\mathcal{E}_2](1 - \mathbf{P}[\mathcal{E}_1])$ to lower bound $\mathbf{P}[\mathcal{E}_2 \cap \mathcal{E}_1^c]$.

$$\begin{aligned} \mathbf{E}\left[\sum_{j \in T} u_j(\mathcal{A})\right] &\geq \mathbf{P}[\mathcal{E}_1] \cdot u_{j_1}(\mathcal{A}^*) + \mathbf{P}[\mathcal{E}_2](1 - \mathbf{P}[\mathcal{E}_1]) \cdot u_{j_2}(\mathcal{A}^*) \\ &\geq 1/3 \cdot u_{j_1}(\mathcal{A}^*) + 2/9 \cdot u_{j_2}(\mathcal{A}^*) \end{aligned}$$

For the last inequality we use $\mathbf{P}[\mathcal{E}_i] \geq 1/3$ and $u_{j_1}(\mathcal{A}^*) \geq u_{j_2}(\mathcal{A}^*)$. It then follows that $\mathbf{E}[\sum_{j \in T} u_j(\mathcal{A})] \geq 5/18 \cdot (u_{j_1}(\mathcal{A}^*) + u_{j_2}(\mathcal{A}^*)) = 5/18 \cdot \sum_{j \in T} u_j(\mathcal{A}^*)$ (again by $u_{j_1}(\mathcal{A}^*) \geq u_{j_2}(\mathcal{A}^*)$).

Case 3 ($T \in \mathcal{T}$): Consider the triple $T = \{j_1, j_2, j_3\}$. For each $j \in \mathcal{T}$ with probability at least $1/3$, all these three nodes are assigned item $\mathcal{A}^*(j)$ under $\mathcal{A}$, and in this event we have $u_j(\mathcal{A}) \geq u_j(\mathcal{A}^*)$. It follows that $\mathbf{E}[\sum_{j \in T} u_j(\mathcal{A})] \geq 1/3 \cdot \sum_{j \in T} u_j(\mathcal{A}^*)$.

Now, we take the sum of the above inequalities $\mathbf{E}[\sum_{j \in T} u_j(\mathcal{A})] \geq 5/18 \cdot \sum_{j \in T} u_j(\mathcal{A}^*)$ for each tuple $T \in \mathcal{T} \cup \mathcal{P} \cup B$, and by linearity of expectation infer that the expected social welfare under the consistent assignment $\mathcal{A}$ is within a factor of $5/18$ of the optimal social welfare. This concludes the proof of the lemma. $\square$

Next, we will give a $(1 - 1/e)$ -approximation algorithm for finding a consistent assignment of items that maximizes the social welfare. Along with Lemma 9, this will imply the main result of this section (see Theorem 6).

We use the term “resource” to refer to either a pair $P \in \mathcal{P}$ or an agent $j \in B$. Let $\mathcal{R} = \mathcal{P} \cup B$ denote the set of all resources. We say that a resource $r \in \mathcal{R}$ *neighbors* a triple $T \in \mathcal{T}$ iff in the graph $G = (V, E)$ either (a) $r = \{j, j'\} \in \mathcal{P}$ and some node in $\{j, j'\}$ is adjacent to some node in T , or (b) $r = j \in B$ and j is adjacent to some node in T . We slightly abuse the notation (see Section 2) and for a triple $T \in \mathcal{T}$ let $N(T) \subseteq \mathcal{R}$ denote the set of resources that are neighbors of T .

By definition, every consistent assignment ensures that if two agents belong to the same triple in $\mathcal{T}$ (resp. the same pair in $\mathcal{P}$), then both of them get the same item. We say that *the item is assigned to a triple (resp. resource)*. Note that the triples do not need externality from outside. To be more specific, the contribution of a triple $T \in \mathcal{T}$ to the social welfare is always equal to $\sum_{j \in T} \lambda_{i,j}$, where i is the item assigned to T . Resources, however, do need outside externality, which by Lemma 8 can come only from a triple in $\mathcal{T}$.

Lemma 10 *In a consistent assignment, if a resource $r \in \mathcal{R}$ makes a positive contribution to the social welfare, then it neighbors some triple $T_r \in \mathcal{T}$, and both the resource r and the triple T_r receive the same item.*

Proof If a resource contributes a nonzero amount to the social welfare, then it must receive nonzero externality from the assignment. By Lemma 8, such externality can come only from a triple in $\mathcal{T}$. The lemma follows. $\square$

Thus, given a consistent assignment $\mathcal{A}$ consider the following mapping $T_{\mathcal{A}}(r)$ of a resource $r \in \mathcal{R}$ to triples in $\mathcal{T}$ in accordance with Lemma 10: If the resource r makes zero contribution towards the social welfare (a case not covered by the lemma), then we let $T_{\mathcal{A}}(r)$ be any arbitrary triple from $\mathcal{T}$. Otherwise $T_{\mathcal{A}}(r)$ denotes an (arbitrary) neighboring triple of $\mathcal{T}$ that receives the same item as r . We say that the triple $T_{\mathcal{A}}(r)$ *claims* the resource r .

For ease of exposition, let $\lambda_{i,r}(T)$ be the valuation of the resource r when both the resource and the triple T that claims it get item $i \in I$, i.e.,

$$\lambda_{i,r}(T) = \begin{cases} \lambda_{i,j} + \lambda_{i,j'} & \text{if } r = \{j, j'\} \in \mathcal{P} \text{ and } r \in N(T); \\ \lambda_{i,j} & \text{if } r = j \in B \text{ and } r \in N(T); \\ 0 & \text{if } r \notin N(T). \end{cases}$$

Now, any consistent assignment $\mathcal{A}$ can be interpreted as follows. Under such an assignment, every triple $T \in \mathcal{T}$ claims the subset of the resources $S_T = \{r \in \mathcal{R} \mid T_{\mathcal{A}}(r) = T\}$; the subsets corresponding to different triples being mutually exclusive. A triple T and the resources in S_T all get the same item (say $i \in I$). The valuation obtained from them is $u_T(S_T, i) = \sum_{j \in T} \lambda_{i,j} + \sum_{r \in S_T} \lambda_{i,r}(T)$.

If our goal is to maximize the social welfare, then, naturally, for every triple T , we will pick the item that maximizes $u_T(S_T, i)$, thereby extracting a valuation of $u_T(S_T) = \max_i u_T(S_T, i)$. The next lemma shows that this function is fractionally subadditive.⁴

Lemma 11 *The function $u_T(S_T)$ is fractionally subadditive in S_T .*

Proof Note that the function $u_T(S_T, i)$ is additive in S_T , for every item $i \in I$. The lemma follows from the fact that the maximum of a set of linear functions is fractionally sub-additive. $\square$

The preceding discussion shows that the problem of computing a consistent assignment for welfare maximization is equivalent to the following setting. We have a collection of triples $\mathcal{T}$ and a set of resources $\mathcal{R}$. We will distribute these resources amongst the triples, i.e., every triple T will get a subset $S_T \subseteq \mathcal{R}$, and these subsets will be mutually exclusive. The goal is to maximize the sum $\sum_{T \in \mathcal{T}} u_T(S_T)$, where the functions $u_T(\cdot)$'s are fractionally subadditive. By a celebrated result of Feige [14], we can get an $(1 - 1/e)$ -approximation algorithm for this problem if we can implement the following subroutine (called *demand oracle*) in polynomial time: Each resource r is given a “cost” $p(r)$ and we need to determine for each triple T a set of resources S_T^* that maximizes $u_T(S_T) - \sum_{r \in S_T} p(r)$ over all sets S_T . Such a demand oracle can be implemented in polynomial time using a simple greedy algorithm for each T and each item i : Add a resource r to S_T^* iff $\lambda_{i,r}(T) > p(r)$.

⁴A function is fractionally subadditive if it can be expressed as the maximum of additive functions [14].

The result of the approximation algorithm assigns each triple T a subset S_T and we then pick the item i that maximizes $u_T(S_T, i)$ over all items i . Together with Lemma 8, this implies the theorem stated below.

Theorem 6 *We can get a polynomial-time $\frac{5}{18} \cdot (1 - 1/e)$ -approximation algorithm for the problem of maximizing social welfare under 2-step function externalities.*

The algorithm can be easily adapted for 1-hop step function externalities. The difference is that instead of computing a maximal collection $\mathcal{T}$ of mutually disjoint triples, one computes a maximal collection of mutually disjoint pairs.

Theorem 7 *We can get a polynomial-time $\frac{1}{2} \cdot (1 - 1/e)$ -approximation algorithm for maximizing social welfare under 1-step function externalities.*

In Appendix D we further generalize the above theorems to s -step functions in a s -hop neighborhoods for $s \geq 2$.

Finally, we present our hardness results for step functions. By a reduction from MAX INDEPENDENT SET (see Appendix E) we can show that, for unbounded s , there is no constant factor approximation. The main idea is that we modify the graph such that we replace each edge by a path of length three and each of the original nodes j wants a different item, while j can only get positive externalities when having a support of $2\delta_j$ (δ_j the node degree of j). The valuations of the newly introduced nodes are set to 0. That is, nodes that are adjacent in the original graph have two common nodes in the 2-neighborhood in the constructed graph, want different items, need all their neighbors as support, and thus only one of them can have positive valuation.

Theorem 8 *For any $\varepsilon > 0$ the problem of maximizing social welfare under arbitrary s -step function externalities is not approximable within $O(n^{1/4-\varepsilon})$ unless $\text{NP} = \text{P}$, and not approximable within $O(n^{1/2-\varepsilon})$ unless $\text{NP} = \text{ZPP}$.*

Second, we show that maximizing social welfare under 2-step function externalities is APX-hard and thus no PTAS can exist. This is by a reduction from MAX COVERAGE. The APX-hardness for the setting with only two items is by a reduction from MAX-SAT.

Theorem 9 *The problem of maximizing social welfare under 2-step function externalities is APX-hard, in particular, there is no polynomial time $1 - \frac{1}{e} + \epsilon$ -approximation algorithm (unless $\text{P} = \text{NP}$).*

Furthermore, the problem remains APX-hard (although with a larger constant) even if there are only two items.

Theorem 10 *Maximizing social welfare with s -step function externalities in a p -neighborhood ($p \geq 1$) is APX-hard even for the case with 2 items, i.e., it is NP-hard to approximate better than a factor of $\frac{23}{24}$.*

6 Conclusion

In this work we considered friends-of-friends externalities in a setting where multiple items are assigned to unit-demand agents and one aims to maximize social-welfare. We considered three kinds of externality functions, i.e., concave, linear, and step-functions, and gave approximation as well as hardness of approximation results. In particular, (1) for concave externalities we gave an $O(\sqrt{n})$ -approximation algorithm, (2) for linear externalities we gave an $O(\log m)$ -approximation algorithm, and (3) for 2-step function externalities we gave an $\frac{5}{18}(1 - 1/e)$ -approximation algorithm. We complimented these algorithms by hardness results showing that maximizing welfare for any of these externalities is APX-hard. Moreover, we improved results for the 1-hop setting, by (a) an improved algorithm for 1-hop step function externalities and (b) by a correction and improvements of the complexity results for 1-hop externalities.

We identify the following directions for future research. First, there are gaps between the approximation ratios of our algorithm and the hardness results we provided. Narrowing these gaps by better algorithms and stronger upper bounds is an obvious direction for future research. Second, beyond the types of externalities studied in this work there are more general notions of externalities, like submodular externalities, that received attention in the 1-hop setting [7]. Another question for future research is how to extend our results to these types of externalities. Finally, friends-of-friends externalities are also relevant for different problem settings with network externalities. For instance one can consider strategic agents and the problem of maximizing social welfare in an incentive-compatible way [7], or the problem of revenue maximization when selling a product [21].

Acknowledgments Open access funding provided by University of Vienna. A preliminary version of this paper has been presented at the 32nd International Symposium on Theoretical Aspects of Computer Science (STACS 2015) [8]. The authors want to thank the anonymous referees of the current paper and the preceding STACS paper for their comments which helped to improve the presentation.

The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013) under grant agreement no. 317532 and from the Vienna Science and Technology Fund (WWTF) through project ICT10-002. The authors are also grateful to Parinya Chalermsook for helpful discussions on “hardness of approximation”.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Appendix A: Multi-Demand Setting for Linear Externalities

We show how to adapt the proof of Lemma 5 to the more general setting. Recall that each user can be assigned up to c_j different items (each at most once). The rounding scheme is slightly adapted for the Multi-Demand setting as following: the sets W_i are replaced by multi-sets W_i such that in W_0 there c_j copies for each $j \in V$; in step 4 when subtracting a set T_{i-1} we delete only one copy of each $j \in T_{i-1}$.

Lemma 12 For every node $t \in V$ and every $i \in I$, we have $\mathbf{P}[t \in W_i] \geq 5/6$. Thus, $\mathbf{P}[\{t_1, t_2, t_3\} \subseteq W_i] \geq 1/2$ for all $t_1, t_2, t_3 \in V$.

Proof Fix any node $j \in V$ and any item $i \in I$, and consider an indicator random variable $\Gamma_{i'j}$ that is set to one iff $j \in T_{i'}$. It is easy to check that

$$\mathbf{E}[\Gamma_{i'j}] = y(i', j)/6$$

for all items $i' \in I$. By constraint 7 and linearity of expectation, we thus have:

$$\mathbf{E}[\sum_{i' < i} \Gamma_{i'j}] = \sum_{i' < i} y(i', j)/6 \leq c_j/6.$$

Applying Markov's inequality, we get $\mathbf{P}[\sum_{i' < i} \Gamma_{i'j} < c_j] \geq 5/6$. In other words, with probability at least $5/6$, we have that less than c_j many items have been assigned to j when item i is processed. Under this event, we must have $j \in W_i$. The lemma follows. $\square$

Appendix B: p -hop Externalities

In this section we generalize our notation for the p -hop case. For $p \geq 2$, let the p -hop neighborhood F_j^p and the p -neighborhood $F_j^{\leq p}$ be defined as follows.

$$F_j^p = \bigcup_{j' \in F_j^{p-1}} (F_{j'}^1) \setminus \left(\bigcup_{q < p} F_j^q \cup \{j\} \right)$$

$$F_j^{\leq p} = \bigcup_{q \leq p} F_j^q$$

Then the p -hop support $N_j^p(i, \mathcal{A})$ and the p -support $S_{ij}^p(\mathcal{A}, G)$ are defined as

$$N_j^p(i, \mathcal{A}) = \left(\bigcup_{j' \in N_j^{p-1}(i, \mathcal{A})} N_{j'}^1(i, \mathcal{A}) \right) \setminus \left(\bigcup_{q < p} N_j^q(i, \mathcal{A}) \cup \{j\} \right)$$

$$S_{ij}^p(\mathcal{A}) = \bigcup_{q \leq p} N_j^q(i, \mathcal{A})$$

Appendix C: Hardness for Linear Externalities

Bhalgat et al. [7] state that “The welfare maximization problem for linear externality and complete graphs is MaxSNP-hard”. As on complete graphs there is no difference between 1-hop, 2-hop and in general p -hop linear externalities this would imply MaxSNP hardness for our setting. Unfortunately, as we explain next, their claim is not true. There is a quite simple method to find the optimal assignment in the case of a complete graph. Consider only the assignments that pick one item $i \in I$ and assign it to every agent, and select the optimal assignment among them. This can

be done in linear time in the number of items m and the number of agents n . Lemma 13 shows that such an assignment is also optimal among all possible assignments. Assuming $\text{NP} \neq \text{P}$ the incorrectness of Theorem 3.1 in [7] follows.

Lemma 13 *Given convex functions $f_{i,j} : \mathbb{N}_0 \rightarrow \mathbb{R}_{\geq 0}$ with $f_{i,j}(0) = 0$ for all $i \in I := \{1, \dots, m\}$ and all $j \in V$, and a partition of V into m sets $S_1, \dots, S_m$ (i.e., $\bigcup_{i \in I} S_i = V$ and $S_i \cap S_{i'} = \emptyset$ for all $i \neq i' \in I$) it holds that $\sum_{i \in I} \sum_{j \in S_i} f_{i,j}(|S_i|) \leq \max_{i \in I} \sum_{j \in V} f_{i,j}(|V|)$.*

Proof For the proof we set $n = |V|$ and define $i^* \in \text{Arg max}_{i \in I} \sum_{j \in V} f_{i,j}(|V|)$.

$$\sum_{i \in I} \sum_{j \in S_i} f_{i,j}(|S_i|) \leq \sum_{i \in I} \sum_{j \in S_i} f_{i,j}(n) \frac{|S_i|}{n} \quad (13)$$

$$= \sum_{i \in I} \frac{|S_i|}{n} \sum_{j \in S_i} f_{i,j}(n) \quad (14)$$

$$\leq \sum_{i \in I} \frac{|S_i|}{n} \sum_{j \in V} f_{i,j}(n) \quad (15)$$

$$\leq \sum_{i \in I} \frac{|S_i|}{n} \sum_{j \in V} f_{i^*,j}(n) \quad (16)$$

$$= \sum_{j \in V} f_{i^*,j}(n) \sum_{i \in I} \frac{|S_i|}{n} = \sum_{j \in V} f_{i^*,j}(n) \quad (17)$$

Equality (13) follows from the convexity of $f_{i,j}(\cdot)$ and $f_{i,j}(0) = 0$. Equality (14) holds because $\frac{|S_i|}{n}$ does not depend on j . Inequality (15) follows from the non-negativity of $f_{i,j}(\cdot)$. For Inequality (16) observe that for any fixed $i \in I$ it holds that $\sum_{j \in V} f_{i,j}(n) \leq \sum_{j \in V} f_{i^*,j}(n)$ by the definition of i^* . Further, $\sum_{j \in V} f_{i^*,j}(n)$ does not depend on i , which implies the left-hand side of Equality (17). For the right-hand side of Equality (17) recall that $S_1, \dots, S_m$ is a partition of V , and thus, $\sum_{i \in I} |S_i| = |V| = n$. This gives $\sum_{j \in V} f_{i^*,j}(n)$ which is by definition $\max_{i \in I} \sum_{j \in V} f_{i,j}(|V|)$. $\square$

Notice that the valuation of an agent with linear externality is indeed a convex function $f_{i,j}$ with $f_{i,j}(0) = 0$ and in a complete graph the externality of a node only depends on the number of nodes having the same item. Moreover if no one has the item the valuation an agent gets from this item is clearly 0.⁵ Thus the above lemma implies that there is an optimal assignment assigning the same item to all nodes. From that the following theorem follows immediately.

Theorem 11 *The welfare maximization problem for linear externalities and complete graphs can be solved in polynomial time.*

⁵Notice that all these observations hold for both, our definition of linear externalities and the slightly different definition of linear externalities in [7].

However, on general graphs the problem becomes NP-hard. We next show this NP-hardness in a generalized version of Theorem 4. That is we consider p -hop linear externalities ($p \geq 1$), i.e. the *support* of an agent $j \in V$ for item $i \in I$ is defined as $S_{ij}^p(\mathcal{A}, G)$ (see Appendix B). To this end we consider a simpler version of MAX COVERAGE (where all weights w_i are set to 1) which is still NP-hard [13, 27].

Definition 5 The input to the MAX COVERAGE problem is a set of domain elements $D = \{1, 2, \dots, n\}$, a collection $\mathcal{S} = \{E_1, \dots, E_m\}$ of subsets of D and a positive integer k . The goal is to find a collection $\mathcal{S}' \subseteq \mathcal{S}$ of cardinality k maximizing $|\bigcup_{E \in \mathcal{S}'} E|$.

Theorem 4 *The problem of maximizing social welfare under p -hop linear function externalities is NP-hard for $p \geq 1$.*

Proof The proof is by a reduction from the NP-hard problem MAX COVERAGE (see Definition 5). The reduction maps an instance $(D, \mathcal{S}, k)$ of the MAX COVERAGE problem to our problem as follows. It constructs a graph G (cf. Fig. 1) with a node set V containing $X = \{x_1, \dots, x_k\}$, $Y = \{y_1, \dots, y_k\}$, D and $\{e_{i,j}^h \mid i \in D, 1 \leq j \leq k, 1 \leq h < p\}$. The undirected edges are given by

- $\{(x_l, y_l) \mid 1 \leq l \leq k\}$;
- $\{(e_{i,j}^h, e_{i,j}^{h+1}) \mid i \in D, 1 \leq j \leq k, 1 \leq h < p-1\}$; and
- If $p > 1$ we have edges $\{(i, e_{i,j}^1), (e_{i,j}^{p-1}, y_j) \mid i \in D, 1 \leq j \leq k\}$.
Otherwise if $p = 1$ then there are edges $\{(i, y_j) \mid i \in D, 1 \leq j \leq k\}$.

The items are given by $\{E_{i,l} \mid E_i \in \mathcal{S}, 1 \leq l \leq k\}$. Finally the intrinsic values of an agent $j \in D$ for getting item $E_{i,l}$ are given by

$$\lambda_{E_{i,l},j} = \begin{cases} 1 & \text{if } j \in E_i \\ 0 & \text{otherwise.} \end{cases}$$

while the intrinsic values of and agent x_j are given by

$$\lambda_{E_{i,l},x_j} = \begin{cases} n \cdot k \cdot p & \text{if } l = j \\ 0 & \text{otherwise.} \end{cases}$$

and for all other agents j and items i the weight $\lambda_{i,j} = 0$.

Lemma 14 *The above reduction maps each instance of the MAX COVERAGE problem to a network of agents with linear function externalities in p -neighborhoods such that for the value opt_{MC} of the optimal selection for the MAX COVERAGE instance and opt_{SW} the social welfare of the optimal item assignment it holds that $opt_{SW} = k^2 np(1 + (p-1)n) + p \cdot opt_{MC}$.*

Proof (1) We first show that $opt_{SW} \geq k^2 np(1 + (p-1)n) + p \cdot opt_{MC}$: W.l.o.g., we assume that an optimal coverage Opt_{MC} is given by $\{E_1, \dots, E_k\}$. Consider the following assignment $\mathcal{A}$. For each $1 \leq j \leq k$, the nodes x_j, y_j and $\{e_{i,j}^h \mid i \in D, 1 \leq$

Fig. 1 An illustration of the graph constructed in the above reduction for $p = 3$, $D = \{1, 2, 3\}$ and $k = 2$

$h < p\}$ are assigned the item $E_{j,j}$. If an element $d \in D$ is covered by Opt_{MC} , i.e., it is contained in some $E_l \in Opt_{MC}$, then d is assigned the item $E_{l,l}$ (if there are several such E_l pick the one with the lowest index). Otherwise, if d is not covered by Opt_{MC} then item $E_{1,1}$ is assigned to d . Now consider the social welfare of $\mathcal{A}$. First of all, by construction, only the nodes in X and D contribute social welfare. A node $x \in X$ has a p -neighborhood of $1 + (p-1) \cdot n$ many nodes, and all of his p -neighbors get the same item. Thus, x has a valuation of $n \cdot k \cdot p \cdot (1 + (p-1) \cdot n)$. As $|X| = k$ the set X has a social welfare of $n \cdot k^2 \cdot p \cdot (1 + (p-1) \cdot n)$. Now consider $d \in D$. If d is covered by an $E_l \in Opt_{MC}$, i.e., $d \in E_l$, then it was assigned the item $E_{l,l}$. Moreover, $E_{l,l}$ is also assigned to the nodes $\{e_{d,l}^h \mid 1 \leq h < p\}$ and y_l which are in the p -neighborhood of d . Thus each covered $d \in D$ has valuation $\geq p$ and therefore $opt_{SW} \geq k^2 np(1 + (p-1)n) + p \cdot opt_{MC}$.

(2) It remains to show that $opt_{SW} \leq k^2 np(1 + (p-1)n) + p \cdot opt_{MC}$: We first show that in an optimal assignment Opt_{SW} each $x_l \in X$ must have valuation $n \cdot k \cdot p \cdot (1 + (p-1) \cdot n)$, i.e., x_l is assigned to the one of the items $\{E_{i,l} \mid E_i \in S\}$, and $\{e_{i,l}^h \mid i \in D, 1 \leq h < p\}$ and y_l is assigned to the same item. Towards a contradiction assume that there is an assignment $\mathcal{A}$ with one $x \in X$ that has lower valuation. Then the social welfare of X in $\mathcal{A}$ is bounded by $nk^2 p(1 + (p-1)n) - nkp$. Each $d \in D$ has a p -neighborhood of size $k \cdot p$. Thus the social welfare of D in $\mathcal{A}$ is bounded by $n \cdot k \cdot p$. We have that the social welfare of $\mathcal{A}$ is bounded by $nk^2 p(1 + (p-1)n)$. But from (1) we know that $opt_{SW} \geq k^2 np(1 + (p-1)n) + p \cdot opt_{MC}$ where $opt_{MC} \geq 1$ for all none trivial instances. We obtain the desired contradiction. Thus we know that in Opt_{SW} the social welfare of X is given by $n \cdot k^2 \cdot (1 + (p-1) \cdot n)$.

Now consider an $i \in D$. The p -neighborhood of i consists of connected sets $\{e_{i,j}^h \mid 1 \leq h \leq p-1\} \cup \{y_j\}$ for $1 \leq j \leq k$. By the above observation for each set $\{e_{i,j}^h \mid 1 \leq h \leq p-1\} \cup \{y_j\}$ in OPT_{SW} all nodes are assigned the same item, but all sets are assigned different items (as they belong to different x_j). Hence for $i \in D$ the valuation is either p if it gets an item $E_{i,l}$ with $d \in E_i$ and one of its neighbor sets gets the same item. Otherwise d has valuation 0. We obtain that $opt_{SW} = k^2 np(1 + (p-1)n) + p \cdot |\{d \in D \mid d \text{ has positive valuation in } opt_{SW}\}|$. Finally, we can construct an k -covering by choosing the sets E_i corresponding to the items assigned to X . By the construction, this k -covering covers all $d \in D$ with positive valuation and thus $opt_{MC} \geq |\{d \in D \mid d \text{ has positive valuation in } opt_{SW}\}|$. Hence we obtain $opt_{SW} \leq k^2 np(1 + (p-1)n) + p \cdot opt_{MC}$. $\square$

By the above lemma, the presented faithfully reduction maps the decision version of MAX COVERAGE where one ask whether $opt_{MC} \geq C$ to a decision version welfare maximization where one ask whether $opt_{SW} \geq k^2 np(1 + (p-1)n) + p \cdot C$. Moreover it can be also performed in polynomial time and thus Theorem 4 follows from the corresponding result for MAX COVERAGE [13, 27]. $\square$

Appendix D: Generalization of Step-Function Externalities

We can generalize the result in Section 5 as follows (Recall the definitions from Appendix B).

Definition 6 In an s -step externality function in a p -neighborhood for all $j \in V$, $i \in I$ and $A : V \rightarrow I$ it holds that $u_j(A, G) = \lambda_{A(j),j} \cdot f_{A(j),j}(|S_{A(j),j}^p(A, G)|)$ with $f_{ij}(t)$ is 1 if $t \geq s$ and 0 otherwise.

Note that we can assume $p \leq s$ as the p -hop neighborhood cannot be reached with fewer than p nodes. We show next how to generalize the algorithm of the previous setting to s -step externality functions in s neighborhoods. We leave it as an open question to give an approximation algorithm for s -step externality functions in p -neighborhoods for arbitrary $p \leq s$. Moreover the algorithm can be also applied to scenarios where different nodes j have different s_j -step functions. Then the parameter s is set to the maximum of all s_j . The generalized algorithm works as follows: Analogous to triples in Section 5 we compute a maximal collection of mutually disjoint connected sets of size $s+1$ in the graph and decompose the remaining graph into maximal collections of connected sets of size $s, \dots, 1$. The definition of a consistent assignment carries directly over to this new decomposition of the graph and Lemma 9 can be generalized as follows.

Lemma 15 *The social welfare from the optimal consistent assignment is at least $(1 - (\frac{s}{s+1})^s) / s \cdot \text{OPT}$, where OPT is the maximum social welfare over all assignments under s -step function externalities in an s -neighborhood.*

Proof Let $\mathcal{A}^*$ be an assignment (not necessarily consistent) that gives maximum social welfare. We convert it into a (random) consistent assignment $\mathcal{A}$ as follows. First, for each connected component T of size $s + 1$, we pick one of the items $\{\mathcal{A}^*(j) \mid j \in T\}$ uniformly at random, and assign that item to all agents in T . The events corresponding to different connected component are mutually independent. Second, for each connected component T of size $\leq s$, we consider agents $j \in T$ such that there is a adjacent component of size $s + 1$ which was assigned item $\mathcal{A}^*(j)$. Among those agents we select the one with maximal $u_j(\mathcal{A}^*)$ and assign the item $\mathcal{A}^*(j)$ to all agents in T . It is easy to see that the resulting assignment $\mathcal{A}$ is consistent. We claim that $\mathbf{E}[\sum_{j \in T} u_j(\mathcal{A})] \geq \left(1 - \left(\frac{s}{s+1}\right)^s\right) / s \cdot \sum_{j \in T} u_j(\mathcal{A}^*)$ for each connected component T . To prove this claim, we consider two cases.

Case 1 (T is a component of size $s + 1$): Consider an arbitrary $j \in T$. With probability at least $1/(s + 1)$, all these nodes are assigned item $\mathcal{A}^*(j)$ under $\mathcal{A}$, and in this event we have $u_j(\mathcal{A}) \geq u_j(\mathcal{A}^*)$. It follows that $\mathbf{E}[u_j(\mathcal{A})] \geq (1/(s + 1)) \cdot u_j(\mathcal{A}^*)$ and thus also

$$\mathbf{E}\left[\sum_{j \in T} u_j(\mathcal{A})\right] \geq \frac{1}{s + 1} \cdot \sum_{j \in T} u_j(\mathcal{A}^*)$$

Case 2 (T is a component of size $k \leq s$): Consider a component $T = \{j_1, j_2, \dots, j_k\}$ such that $u_j(\mathcal{A}^*)$ is ordered non-increasingly. Let $\mathcal{E}_\ell$ be the event that all nodes in T and all agents of an adjacent $(s + 1)$ -component are assigned item $\mathcal{A}^*(j_\ell)$ under $\mathcal{A}$; then

$$\mathbf{E}\left[\sum_{j \in T} u_j(\mathcal{A})\right] \geq \sum_{\ell=1}^k \mathbf{P}[\mathcal{E}_\ell] \cdot u_{j_\ell}(\mathcal{A}^*)$$

W.l.o.g. assume that $\mathcal{A}^*(j) \neq \mathcal{A}^*(j')$ for $j, j' \in T$, otherwise sum the utilities of the two nodes and consider them as one. Moreover, we assume that $u_j(\mathcal{A}^*) > 0$ for all $j \in T$; that is, we ignore nodes that do not contribute to the social welfare of $\mathcal{A}^*$.

The probability $\mathbf{P}[\mathcal{E}_\ell]$ can be split into two parts. First the probability p_ℓ that item $\mathcal{A}^*(j_\ell)$ is assigned to an adjacent triple under the condition that none of the higher valued items is assigned to an adjacent triple. Second the probability q_ℓ that none of the higher valued items is assigned to an adjacent triple. We then have $\mathbf{P}[\mathcal{E}_\ell] = \mathbf{P}[\mathcal{E}_\ell \cap (\bigcup_{t < \ell} \mathcal{E}_t)^c] = p_\ell \cdot q_\ell$. These probabilities are given as follows.

$$\begin{aligned} p_\ell &= \mathbf{P}[\mathcal{E}_\ell | (\bigcup_{t < \ell} \mathcal{E}_t)^c] \\ q_\ell &= \mathbf{P}[(\bigcup_{t < \ell} \mathcal{E}_t)^c] = 1 - \mathbf{P}[\bigcup_{t < \ell} \mathcal{E}_t] \end{aligned}$$

We can compute q_ℓ with $\ell > 1$ by the following recursion $q_\ell = q_{\ell-1} \cdot (1 - p_{\ell-1})$. Then, as $q_1 = 1$, we obtain that $q_\ell = \prod_{t < \ell} (1 - p_t)$.

Claim 1: It holds that $p_\ell \geq 1/(s + 1)$ for all $1 \leq \ell \leq k$.

Proof As $u_{j_\ell}(\mathcal{A}^*) > 0$ there exists an agent j' in an adjacent $s + 1$ -component T' with $\mathcal{A}^*(j') = \mathcal{A}^*(j_\ell)$. Thus with probability of at least $1/(s + 1)$ all agents in T' get assigned item $\mathcal{A}^*(j_\ell)$ by $\mathcal{A}$. This concludes the proof of Claim 1. $\square$

Using p_ℓ, q_ℓ the lower bound for the expected social welfare of $\mathcal{A}$ can be rewritten as follows.

$$\mathbf{E} \left[\sum_{j \in T} u_j(\mathcal{A}) \right] \geq \sum_{\ell=1}^k p_\ell \cdot q_\ell \cdot u_{j_\ell}(\mathcal{A}^*) = \sum_{\ell=1}^k p_\ell \cdot \prod_{t < \ell} (1 - p_t) \cdot u_{j_\ell}(\mathcal{A}^*) \quad (18)$$

Claim 2: For a lower bound we can assume that $p_\ell = 1/(s+1)$ for all $1 \leq \ell \leq k$.

Proof Towards a contradiction assume there are indices ℓ where $p_\ell > 1/(s+1)$ and setting them to $1/(s+1)$ would not result in a lower bound. Let ℓ' be the largest index of them.

If $\ell' = k$ there is just one summand containing p_k , that is $p_k \cdot \prod_{t < k} (1 - p_t) \cdot u_{j_k}(\mathcal{A}^*)$. This summand is monotone in p_k and thus picking the minimal possible value $1/(s+1)$ (Claim 1), would only decrease the sum and result in a lower bound. Thus we can assume that $\ell' < k$. We consider the summands in which $p_{\ell'}$ occurs in (18) and apply the assumption that p_ℓ equals $1/(s+1)$ for $\ell > \ell'$.

$$\begin{aligned} p_{\ell'} \cdot q_{\ell'} \cdot u_{j_{\ell'}}(\mathcal{A}^*) + \sum_{\ell=\ell'+1}^k p_\ell \cdot \prod_{t < \ell} (1 - p_t) \cdot u_{j_\ell}(\mathcal{A}^*) &= p_{\ell'} \cdot q_{\ell'} \cdot u_{j_{\ell'}}(\mathcal{A}^*) \\ &+ \sum_{\ell=\ell'+1}^k \frac{1}{s+1} \cdot q_{\ell'} \cdot (1 - p_{\ell'}) \prod_{\ell' < t < \ell} (1 - p_t) \cdot u_{j_\ell}(\mathcal{A}^*) \end{aligned}$$

Now, as the $u_{j_\ell}(\mathcal{A}^*)$ are ordered non-increasingly and $k \leq s$ the above sum is monotone in $p_{\ell'}$ and thus can be lower bounded by replacing $p_{\ell'}$ with $1/(s+1)$. This concludes the proof of Claim 2. $\square$

By Claim 2 we obtain the following lower bound.

$$\begin{aligned} \mathbf{E} \left[\sum_{j \in T} u_j(\mathcal{A}) \right] &\geq \sum_{\ell=1}^k \frac{1}{s+1} \left(1 - \frac{1}{s+1} \right)^{\ell-1} \cdot u_{j_\ell}(\mathcal{A}^*) \\ &\geq \sum_{\ell=1}^k \frac{1}{s+1} \left(1 - \frac{1}{s+1} \right)^{\ell-1} \cdot \frac{1}{k} \sum_{j \in T} u_j(\mathcal{A}^*) \\ &= \left(1 - \left(\frac{s}{s+1} \right)^k \right) \cdot \frac{1}{k} \sum_{j \in T} u_j(\mathcal{A}^*) \end{aligned}$$

This concludes the analysis of case 2.

Now, we take the sum of the inequalities for each component, and by linearity of expectation infer that the expected social welfare under the consistent assignment $\mathcal{A}$ is within a factor of

$$\min_{1 \leq k \leq s} \left\{ \left(1 - \left(\frac{s}{s+1} \right)^k \right) \cdot \frac{1}{k} \right\} = \left(1 - \left(\frac{s}{s+1} \right)^s \right) \frac{1}{s}$$

of the optimal social welfare. This concludes the proof of the lemma. $\square$

The remaining arguments carry through without any changes.

Theorem 12 *We have a polynomial-time $\left(1 - \left(\frac{s}{s+1}\right)^s\right) / s \cdot (1 - 1/e)$ -approximation algorithm for maximizing social welfare under s -step function externalities in a s -neighborhood.*

For $s = 1$ this leads to a $\frac{1}{2} \cdot (1 - 1/e)$ -approximation algorithm, improving the $\frac{1}{16} \cdot (1 - 1/e)$ -approximation algorithm of [7].

Towards a simpler approximation ratio we can exploit that $1/2 \leq \left(1 - \left(\frac{s}{s+1}\right)^s\right)$ for $s \geq 1$ and then obtain that our algorithm is a polynomial-time $\frac{1}{2s} \cdot (1 - 1/e)$ -approximation algorithm for maximizing social welfare under s -step function externalities in a s -neighborhood.

Appendix E: Hardness of Approximation

E.1 Proof of Theorem 8

Theorem 8 *For any $\varepsilon > 0$ the problem of maximizing social welfare under arbitrary s -step function externalities is not approximable within $O(n^{1/4-\varepsilon})$ unless $\text{NP} = \text{P}$, and not approximable within $O(n^{1/2-\varepsilon})$ unless $\text{NP} = \text{ZPP}$.*

Proof The proof is by an approximation ratio preserving reduction from the MAX INDEPENDENT SET problem defined below.

Definition 7 The input to the MAX INDEPENDENT SET problem is an (undirected) graph $G = (V, E)$. The goal is to find an independent set, i.e., a set $S \subseteq V$ such that no two vertices in S are incident to the same edge, of maximal cardinality.

Now consider the following reduction from the MAX INDEPENDENT SET instance $G = (V, E)$ to our problem. We first compute the degree δ_j of each node $j \in V$. We then construct the graph $\tilde{G} = (\tilde{V}, \tilde{E})$ as follows:

$$\begin{aligned}\tilde{V} &= V \cup \{(v, v'), (v', v) \mid (v, v') \in E\} \\ \tilde{E} &= \{(v, (v, v')), ((v, v'), (v', v)), ((v', v), v') \mid (v, v') \in E\}\end{aligned}$$

The set of items is given by $I = \{i_j \mid j \in V\}$. and the valuations u_j for $j \in V$ are given by:

$$u_j(\mathcal{A}, \tilde{G}) = \begin{cases} 1 & \text{if } \mathcal{A}(j) = i_j \text{ and } |S_{ij}(\mathcal{A}, \tilde{G})| \geq 2\delta_j \\ 0 & \text{otherwise} \end{cases}$$

That is each agent $j \in V$ just wants one item, the item i_j and only if at least $2\delta_j$ many 1- or 2-hop neighbors in $\tilde{G}$ have the same product. Also for each item there is just one agent that can get revenue from it. For nodes $j \in \tilde{V} \setminus V$ the valuation $u_j(\mathcal{A}, \tilde{G})$ is always zero. Moreover, by construction each node $j \in V$ has exactly $2\delta_j$ many 1-

and 2-hop neighbors in $\tilde{G}$, i.e. for an agent j to get a positive externality all his 1- and 2-hop neighbors have to get the same item. The construction is illustrated in Fig. 2.

Lemma 16 *The above reduction maps each instance of MAX INDEPENDENT SET to a network of agents with step function externalities in 2-neighborhoods such that the cardinality of the maximum independent set is equal to the social welfare of the optimal item assignment.*

Proof We will show that (a) given an independent set of size k we can construct an item assignment with social welfare $\geq k$ and (b) given an item assignment with social welfare k we can construct an independent set of size $\geq k$.

(a) Given an independent set $S \subseteq V$ we define $\mathcal{A}_S$ as follows. Each $j \in S$ and all of its 1- or 2-hop neighbors get item i_j assigned. $\mathcal{A}$ is well-defined as nodes in $j, j' \in S$ are not adjacent in G and thus have a distance of at least 6 in G . Hence no node of $\tilde{V}$ has more than one item assigned. Now, by construction, for each node $j \in V$ we have $u_j(\mathcal{A}, \tilde{G}) = 1$ and thus the social welfare is at least $|S| = k$.

(b) Given an item assignment $\mathcal{A}$ with social welfare k , we consider the set S of nodes $j \in \tilde{V}$ that have positive utility, i.e. $S = \{j \in V \mid u_j(\mathcal{A}, \tilde{G}) = 1\}$. We use that only nodes in V can have positive utility and that the utility of a node is either 0 or 1. As the social welfare is the sum over the utilities we have that $|S| = k$. It remains to show that S is an independent set. Suppose that not, then there are $j, j' \in S$ with $(j, j') \in E$. Thus in $\tilde{G}$ the nodes (j, j') and (j', j) are 2-neighbors of both j and j' . As j has utility 1 it must be assigned item i_j and as its threshold is $2\delta_j$ also all its 2-neighbors must be assigned item i_j , in particular (j, j') , (j', j) must be assigned item i_j . A symmetric argument for j' results that (j, j') , (j', j) must be assigned item $i_{j'}$. As $\mathcal{A}$ can only assign one item to each node this yields the desired contradiction. Hence S is an independent set of size k . $\square$

By the above lemma, the presented reduction is approximation ratio preserving and, it can be also performed in polynomial time. We know that MAX INDEPENDENT SET is not approximable within $|V|^{1/2-\epsilon}$ unless $P = NP$ [22] and not approximable within $|V|^{1-\epsilon}$ unless $ZPP = NP$ [22]. By the above we get the same ratio for our problem but in terms of $|V|$ and not $|\tilde{V}|$. Now as $n = |\tilde{V}| = |V| + 2|E| \leq 3|V|^2$ we get that our problem is not approximable within $O(n^{1/4-\epsilon})$, and $O(n^{1/2-\epsilon})$ respectively. $\square$

Fig. 2 An illustration of the reduction in the proof of Theorem 8

E.2 Proof of Theorem 9

Here we restate and prove Theorem 9 for the more general s -step externality functions in a p -neighborhood (for the formal definitions see Appendix D). These results immediately imply the statement in Section 5 as the notion of 2-step externality functions in a 2-neighborhood coincides with 2-step function externalities from Definition 3.

Theorem 9 *The problem of maximizing social welfare under s -step function externalities in a p -neighborhood is APX-hard (even for $s = p$), that is there is no polynomial-time $1 - \frac{1}{e} + \epsilon$ -approximation algorithm (unless $P=NP$).*

Proof The proof is by an approximation ratio preserving reduction from the MAX COVERAGE problem defined below.

Definition 8 The input to the MAX COVERAGE problem is a set of domain elements $D = \{1, 2, \dots, n\}$, together with non-negative integer weights $w_1, \dots, w_n$, a collection $\mathcal{S} = \{E_1, \dots, E_m\}$ of subsets of D and a positive integer k . The goal is to find a collection $\mathcal{S}' \subseteq \mathcal{S}$ of cardinality k maximizing $\sum_{j \in \bigcup_{E \in \mathcal{S}'} E} w_j$.

Now consider the following reduction from instance $(D, \mathcal{S}, k)$ of the MAX COVERAGE problem to our problem. We construct a bipartite graph G with node set V and an item set $I = \mathcal{S}$, as follows: For each element $j \in D$ we create one node j in V and $s - 1$ nodes g_j^t for $1 \leq t \leq s - 1$. Additionally V contains k nodes e_l with $1 \leq l \leq k$. There is an edge in G from each node j to every node g_j^t for all $1 \leq t \leq s - 1$. Additionally there is an edge from each node j to every node e_l for $1 \leq l \leq k$. We exemplify this construction in Fig. 3. Finally the intrinsic values of an agent $j \in D$ for getting item $E \in \mathcal{S}$ are given by

$$\lambda_{E,j} = \begin{cases} w_j & \text{if } j \in E \\ 0 & \text{otherwise.} \end{cases}$$

while the intrinsic values of an agent $j \notin D$ are given by $\lambda_{E,j} = 0$ for all E . The valuations of the agents for a given assignment are then given by $u_j(\mathcal{A}) = \lambda_{\mathcal{A}(j),j} \cdot f_{\mathcal{A}(j),j}(|S_{\mathcal{A}(j),j}^p(\mathcal{A})|)$, where $f_{i,j}(t)$ are s -step function externalities.

Fig. 3 An illustration of the reduction for 2-step functions and $(D, \mathcal{S}, k) = (\{1, 2, 3, 4, 5\}, \{\{1, 2\}, \{2, 3\}, \{3, 4\}, \{4, 5\}\}, 2)$. Notice that the graph does not depend on the collection $\mathcal{S}$

Lemma 17 *The above reduction maps each instance of the MAX COVERAGE problem to a network of agents with s -step function externalities in p -neighborhoods such that the value of the optimal selection for the MAX COVERAGE problem is equal to the social welfare of the optimal item assignment.*

Proof We will show that (a) given a solution for the MAX COVERAGE instance with value W we can construct an assignment for the s -step function externalities in p -neighborhoods with social welfare at least W and (b) given an assignment for the s -step function externalities in p -neighborhoods with social welfare W we can construct a solution for the MAX COVERAGE instance with value at least W . The lemma then follows.

(a) Consider a feasible solution S' for the MAX COVERAGE instance with value $W = \sum_{j \in \bigcup_{E \in S'} E} w_j$. W.l.o.g. we can that assume $S' = \{E_1, \dots, E_k\}$. One can construct an item assignment to the nodes as follows: For each $1 \leq l \leq k$ assign item E_l to node e_l and for each $j \in \bigcup_{E \in S'} E$ pick one $E_l \in S'$ such that $j \in E$ and assign E_l to j and g_j^t for $1 \leq t \leq s - 1$. Finally, assign arbitrary items to the remaining nodes. It remains to show that the social welfare of this assignment is at least W . To this end consider an arbitrary $j \in \bigcup_{E \in S'} E$. For this j the corresponding node has utility w_j as there are at least s neighbors in his p -neighborhood having the same item. This is because there is a node e_l such that (i) $j \in E_l$ and (ii) e_l, j , and all g_j^t for $1 \leq t \leq s - 1$ get the same item E_l . By construction of the graph j is connected to all these s nodes by an edge. As this holds for each $j \in \bigcup_{E \in S'} E$ the social welfare of the given assignment is at least W .

(b) Given an item assignment $\mathcal{A}$ one can construct a feasible solution for the MAX COVERAGE instance as follows. Consider the nodes $e_1, \dots, e_k$ and the collection S' of assigned items $\{E_{e_1}, \dots, E_{e_k}\}$. Now if we consider the items E_{e_j} again as subsets of D we have that S' is a feasible solution for the MAX COVERAGE instance (as $|S'| \leq k$). Now let D^* be the set of nodes j in the graph with $j \in D$ such that j contributes a positive amount to the social welfare in $\mathcal{A}$. Each node $j \in D^*$ contributes exactly w_j to the social welfare. For each $j \in D^*$ there is at least one node e_l with the same item E_l s.t. $j \in E_l$, as j has only $s - 1$ neighbors that are not e_l nodes and none of these neighbors has further edges. By construction $E_l \in S'$ and thus j is covered by S' and contributes w_j to the value of S' . As this holds for each $j \in D^*$ the value achieved by the feasible solution S' is at least as high as the social welfare of the item assignment $\mathcal{A}$. $\square$

Thus, by the above lemma, the presented reduction is approximation ratio preserving and, as it can be also performed in polynomial time, Theorem 9 follows from the corresponding result for MAX COVERAGE [13, 27]. $\square$

E.3 Proof of Theorem 10

Notice that the reduction in the proof of Theorem 9 introduces an arbitrary large number m of items. So the critical reader might ask whether the problem admits a PTAS if one bounds the number of items. The answer to this question is no as we can show that the problem is still APX-hard if we just consider two items, however for a smaller constant.

Theorem 10 *Maximizing social welfare with s -step function externalities in a p -neighborhood ($p \geq 1$) is APX-hard even for the case with 2 items, i.e., it is NP-hard to approximate better than a factor of $\frac{23}{24}$.*

For showing APX-hardness we make use of a variant of the MAX-SAT problem, where each clause contains either only positive literals or only negative literals. We first have to show that this variant is still APX-hard, using a reduction from MAX3-SAT which is well-known to be APX-hard.

Proposition 1 *Monotone MAX-SAT is APX-hard, i.e., it is NP-hard to approximate it better than a factor of $\frac{23}{24}$, even if we assume that each clause has at most 3 literals and there are exactly as many positive as negative clauses.*

Proof We show the assertion by a reduction from MAX3-SAT for which it is well-known that approximations better than $7/8$ are NP-hard [23]. To this end we consider the following reduction from [17] which shows the decision version of monotone SAT to be NP-hard.

Assume we are given an instance of MAX3-SAT; i.e., a CNF formula $\varphi = \bigwedge_{\gamma \in \Gamma} \bigvee_{l \in \gamma} l$ with Γ being a collection of clauses, where a clause γ is a set of literals of $X = \{x_1, \dots, x_n\}$.

For each $\gamma \in \Gamma$ let γ_p be the set of positive literals in γ and let γ_n be the set of negative literals in γ . We now construct an instance $\psi = \bigwedge_{\gamma \in \widehat{\Gamma}} \gamma$ of Monotone MAX-SAT as follows. For each $\gamma \in \Gamma$, if $\gamma_p = \gamma$ or $\gamma_n = \gamma$ we have $\gamma \in \widehat{\Gamma}$, otherwise $\gamma_p \cup \{z_\gamma\} \in \widehat{\Gamma}$ and $\gamma_n \cup \{\neg z_\gamma\} \in \widehat{\Gamma}$ where z_γ is a new variable. Clearly each of these clauses contains either only positive literals or only negative literals. Further let m be the number of mixed clauses of φ , i.e., clauses that contain both positive and negative literals, and M the maximal number of simultaneously satisfied clauses of φ . $\square$

Claim: The maximal number of simultaneously satisfied clauses in ψ is $M + m$.

Proof (Claim) To show the claim let us first assume we have a truth assignment τ satisfying M clauses of φ . We can extend τ to a truth assignment τ' for the Monotone MAX-SAT as follows. For each mixed clause γ , if τ satisfies a positive literal of γ then we set $\tau'(z_\gamma) = \text{false}$ otherwise $\tau'(z_\gamma) = \text{true}$. Now we have that if τ satisfies clause γ then both corresponding clauses in ψ are satisfied by τ' and if γ is not satisfied by τ then only the positive clause $\gamma_p \cup \{z_\gamma\}$ in ψ is satisfied by τ' . Thus we have that τ' satisfies $M + m$ clauses of ψ .

Now let us assume we have an assignment τ' for ψ satisfying M' clauses. Consider the assignment τ for variables X which is given by $\tau(x) = \tau'(x)$ for all $x \in X$, i.e., τ is the projection of τ' on X . Clearly, monotone clauses of φ are satisfied by τ iff they are satisfied by τ' . Thus consider a mixed clause $\gamma \in \Gamma$ where both corresponding clauses in ψ are true. At most one of these clauses can be satisfied by the variable z_γ and thus γ must be satisfied by τ . Finally, observe that there are only m mixed

clauses and thus at least $M' - m$ clauses of φ are satisfied, which concludes the proof of the claim. $\square$

We next show that if we could $(\frac{23}{24} + \epsilon)$ -approximate Monotone MAX-SAT then we could also $(7/8 + \epsilon)$ -approximate MAX3-SAT. Given a MAX3-SAT instance φ and ψ constructed from φ as above, let M be the maximal number of simultaneously satisfied clauses of φ , and let M' be the maximal number of simultaneously satisfied clauses of ψ . Towards a contradiction assume there exists a $(\frac{23}{24} + \epsilon)$ -approximation algorithm for Monotone MAX-SAT. Then, by the above observation, we can construct in polynomial time an assignment for φ that satisfies $a' - m$ many clauses, where a' is the number of clauses of ψ satisfied by the algorithm.

$$\begin{aligned} a' - m &\geq (\frac{23}{24} + \epsilon)M' - m = (\frac{23}{24} + \epsilon)(M + m) - m \\ &\geq (\frac{23}{24} + \epsilon)M - (\frac{1}{24} - \epsilon)m \\ &\geq (\frac{23}{24} + \epsilon)M - (\frac{1}{24} - \epsilon)2M = (\frac{7}{8} + 3\epsilon) \cdot M \end{aligned}$$

For the last inequality we used the fact that in each CNF-formula at least a half of the clauses can be satisfied, i.e., $m \leq |\Gamma| \leq 2M$. By the above, any $(\frac{23}{24} + \epsilon)$ -approximation algorithm for Monotone MAX-SAT results in a $(\frac{7}{8} + 3\epsilon)$ -approximation of the MAX3-SAT instance, a contradiction.

Now notice that by construction all clauses of $\widehat{\Gamma}$ are either of size 2 or 3. Moreover, for each monotone formula $\psi(X)$ we can construct an formula ψ^* such that

- (i) ψ^* has as many positive as negative clauses and
- (ii) the number of simultaneously satisfiable clauses in ψ^* is exactly twice the number of simultaneously satisfiable clauses of $\psi(X)$.

To this end we pick a set of fresh variables $Y = \{y_1, \dots, y_n\}$ and construct a formula $\psi'(Y)$ by replacing each literal x_i in $\psi(X)$ by $\neg y_i$ and each literal $\neg x_i$ in $\psi(X)$ by y_i . Thus each positive clause becomes negative and vice versa while the number of simultaneously satisfiable clauses is unchanged. Finally we define $\psi^*(X \cup Y) = \psi(X) \wedge \psi'(Y)$.

Proof (Theorem 10) We show hardness by the following reduction from monotone MAX-SAT. Assume we are given an instance of monotone MAX-SAT; i.e., a CNF formula $\varphi = \bigwedge_{\gamma \in \Gamma \cup \bar{\Gamma}} \gamma$ with Γ being a set of clauses over variables X containing only positive literals and $\bar{\Gamma}$ being a set of clauses over variables X containing only negative literals. We construct the following network. The agents are given by

$$V = X \cup \Gamma \cup \bar{\Gamma} \cup \bigcup_{i=1}^{s-1} \Gamma^i \cup \bigcup_{i=1}^{s-1} \bar{\Gamma}^i,$$

where Γ^i and $\bar{\Gamma}^i$ are copies of Γ and $\bar{\Gamma}$, and there are only two items T and F . The (undirected) edges of G are given by

- (i) (γ, γ^i) for each $\gamma \in \Gamma \cup \bar{\Gamma}$ and copies $\gamma^i \in \Gamma^i \cup \bar{\Gamma}^i$, and

Fig. 4 An illustration of the reduction in the proof of Theorem 10 for $(x_1 \vee x_2 \vee x_3) \wedge (\neg x_2 \vee \neg x_3 \vee \neg x_4)$ and $s = 2$

- (ii) (x, γ) for each γ and variable x with $x \in \gamma$ or $\neg x \in \gamma$.

The constructed graph is exemplified in Fig. 4.

Next consider the intrinsic values $\lambda_{i,j}$ in the valuation functions: For each $\gamma \in \Gamma$ set $\lambda_{T,\gamma} = 1$; for each $\gamma \in \bar{\Gamma}$ set $\lambda_{F,\gamma} = 1$; and $\lambda_{i,j} = 0$ for all other combinations of $i \in \{T, F\}$ and $j \in V$. The valuations of the agents for a given assignment are then given by $u_j(\mathcal{A}) = \lambda_{\mathcal{A}(j),j} \cdot f_{\mathcal{A}(j),j}(|S_{\mathcal{A}(j),j}^p(\mathcal{A})|)$, where $f_{i,j}(\mathcal{A})$ are s -step function externalities.

Next we show that the maximal number of simultaneously satisfied clauses of φ is exactly the maximal social welfare one can obtain in the network.

- Consider a truth assignment τ for φ and construct the following item assignment.
 - Agent $x \in X$ gets item T if $\tau(x) = \text{true}$ and $x \in X$ gets item F if $\tau(x) = \text{false}$.
 - For $\gamma \in \Gamma$ agent γ and γ 's copies γ^i get item T , and for $\gamma \in \bar{\Gamma}$ agents γ and γ 's copies γ^i get item F . Consider a clause $\gamma \in \Gamma$ satisfied by τ via literal x . Then the valuation of agent γ is one as he has s neighbors, $\gamma^1, \dots, \gamma^{s-1}$ and x , that also get item T . Now consider a clause $\gamma \in \bar{\Gamma}$ satisfied by τ via literal $\neg x$. Then the valuation of agent γ is one as he has s neighbors, $\gamma^1, \dots, \gamma^{s-1}$ and x , that also get item F . Thus for each satisfied clause we achieve a valuation of one.
- Consider an item assignment and the truth assignment τ defined by $\tau(x) = \text{true}$ iff agent x gets item T and $\tau(x) = \text{false}$ iff agent x gets item F . Notice that we can only achieve positive valuation from agents $\gamma \in \Gamma \cup \bar{\Gamma}$. First consider an agent $\gamma \in \Gamma$ that contributes to the social welfare. Then, as γ has only $s - 1$ neighbors not in X and those neighbors have no neighbors except γ , there has to exist an $x \in \gamma$ such that agent x gets item T . By definition of τ we then have that τ satisfies the clause γ . Second consider an agent $\gamma \in \bar{\Gamma}$ that contributes to the social welfare. Then, by a similar argument, there has to exist an x with $\neg x \in \gamma$ such that agent x gets item F . Again by the definition of τ we then have that τ satisfies the clause γ . Thus, we have that for each agent that contributes to the social welfare we satisfy one clause and as each of these agent contributes exactly 1 the claim follows.

Now as we have a one-to-one correspondence between the number of satisfied clauses and the social welfare, not only NP-hardness but also hardness of approximation follows from the corresponding result for monotone MAX3-SAT. $\square$

E.4 Proof of Theorem 5

We can further exploit Proposition 1 to prove Theorem 5.

Theorem 5 *Maximizing social welfare with linear externalities in a 2-neighborhood is APX-hard even for the case with 2 items, i.e., it is NP-hard to approximate better than a factor of $\frac{143}{144}$.*

Proof We show hardness by a reduction from monotone MAX-SAT, where we assume the formula has only clauses of size ≤ 3 and exactly as many positive as negative clauses. The reduction proceeds as follows.

Assume we are given an instance of monotone MAX-SAT; i.e., a CNF formula $\varphi = \bigwedge_{\gamma \in \Gamma \cup \bar{\Gamma}} \gamma$ with Γ being a set of clauses over variables X containing only positive literals and $\bar{\Gamma}$ being a set of clauses over variables X containing only negative literals. Moreover, $|\Gamma| = |\bar{\Gamma}| = m$. The reduction constructs a graph $G = (V, E)$ with

$$\begin{aligned} V &= X \cup \Gamma \cup \bar{\Gamma} \cup \{t_i, f_i \mid 1 \leq i \leq 7m+1\} \cup \{t, t', f, f', C, \bar{C}\} \\ E &= \{(t, t'), (f, f')\} \cup \{(t', t_i), (f', f_i), (x, t_i), (x, f_i) \mid 1 \leq i \leq 7m+1, x \in X\} \cup \\ &\quad \{(x, \gamma) \mid x \in \gamma \in \Gamma\} \cup \{(x, \gamma) \mid \neg x \in \gamma \in \bar{\Gamma}\} \cup \{(C, \gamma) \mid \gamma \in \Gamma\} \cup \{(\bar{C}, \gamma) \mid \gamma \in \bar{\Gamma}\} \end{aligned}$$

The resulting graph is illustrated in Fig. 5 (for $\varphi = (x_1 \vee x_2 \vee x_3) \wedge (\neg x_2 \vee \neg x_3 \vee \neg x_4)$).

Next consider the items and the intrinsic values $\lambda_{i,j}$ in the valuation functions. As items we consider T, F and for each $\gamma \in \Gamma$ we set $\lambda_{T,\gamma} = 1$ while for each $\gamma \in \bar{\Gamma}$ we set $\lambda_{F,\gamma} = 1$. Moreover, $\lambda_{T,t} = 2m+1$ and $\lambda_{F,f} = 2m+1$. Finally, we set $\lambda_{i,j} = 0$ for all other combinations of $i \in \{T, F\}$ and $j \in V$.

We next correlate the maximal number M of simultaneously satisfied clauses of φ to the maximal social welfare W one can obtain in the network.

$$W \geq (7m+2)M + 2m^2 + 2(2m+1)(7m+2) \quad (19)$$

$$M = \left\lfloor \frac{W - 2m^2}{7m+2} - 2(2m+1) \right\rfloor \quad (20)$$

Claim 1: $W \geq (7m+2)M + 2m^2 + 2(2m+1)(7m+2)$

Proof Towards $W \geq (7m+2)M + 2m^2 + 2(2m+1)(7m+2)$ consider an optimal truth assignment α for φ . We build an assignment $\mathcal{A}$ of items to agents as follows.

$$\mathcal{A}(v) = \begin{cases} T & \text{if } v \in \{t, t', C\} \cup \{t_i \mid 1 \leq i \leq 7m+1\} \cup \Gamma \\ F & \text{if } v \in \{f, f', \bar{C}\} \cup \{f_i \mid 1 \leq i \leq 7m+1\} \cup \bar{\Gamma} \\ T & \text{if } v \in X \text{ and } \alpha(v) = 1 \\ F & \text{if } v \in X \text{ and } \alpha(v) = 0 \end{cases}$$

Notice, that only the item assignments for $v \in X$ actually depend on α .

Fig. 5 Graph constructed in the proof of Theorem 5 for $\varphi = (x_1 \vee x_2 \vee x_3) \wedge (\neg x_2 \vee \neg x_3 \vee \neg x_4)$

Now consider the social welfare of this assignment $\mathcal{A}$. Only nodes t, f and $\gamma \in \Gamma \cup \bar{\Gamma}$ can contribute to the social welfare of $\mathcal{A}$. First, we have that t and $7m + 2$ of its 2-hop neighbors, i.e. t' and t_i for $1 \leq i \leq 7m + 1$, got item T . Thus t contributes $(2m + 1)(7m + 2)$ to the social welfare. Similarly, we get that f also contributes $(2m + 1)(7m + 2)$ to the social welfare.

For $\gamma \in \Gamma$ we distinguish the cases whether α satisfies γ or not. In the former case at least one adjacent $x \in X$ gets the item T and thus connects γ to the t_i . Moreover all the other nodes in Γ are connected via C and get item T . Hence, the node γ then has at least $(7m + 2) + (m)$ many 2-hop neighbors with item T and thus contributes $(7m + 2) + (m)$ to the social welfare. On the other hand if α does not satisfy γ then all adjacent $x \in X$ get the item F and thus γ is not connected to any t_i . Still all the other nodes in Γ are connected via node C and get item T and thus γ contributes m .

With a symmetric argument we obtain that for $\gamma \in \bar{\Gamma}$ we get at least utility $(7m + 2) + m$ if α satisfies γ and utility m otherwise.

Finally, by summing up the valuations we obtain that the social welfare of $\mathcal{A}$ is at least $(7m + 2)M + 2m^2 + 2(2m + 1)(7m + 2)$ $\square$

$$\text{Claim 2: } M = \left\lfloor \frac{W - 2m^2}{7m + 2} - 2(2m + 1) \right\rfloor$$

Proof Consider an optimal assignment $\mathcal{A}$. We first show that we can assume the following

1. $\mathcal{A}(v) = T$ for $v \in \{t, t', C\}$ and $\mathcal{A}(v) = F$ for $v \in \{f, f', \bar{C}\}$
2. $\mathcal{A}(v) = T$ for $v \in \{t_i \mid 1 \leq i \leq 7m + 1\}$ and $\mathcal{A}(v) = F$ for $\{f_i \mid 1 \leq i \leq 7m + 1\}$
3. $\mathcal{A}(v) = T$ for $v \in \Gamma$ and $\mathcal{A}(v) = F$ for $v \in \bar{\Gamma}$

For all three statements we only show the case $\mathcal{A}(v) = T$ the case $\mathcal{A}(v) = F$ then follows by symmetry.

1. Only nodes t , f and $\gamma \in \Gamma \cup \bar{\Gamma}$ can contribute to the social welfare. As t is not in the 2-hop neighborhood of any other of these nodes we can assign T to it without decreasing social welfare. Moreover, t' , C are only in the neighborhoods of nodes that only have positive utility for item T so we can also assign T to t' , C without decreasing the social welfare.
2. By the above we can already assume that $\mathcal{A}(t) = T$. Then setting $\mathcal{A}(t_i) = T$ (for some $1 \leq i \leq 7m + 1$) contributes at least $2m + 1$ to the social welfare. On the other hand setting $\mathcal{A}(t_i) = F$ can only contribute at nodes $\gamma \in \bar{\Gamma}$ and thus at most m . Hence, we can assign T to t_i without decreasing the social welfare.
3. We already have $\mathcal{A}(t_i) = T$. Assume there are k nodes in Γ that $\mathcal{A}$ assigns to F . These nodes themselves have utility 0 and only contribute to the social welfare if they are connected to nodes in $\bar{\Gamma}$. Thus their individual contributions to the social welfare are bounded by m . So when setting $\mathcal{A}(\gamma) = T$ for all $\gamma \in \Gamma$ these k nodes are connected to C and all other nodes in Γ and thus each of them contributes at least m to the social welfare. Hence, we can assign T to all $\gamma \in \Gamma$ without decreasing the social welfare.

Now, given an assignment $\mathcal{A}$ with the above properties the social welfare of $\mathcal{A}$ is composed as follows: (i) t and f both have utility $(2m + 1)(7m + 2)$; (ii) a $\gamma \in \Gamma \cup \bar{\Gamma}$ either contributes utility between $8m + 2$ and $8m + 4$, or utility m . The former is simply by the fact that t and f have $7m + 2$ many 2-hop neighbors with the same item. The latter is by the fact that if $\gamma \in \Gamma$ is connected to the t_i then it has at least $8m + 2$ many 2-hop neighbors with the same item and as each clause has at most 3 literals it can have at most $8m + 4$ such neighbors. On the other hand if t is not connected to t_i it has still m 2-hop neighbors (Γ and C) with the same item. The same argument with $\bar{\Gamma}$, $\bar{C}$ and f_i applies to $\gamma \in \bar{\Gamma}$. Now we can define a truth assignment α as follows.

$$\alpha(x) = \begin{cases} 1 & \text{if } \mathcal{A}(x) = T \\ 0 & \text{if } \mathcal{A}(x) = F \end{cases}$$

By construction a clause γ is satisfied by α iff the node γ has utility $\geq 8m + 2$. Moreover a clause γ is not satisfied by α iff the node γ has utility m . Now to obtain the number of satisfied clauses we start from W and subtract the utility of t and f . As each clause has at least utility m and there a $2m$ many clauses we next subtract $2m^2$ and finally divide by $7m + 2$ to obtain the number of satisfied clauses. Thus, the number of satisfied clauses is given by $\left\lfloor \frac{W - 2(2m + 1)(7m + 2) - 2m^2}{7m + 2} \right\rfloor$. Notice that for a satisfied clause we do not know the exact utility but for each clause it only varies by 2. So in total this effect is bounded by $4m$ and eliminated by dividing with $7m + 2$ and rounding down. $\square$

Now, towards a contradiction, assume there is an $(1 - \epsilon)$ approximation for the social welfare problem and consider $m \geq 7/\epsilon$. Let A be the social welfare of a $(1 - \epsilon)$ approximation of W , that is

$$A \geq (1 - \epsilon)W \geq (1 - \epsilon) \left((7m + 2)M + 2m^2 + 2(2m + 1)(7m + 2) \right). \quad (21)$$

From our earlier observations we know that $M \geq \left\lfloor \frac{A-2m^2}{7m+2} - 2(2m+1) \right\rfloor$. Below we show that also $\left\lfloor \frac{A-2m^2}{7m+2} - 2(2m+1) \right\rfloor \geq (1-6\epsilon)M$.

$$\begin{aligned} \frac{A-2m^2}{7m+2} - 2(2m+1) &\geq (1-\epsilon)M - \epsilon \left(2m^2/(7m+2) + 4m+2 \right) \\ &\geq (1-\epsilon)M - \epsilon (2Mm/(7m+2) + 4M+2) \\ &\geq (1-\epsilon)M - \epsilon (2/7M + 4M+2) \end{aligned}$$

We first used (21) and then the fact that $M \geq m$.

Next we add the rounding to the left side. This rounding decreases the value by at most 1 and we then use $M \geq m$ and the assumption $m \geq 7/\epsilon$

$$\begin{aligned} \left\lfloor \frac{A-2m^2}{7m+2} - 2(2m+1) \right\rfloor &\geq (1-\epsilon)M - \epsilon (2/7M + 4M+2) - 1 \\ &\geq (1-\epsilon)M - \epsilon (2/7M + 4M+2) - \epsilon(M/7) \\ &\geq (1-6\epsilon)M \end{aligned}$$

We now have that social welfare optimization with linear externalities is APX-hard, in particular if we set $6\epsilon = 1/24$ (in accordance with Proposition 1) we get that it is NP-hard to approximate better than $143/144$. $\square$

References

1. Akhlaghpour, H., Ghodsi, M., Haghpahan, N., Mirrokni, V.S., Mahini, H., Nikzad, A.: Optimal Iterative Pricing over Social Networks. In: 6Th WINE, pp. 415–423 (2010)
2. Alon, N., Feldman, M., Procaccia, A.D., Tennenholtz, M.: A note on competitive diffusion through social networks. Inf. Process. Lett. **110**(6), 221–225 (2010)
3. Anari, N., Ehsani, S., Ghodsi, M., Haghpahan, N., Immorlica, N., Mahini, H., Mirrokni, V.S.: Equilibrium pricing with positive externalities. Theor. Comput. Sci. **476**, 1–15 (2013)
4. Anshelevich, E., Bhardwaj, O., Usher, M.: Friend of my friend: Network formation with two-hop benefit. In: Vöcking, B. (ed.) Algorithmic Game Theory - 6th International Symposium, SAGT 2013, Aachen, Germany, October 21–23, 2013. Proceedings, volume 8146 of Lecture Notes in Computer Science, pp. 62–73. Springer (2013)
5. Arthur, D., Motwani, R., Sharma, A., Ying, X.: Pricing Strategies for Viral Marketing on Social Networks. In: 5Th WINE, pp. 101–112 (2009)
6. Bensaid, B., Lesne, J.-P.: Dynamic monopoly pricing with network externalities. Int. J. of Industrial Organization **14**(6), 837–855 (1996)
7. Bhargat, A., Gollapudi, S., Munagala, K.: Mechanisms and Allocations with Positive Network Externalities. In: 13Th EC, pp. 179–196 (2012)
8. Bhattacharya, S., Dvořák, W., Henzinger, M., Starnberger, M.: Welfare Maximization with Friends-of-Friends Network Externalities. In: Mayr, E.W., Ollinger, N. (eds.) 32nd International Symposium on Theoretical Aspects of Computer Science (STACS 2015), volume 30 of Leibniz International Proceedings in Informatics (LIPIcs), pp. 90–102, Dagstuhl, Germany, 2015. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik
9. Bhattacharya, S., Korzyk, D., Conitzer, V.: Computing a Profit-Maximizing Sequence of Offers to Agents in a Social Network. In: 8Th WINE, pp. 482–488 (2012)
10. Cigler, L., Dvořák, W., Henzinger, M., Starnberger, M.: Limiting price discrimination when selling products with positive network externalities. In: Tie-Yan Liu, Qi Qi, and Yinyu Ye, editors, Web and

- Internet Economics - 10th International Conference, WINE 2014, Beijing, China, December 14–17, 2014. Proceedings, volume 8877 of Lecture Notes in Computer Science, pp. 44–57. Springer (2014)
11. Dobzinski, S., Nisan, N., Schapira, M.: Approximation algorithms for combinatorial auctions with complement-free bidders. *Math. Oper Res.* **35**(1), 1–13 (2010)
 12. Dubey, P., Garg, R., Meyer, B.D.: Competing for Customers in a Social Network: The Quasi-Linear Case. In: 2Nd WINE, pp. 162–173 (2006)
 13. Feige, U.: A threshold of $\ln n$ for approximating set cover. *J. ACM* **45**(4), 634–652 (1998)
 14. Feige, U.: On maximizing welfare when utility functions are subadditive. *SIAM J. Comput.* **39**(1), 122–142 (2009)
 15. Feld, S.L.: Why your friends have more friends than you do. *American J. of Sociology* **96**(6), 1464–1477 (1991)
 16. Fotakis, D., Siminelakis, P.: On the Efficiency of Influence-And-Exploit Strategies for Revenue Maximization under Positive Externalities. In: 8Th WINE, pp. 270–283 (2012)
 17. Mark Gold, E.: Complexity of automaton identification from given data. *Inf. Control.* **37**(3), 302–320 (1978)
 18. Goyal, S., Kearns, M.: Competitive Contagion in Networks. In: 44Th STOC, pp. 759–774 (2012)
 19. Haghpahan, N., Immorlica, N., Mirrokni, V.S., Munagala, K.: Optimal auctions with positive network externalities. *ACM Trans. Economics and Comput.* **1**(2), 13:1–13:24 (2013)
 20. Hampton, K.N., Goulet, L.S., Marlow, C., Rainie, L.: Why most facebook users get more than they give Pew Internet & American Life Project (2012)
 21. Hartline, J., Mirrokni, V.S., Sundararajan, M.: Optimal Marketing Strategies over Social Networks. In: 17Th WWW, pp. 189–198 (2008)
 22. Håstad, J.: Clique is hard to approximate within $n^{1-\epsilon}$. *Electronic Colloquium on Computational Complexity (ECCC)*, 4(38) (1997)
 23. Håstad, J.: Some optimal inapproximability results. *J. ACM* **48**(4), 798–859 (2001)
 24. He, X., Kempe, D.: Price of Anarchy for the N-Player Competitive Cascade Game with Submodular Activation Functions. In: 9Th WINE, pp. 232–248 (2013)
 25. Jackson, M.O., Rogers, B.W.: Meeting strangers and friends of friends: How random are social networks? *Am. Econ. Rev.* **97**(3), 890–915 (2007)
 26. Kempe, D., Kleinberg, J.M., Tardos, É.: Maximizing the Spread of Influence through a Social Network. In: 9Th KDD, pp. 137–146 (2003)
 27. Khuller, S., Moss, A., Naor, J.: The budgeted maximum coverage problem. *Inf. Process. Lett.* **70**(1), 39–45 (1999)
 28. Mirrokni, V.S., Roch, S., Sundararajan, M.: On Fixed-Price Marketing for Goods with Positive Network Externalities. In: 8Th WINE, pp. 532–538 (2012)
 29. Seeman, L., Singer, Y.: Adaptive Seeding in Social Networks. In: 54Th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26–29 October, 2013, Berkeley, CA, USA, pages 459–468. IEEE Computer Society (2013)
 30. Simon, S., Apt, K.R.: Choosing Products in Social Networks. In: 8Th WINE, pp. 100–113 (2012)
 31. Takehara, R., Hachimori, M., Maiko, S.: A comment on pure-strategy nash equilibria in competitive diffusion games. *Inf Process. Lett.* **112**(3), 59–60 (2012)
 32. Tzoumas, V., Amanatidis, C., Markakis, E.: A Game-Theoretic Analysis of a Competitive Diffusion Process over Social Networks. In: 8Th WINE, pp. 1–14 (2012)