Noname manuscript No.
(will be inserted by the editor)

Exact and Approximate Equilibria for Optimal Group Network Formation

Elliot Anshelevich · Bugra Caskurlu

Received: December 2009 / Accepted:

Abstract We consider a process called Group Network Formation Game, which represents the scenario when strategic agents are building a network together. In our game, agents can have extremely varied connectivity requirements, and attempt to satisfy those requirements by purchasing links in the network. We show a variety of results about equilibrium properties in such games, including the fact that the price of stability is 1 when all nodes in the network are owned by players, and that doubling the number of players creates an equilibrium as good as the optimum centralized solution. For the general case, we show the existence of a 2-approximate Nash equilibrium that is as good as the centralized optimum solution, as well as how to compute good approximate equilibria in polynomial time. Our results essentially imply that for a variety of connectivity requirements, giving agents more freedom can paradoxically result in more efficient outcomes.

1 Introduction

Many modern computer networks, including the Internet itself, are constructed and maintained by self-interested agents. This makes network design a fundamental problem for which it is important to understand the effects of strategic behavior. Modeling and understanding of the evolution of nonphysical networks created by many heterogonous agents (like social networks, viral networks, etc.) as well as physical networks (like computer networks, transportation networks, etc.) has been studied extensively in the last several years. In networks constructed by several self-interested agents, the global performance of the system may not be as good as in the case where a central authority can simply dictate a solution; rather, we need to understand the quality of solutions that are consistent with self-interested behavior. Much research in the theoretical computer science community has focused on this performance gap and specifically on the notions of the *price of anarchy* and the *price of stability* — the ratios between the costs of the worst and best Nash equilibrium[1], respectively, and that of the globally optimal solution.

In this paper, we study a network design game that we call the *Group Network Formation Game*, which captures the essence of strategic agents building a network together. In this game, players correspond to

A preliminary version of this paper appeared in ESA 2009 and this research supported in part by NSF CCF-0914782.

Elliot Anshelevich
Rensselaer Polytechnic Institute, Troy, NY.
Tel.: +1-518-2766491
Fax: +1-518-2764033
E-mail: eanshel@cs.rpi.edu

Bugra Caskurlu
Rensselaer Polytechnic Institute, Troy, NY.
Tel.: +1-518-2768988
Fax: +1-518-2764033
E-mail: caskub@cs.rpi.edu

[1] Recall that a (pure-strategy) Nash equilibrium is a solution where no single player can switch her strategy and become better off, given that the other players keep their strategies fixed.

nodes of a graph (although not all nodes need to correspond to players), and the players can have extremely varied connectivity requirements. For example, there might be several different "types" of nodes in the graph, and a player desires to connect to at least one of every type (so that this player's connected component forms a Group Steiner Tree [13]). Or instead, a player might want to connect to at least k other player nodes. The first example above is useful for many applications where a set of players attempt to form groups with "complementary" qualities. The second example corresponds to a network of servers where each server want to be connected to at least k other servers so that it can have a backup of its data; or in the context of IP networks, a set of ISPs that want to increase the reliability of the Internet connection for their customers, and so decide to form multi-homing connections through k other ISPs [25]. Many other types of connectivity requirements fit into our framework, and so the results we give in this paper will be relevant to many different types of network problems.

Game Definition We now formally define the *Group Network Formation Game* as follows. Let an undirected graph $G = (V, E)$ be given, with each edge e having a nonnegative cost $c(e)$. This graph represents the possible edges that can be built. Each player i corresponds to a single node in this graph (that we call a *player* or *terminal* node), which we will also denote by i. Similarly to [4], a strategy of player i is a payment vector p_i of size $|E|$, where $p_i(e)$ is how much player i is offering to contribute to the cost of edge e. We say that an edge e is *bought*, i.e., it is included in the network, if the sum of payments of all the players for e is at least as much as the cost of e ($\sum_i p_i(e) \geq c(e)$). Let G_p denote the subgraph of bought edges corresponding to the strategy vector $p = (p_1, \ldots, p_N)$. G_p is the outcome of this game, since it is the network which is purchased by the players.

To define the utilities/costs of the players, we must consider their connectivity requirements. *Group Network Formation Game* considers the class of problems where the players' connectivity requirements can be compactly represented with a function $F : 2^U \to \{0, 1\}$, where $U \subseteq V$ is the set of player nodes, similar to [14]. This function F has the following meaning. If S is a set of terminals, then $F(S) = 1$ if and only if the connectivity requirements of all players in S would be satisfied if S is the set of terminals of a connected component in G_p. For the example above, where each player wants to connect to at least one player from each "type", the function $F(S)$ would evaluate to 1 exactly when S contains at least one player of each type. Similarly, for the "data backup" example above, the function $F(S)$ would evaluate to 1 exactly when S contains at least $k + 1$ players. In general, we will assume that the connectivity requirements of the players are represented by a monotone "happiness" function F. The monotonicity of F means that if the connectivity requirement of a player is satisfied in a graph G_p, then it is still satisfied when this player is connected to strictly more nodes. We will call a set of player nodes S a "happy" group if $F(S) = 1$. While not all connectivity requirements can be represented as such a function, it is a reasonably general class that includes the examples given above. Therefore an instance of our game consists of a graph $G = (V, E)$, player nodes $U \subseteq V$, and a function F that states the connectivity requirements of the players. We will say that player i's connectivity requirement is *satisfied* in G_p if and only if $F(S_i(G_p)) = 1$ for $S_i(G_p)$ being the terminals of i's connected component in G_p. While required to connect to a set of terminal nodes satisfying her connectivity requirement, each player also tries to minimize her total payments, $\sum_{e \in E} p_i(e)$ (which we will denote by $|p_i|$). We conclude the definition of our game by defining the cost function for each player i as:

- $cost(i) = \infty$ if $F(S_i(G_p)) = 0$
- $cost(i) = |p_i|$ otherwise.

In our game, all players want to be a part of a happy group which can correspond to many connectivity requirements, some of which are mentioned above. The socially optimal solution (which we denote by OPT) for this game is the cheapest possible network where every connected component is a happy group, since this is the solution maximizing social welfare[2]. For our first example above, OPT corresponds to the cheapest forest where every component is a Group Steiner Tree, for the second to the Terminal Backup problem [5], and in general it can correspond to a variety of constrained forest problems [14]. Our goals include understanding the quality of exact and approximate Nash equilibria by comparing them to OPT, and thereby understanding the efficiency gap that results because of the players' self-interest. By studying the price of stability, we also

[2] The solution that maximizes the social welfare is the one that minimizes the total cost of all the players.

seek to reduce this gap, as the best Nash equilibrium can be thought of as the best outcome possible if we were able to suggest a solution to all the players simultaneously.

In the *Group Network Formation Game*, we don't assume the existence of a central authority that designs and maintains the network, and decides on appropriate cost-shares for each player. Instead we use a cost-sharing scheme which is sometimes referred to as "arbitrary cost sharing" [4,10] that permits the players to specify the actual amount of payment for each edge. This cost-sharing mechanism is necessary in scenarios where very little control over the players is available, and gives more freedom to players in specifying their strategies, i.e., has a much larger strategy space. The main advantage of such a model is that the players have more freedom in their choices, and less control is required over them. A disadvantage of such a system, however, is that it does not guarantee the existence of Nash equilibria (unlike more constrained systems such as fair sharing [3]). Studying the existence of Nash equilibria under arbitrary cost sharing has been an interesting research problem and researchers have proven existence for many important games [2,4,10,15, 16]. Interestingly, in many of these problems it has been shown that the equilibrium is indeed cheap, i.e., costs as much as the socially optimal network. As we show in this paper, this tells us that in the network design contexts we consider, *arbitrary sharing produces more efficient outcomes while giving the players more freedom.*

Related Work Over the last few years, there have been several new papers using arbitrary cost-sharing, e.g., [2,10,15,17]. Recently, Hoefer [16] proved some interesting results for a generalization of the game in [4], and considered arbitrary sharing in variants of Facility Location.

Unquestionably one of the most important decisions when modeling network design involving strategic agents is to determine how the total cost of the solution is going to be split among the players. Among various alternatives [8], the "fair sharing" mechanism is the most relevant to ours [3,6,7,12]. In this cost sharing mechanism, the cost of each edge of the network is shared equally by the players using that edge. This model has received much attention, mostly because of the following three reasons. Firstly, it nicely quantifies what people mean by "fair" and has an excellent economic motivation since it is strongly related to the concept of Shapley value[3]. Secondly, fair sharing naturally models the congestion effects of network routing games, and so network design games with fair sharing fall into the well-studied class of "congestion games" [6,9,18,24]. Thirdly, this model has many attractive mathematical properties including guarantees on the existence of Nash equilibrium that can be obtained by natural game playing [3].

Despite all of the advantages of congestion games mentioned above, there are extremely important disadvantages as well. Firstly, although congestion games are guaranteed to have Nash equilibria, these equilibria may be very expensive. Anshelevich et al. [3] showed that the cheapest Nash equilibrium solution can be $\Omega(\log n)$ times more expensive than OPT in network games with fair sharing, and that this bound is tight. As we prove in this paper, arbitrary cost-sharing will often guarantee the existence of Nash equilibria that are as cheap as the socially optimal solution. Secondly, fair sharing inherently assumes the existence of a central authority that regulates the agent interactions or determines the cost shares of the agents, which may not be realistic in many network design scenarios. Arbitrary cost sharing allows the agents to pick their own cost shares, without any requirements by the central authority. Thirdly, although the players are trying to minimize their payments in fair cost sharing, they are not permitted to adjust their payments freely, i.e., a player cannot directly specify her payments on each edge, but is rather asked to specify which edges she wants to use. In the network design contexts that we consider here, we prove that giving players more freedom can often result in better outcomes.

The research on non-cooperative network design and formation games is too much to survey here, see [19,21,24] and the references therein. See [1] for the preliminary conference version of this manuscript.

Our Results Our main results are about the existence and computation of cheap approximate equilibria. By an α-approximate Nash equilibrium, we mean that no player in such a solution has a deviation that will improve her cost by a factor of more than α. While our techniques are inspired by [4], our problem and connectivity requirements are much more general, and so require the development of much more general arguments and payment schemes.

- In Section 3, we show that in the case where all nodes are player nodes, there exists a Nash equilibrium as good as OPT, i.e., the price of stability is 1.

- In Section 4, we show that in the general case where some nodes may not be player nodes, there exists a 2-approximate Nash equilibrium as good as OPT.
- We show that if every player is replaced by two players (or if every player node has at least two players associated with it), then the price of stability is 1. This is in the spirit of similar results from selfish routing [3,24], where increasing the total amount of players reduces the price of anarchy.
- Starting with a β-approximation to OPT, we provide poly-time algorithms for computing an $(1+\epsilon)$-approximate equilibrium with cost no more than β times OPT, for the case where all nodes are player nodes. The same holds for the general case with the factor being $(3.1+\epsilon)$ instead.

Since for monotone happiness functions F, OPT corresponds to a constrained forest problem [14], then the last result gives us a poly-time algorithm with $\beta = 2$. Notice that we assumed that the function F is monotone, i.e., addition of more terminals to a component does not hurt. This assumption is necessary, since as we prove in Section 6, if F is not monotone there may not exist *any* approximate Nash equilibria. We also show that the results above are only possible in our model with arbitrary cost-sharing, and not with fair sharing.

Because of its applications to multi-homing [5,25], we are especially interested in the behavior of Terminal Backup connectivity requirements, i.e., when a player node desires to connect to at least k other player nodes. For this special case, we prove a variety of results, such as price of anarchy bounds and the improvement of fair sharing results from [3] for this new problem. The lower bounds for Terminal Backup also hold for the general *Group Network Formation Game*, showing that while the price of stability may be low, the price of anarchy can be as high as the number of players.

2 Properties of the Socially Optimal Network

In this section, we will show some useful properties of the socially optimal network for the *Group Network Formation Game*, which we refer to as OPT. For notational convenience, we will extend the definition of the happiness function to subgraphs and use $F(S)$ to denote the value of the happiness function for the set of terminal nodes in a connected component S.

The cost of a network for player i in which her connectivity requirement is not satisfied is ∞. Therefore, OPT is the minimum cost network that satisfies the connectivity requirements of all the players. Furthermore, since the satisfaction of the players only depends on the terminal nodes they are connected to, then OPT is acyclic, since otherwise one can obtain a cheaper network that satisfies all the players simply by deleting any one of the edges of a cycle included in OPT.

Observation 1 *The socially optimal network for the Group Network Formation Game is the minimum cost forest that satisfies all the players.*

Let $e = (i, j)$ be an arbitrary edge of a tree T of OPT. Removal of e will divide T into 2 subtrees, namely T_i and T_j (let T_i be the tree containing node i). After removal of e, connectivity requirements of some of the players in T will be dissatisfied, i.e., either $F(T_i) = 0$ or $F(T_j) = 0$, since otherwise $OPT - e$ would be a network that is cheaper than OPT and satisfies all the players. Therefore, once e is deleted from OPT, all the players in T_i or T_j or both will be dissatisfied. The players that are dissatisfied upon removal of e are said to *witness* e. If e is witnessed by only the players in T_i or only the players in T_j then e is said to be an edge *witnessed from* 1*-side*. Analogously, we say e is *witnessed from* 2*-sides* if it is witnessed by all the players in T.

In general, some of the edges of a tree T may be witnessed from 1-side whereas some others are witnessed from 2-sides. We show that the edges of T witnessed from 2-sides form a connected component in T.

Proposition 1 *Let $e = (i, j)$ be an edge of T that is witnessed from 1-side, w.l.o.g. from the side of i. Then all the edges in T_i are also witnessed from 1-side.*

Proof. Let $f = (u, v)$ be an arbitrary edge in T_i and let v be the node closer to i in T_i. If f is removed from T then T would be divided into 2 trees, namely T_u and T_v where $T_u \subset T_i$ and $T_v \supset T_j$. Since e is witnessed only by the players in T_i, $F(T_i) = 0$ and $F(T_j) = 1$. Since F is a monotone function, $F(T_u) = 0$ and $F(T_v) = 1$ and therefore, f is witnessed from 1-side, from the side of u. ■

Corollary 1 *The edges of T witnessed from 2-sides form a connected component in T.*

Proof. If there is no or exactly one edge in T that is witnessed from 2-sides then the result trivially holds. Assume there exists an edge f that is witnessed from 2-sides. Let r be a node of T that is incident to f, and root T at r. Let e be an arbitrary edge in T that is witnessed from 2-sides. Observe that all the edges between e and the root r are witnessed from 2-sides, since if an edge of this path were witnessed from 1-side, then by Proposition 1 so would e. Therefore, the set of edges of T that are witnessed from 2-sides form a connected component in T that contains r. ■

3 When all Nodes are Terminals

For the *Group Network Formation Game*, we don't know whether there exists an exact Nash equilibrium for all possible instances of the problem. However, for the special case where each node of G is a terminal node, we prove that Nash equilibrium is guaranteed to exist. Specifically, there exists a Nash equilibrium whose cost is as much as OPT, and therefore price of stability is 1. In this section, we will prove this result by explicitly forming the stable payments on the edges of OPT by giving a payment algorithm. The payment algorithm, which will be formally defined below, first roots the tree and then loops through all the players/nodes in reverse BFS order and decides their payments for all their incident edges. The algorithm never asks a player i to pay for the cost of an edge e that is not incident to i.

Since we are trying to form a Nash equilibrium, no player i should have an incentive of unilateral deviation from her strategy p_i when the algorithm terminates, i.e., $|p_i|$ should not be more than the cost of the best deviation of player i. Observe that a best deviation of player i, which we denote by $\chi_i(p_{-i})$, is the cheapest strategy of player i that satisfies her connection requirement given the strategies p_{-i} of other players. While such a deviation may not be unique, for our purposes it is enough to let $\chi_i(p_{-i})$ denote an arbitrary best response of player i to strategy p_{-i}. We will show that $p_i + p_{-i}$ buys all the edges of OPT when our payment algorithm terminates, and that those payments form a Nash equilibrium.

Notation and Invariant Let p^* denote the cheapest strategy vector that buys all the edges of OPT, i.e., $p^*(e) = c(e)$ if e is in OPT and $p^*(e) = 0$ otherwise. Let $\overline{p_i}$ denote the minimum payment to be made by other players to buy all the edges of OPT given that player i plays the strategy p_i, i.e., $\overline{p_i} = p^* - p_i$. To have an easier analysis we want our algorithm to have a stronger property: we not only want it to ensure stability at termination but also at each intermediate step. In other words, at any step of the algorithm, the inequality $|p_i| \le |\chi_i(\overline{p_i})|$ will hold, i.e., the payment strategy p_i assigned to i should be the cheapest strategy of i that satisfies her connectivity requirement, assuming the rest of the payments to buy all the edges of OPT are made by other players. Note that if all the edges of OPT are bought, i.e., $p_i + p_{-i} = p^*$, then $p_{-i} = \overline{p_i}$ and the invariant $|p_i| \le |\chi_i(\overline{p_i})|$ turns into the Nash equilibrium condition. To show that our algorithm produces a Nash equilibrium as cheap as OPT, it is thus enough to prove the following two statements:

- The invariant $|p_i| \le |\chi_i(\overline{p_i})|$ holds at every step of our algorithm for all players i.
- When the algorithm terminates, all the edges of OPT are bought by the players.

Computing deviations Here we discuss how deviations can be computed. This will be important in Section 5 when we talk about our polynomial-time results, but in this section we are only concerned with existence results and so include this discussion only in order to improve intuition about the nature of deviations. When computing $\chi_i(\overline{p_i})$, note that all edges of OPT such that i is not contributing any payment to them can be used by i freely to satisfy her connectivity requirement. Therefore, when computing the cheapest deviation for a player i, we should not use the actual cost of the edges in G, but instead for each edge f, we should use the cost i would face if she is to use f, which will be referred to as *modified cost of f for i,* and denoted by $c'_i(f)$ in the rest of the paper. Specifically, for f not in OPT, $c'_i(f) = c(f)$, the actual cost of f. For the edges f of OPT that i has not contributed anything to (i.e., $p_i(f) = 0$), we have that $c'_i(f) = 0$, since from i's perspective, she can use these edges for free because other players have paid for them. For all the other edges f that i is paying $p_i(f)$ for, $c'_i(f) = p_i(f)$, since that is how much it costs for i to use f in her deviation

```
Input: The socially optimal network OPT
Output: The payment scheme for OPT that is a Nash equilibrium
Initialize p_i(e) = 0 for all players i and edges e;
Root each tree T of OPT by an arbitrary node incident to an edge witnessed from 2-sides;
Loop through all trees T of OPT;
    Loop through all nodes i of T in reverse BFS order;
        Let T_i be the subtree of T below i;
        Loop through all edges e of T_i incident to i;
            Let d(e) = c(e) - ∑_{j≠i} p_j(e);
            If |χ_i(p̄_i, e)| - |p_i| ≥ d(e)
                Set p_i(e) = d(e);
            Else break;
        Let g to be the parent edge of node i;
        Set p_i(g) = min{|χ_i(p̄_i, e)| - |p_i|, c(g)};
```

Algorithm 1: Algorithm that generates payments on the edges of OPT

from the payment strategy p_i. Using these modified costs, $\chi_i(\overline{p_i})$ is simply the cheapest set of edges which fulfill player i's connectivity requirements.

We now present an algorithm which satisfies the two properties mentioned above (the pseudocode is shown in Algorithm 1). At the beginning of the algorithm $|p_i| = 0$ for all players i, and therefore the invariant is trivially satisfied by all the players. At every step, the algorithm asks a player i to make a payment for an incident edge e of i. Let $d(e)$ denote the amount of payment i should make in order to buy e, i.e., $d(e) = c(e) - \sum_{j \neq i} p_j(e)$. Recall that the algorithm should never assign a payment for a player that violates the invariant $|p_i| \leq |\chi_i(\overline{p_i})|$. Let x denote the maximum amount of payment player i can make for e in order not to violate the invariant. If $x \geq d(e)$ then the algorithm should ask i to pay $d(e)$ for e and ask it to pay x for e otherwise, and therefore the invariant is never violated throughout the algorithm.

What is this value x, however, and how to we compute it? For a strategy p_i of player i, let $\chi_i(\overline{p_i}, e)$ denote the cheapest deviation of player i from the strategy p_i that does not use the edge e, assuming that the rest of the players are paying for $\overline{p_i}$. Observe that $|\chi_i(\overline{p_i}, e)| \geq |\chi_i(\overline{p_i})|$. Then we argue below that $x \geq \min\{|\chi_i(\overline{p_i}, e)| - |p_i|, d(e)\}$. To see this, we consider the strategy $p_i + x$ where player i pays x for edge e. We are abusing notation here, since x is a number, not a payment vector. Formally, by $p_i + x$ we will mean the payment strategy which equals p_i everywhere except at e, with $(p_i + x)(e) = p_i(e) + x$.

Lemma 1 *Given payments p_i which do not violate the invariant, player i can increase her payments on edge e by* $\min\{|\chi_i(\overline{p_i}, e)| - |p_i|, d(e)\}$ *and not violate the invariant.*

Proof. To see this, suppose that x is the maximum amount that i can pay for edge e without violating the invariant, and $x < d(e)$. Now suppose to the contrary that $x < |\chi_i(\overline{p_i}, e)| - |p_i|$. This means that $|p_i + x| = |p_i| + x < |\chi_i(\overline{p_i}, e)|$. By Lemma 2, we know that $|\chi_i(\overline{p_i}, e)| = |\chi_i(\overline{p_i + x})|$. Therefore, $|p_i + x| < |\chi_i(\overline{p_i + x})|$, and so we can increase x and still not violate the invariant. This gives us a contradiction with x being maximum. ∎

Lemma 2 *Let p_i denote the strategy of player i right before she is asked to make a payment for e and let $x < d(e)$ be the maximum amount of payment i can make for e without violating the invariant. Then, $|\chi_i(\overline{p_i}, e)| = |\chi_i(\overline{p_i + x})|$, i.e, at least one of the best deviations of player i right after she makes a payment of x for e does not use e.*

Proof. For the purpose of contradiction, assume that all of the cheapest deviations of player i after she pays x for e, i.e., $\chi_i(\overline{p_i + x})$, uses e. Let $\mu = |\chi_i(\overline{p_i + x}, e)| > |\chi_i(\overline{p_i + x})|$ be the cost of the cheapest deviation of player i that does not use e right after i pays x for e. Observe that if the payment of i for e is increased by $y = \min\{\mu - |\chi_i(\overline{p_i + x})|, d(e) - x\}$, the cost of all deviations of player i that use edge e increases by the same amount. The invariant is still not violated after this increase, which contradicts with the assumption that x is the maximum amount of payment she can make for e without violating the invariant. ∎

Because of Lemma 1, it is clear that Algorithm 1 maintains the invariant at every step, since we never ask a player to pay for more than $\min\{|\chi_i(\overline{p_i}, e)| - |p_i|, d(e)\}$ for an edge e. We must now show that Algorithm

1 purchases all the edges of OPT. Recall that the algorithm asks the players to pay for their incident edges only. Therefore, each edge is considered for payment twice. For each edge $e = (i, j)$ where j is the parent of i, first i is asked to make her maximum amount of payment for e that will not violate the invariant. At the later iterations of the algorithm, when j is processed, the algorithm asks j to pay for the remaining cost of e. Therefore, if the payment algorithm successfully pays for the cost of all the edges of OPT, i.e., it does not execute the 'break' command, then it finds a Nash equilibrium whose cost is as much as OPT. To prove our result all we need to do is to prove that the algorithm never executes the 'break' command at any intermediate step. We will prove this by constructing a network cheaper than OPT which satisfies all the players' connectivity requirements whenever the algorithm executes 'break', thus forming a contradiction.

Specifically, we will consider networks formed by players' deviations. Recall that by Lemma 2, when a player i cannot pay $d(e)$ but some amount $x < d(e)$ for an incident edge e without violating the invariant, she does have a deviation that costs as much as her strategy $p_i + x$ and does not use e. Define $X_i(p_i, e)$ to be the graph formed by removing the edges paid for by $p_i + x$ from OPT, and then adding the edges paid for in $\chi_i(\overline{p_i}, e)$. In other words, $X_i(p_i, e)$ is the network of bought edges formed if player i deviates from her current strategy $p_i + x$ to $\chi_i(\overline{p_i}, e)$, with the payments of the other players being $\overline{p_i + x}$. The edges added to OPT by this deviation cost at most $|p_i + x|$, and the edges removed cost strictly greater than $|p_i + x|$. The cost is *strictly* greater because e is one of the edges removed, and player i does not fully pay for edge e in the payment $p_i + x$. Therefore, we know that the graph $X_i(p_i, e)$ is strictly cheaper than OPT.

Since the algorithm roots T by a node incident to an edge witnessed from 2-sides and the edges of T that are witnessed from 2-sides form a connected component in T by Corollary 1, every edge $e = (i, j)$ of T that is witnessed from 1-side is witnessed from the side of the lower level adjacent node. Without loss of generality, assume i is the lower level adjacent node of e, i.e., $e \notin T_i$ and $e \in T_j$. The algorithm cannot execute the 'break' command while a player is asked to pay for the cost of an edge e witnessed from 1-side since, as Lemma 3 proves, i will pay for the whole cost of e when she is asked to make a payment for e. In order to show that the algorithm does not execute the 'break' command when a player is asked to pay for an edge that is witnessed from 2-sides, we need the nice properties of graphs $X_i(p_i, e)$ given by Lemma 4 and Lemma 5.

Lemma 3 *Let $e = (i, j)$ be an edge of T that is witnessed from 1-side, and let i be the lower level incident node. Then when the algorithm asks player i to make payment for e, she will pay for the entire cost of e.*

Proof. We will prove the lemma by induction on the number of edges in the subtree T_i. First consider the case where i is a leaf-node of T, i.e., the number of edges in T_i is 0, as the base case of induction. If the player in i does not pay an amount $c(e)$ for the edge e then she has a deviation $\chi_i(\overline{p_i}, e)$ whose cost is less than $c(e)$. Then we will ask player i to play $\chi_i(\overline{p_i}, e)$ as her strategy. The graph $X_i(p_i, e)$ obtained by deleting e from OPT and adding the edges bought by the strategy $\chi_i(\overline{p_i}, e)$ is clearly cheaper than OPT since the total cost of the edges bought by $\chi_i(\overline{p_i}, e)$ is less than $c(e)$. Since none of the players except player i was witnessing e in OPT, the connectivity requirements of all other players will be satisfied in $X_i(p_i, e)$. The connectivity requirement of player i is trivially satisfied since she is the only deviating player. Since $X_i(p_i, e)$ satisfies all the players and is cheaper than OPT, the player in i will pay $c(e)$ for e (otherwise we have a contradiction).

Now consider an arbitrary edge $e = (i, j)$ that is witnessed from 1-side. Note that all the edges in T_i are witnessed from 1-side by Proposition 1 and therefore are bought by their lower level adjacent players by the inductive assumption. So player i is not asked to pay for any edge but e by the algorithm. Assume player i does not pay an amount $c(e)$ for the edge e when she is asked to make payment for e. Then she has a deviation $\chi_i(\overline{p_i}, e)$ whose cost is less than $c(e)$. Then we will ask player i to play $\chi_i(\overline{p_i}, e)$ as her strategy. Similar to the above case, $X_i(p_i, e)$ is clearly cheaper than OPT since the total cost of the edges bought by $\chi_i(\overline{p_i}, e)$ is less than $c(e)$. Since none of the players except the ones in T_i were witnessing e in OPT and e is the only edge of OPT that is not part of $X_i(p_i, e)$, then the connectivity requirements of all the players except the ones in T_i are satisfied in $X_i(p_i, e)$. However, since all the edges of T_i are part of $X_i(p_i, e)$, all the players of T_i are in the same connected component of $X_i(p_i, e)$. Since $X_i(p_i, e)$ satisfies the connectivity requirement of player i(the happiness function F evaluates to 1 for the connected component of i in $X_i(p_i, e)$), it also satisfies the connectivity requirements of all the players in T_i as well. Since $X_i(p_i, e)$ satisfies the connectivity requirements of all the players and cheaper than OPT, we have a contradiction. Therefore, player i pays the whole cost $c(e)$ of e when the algorithm asks her to make a payment for e. ■

Lemma 4 *Let C be the connected component containing i in $X_i(p_i, e)$. Then, C contains either all players of T or another tree T' of OPT.*

Proof. By Lemma 3, all the subtrees T_u of i that are linked to i by a 1-sided edge (u, i) will be in C, since the edge (u, i) will be entirely paid for by u. Let T_v be a subtree linked to i by a 2-sided edge (v, i), and suppose to the contrary that T_v is not in C. Since (v, i) is a 2-sided edge, then $T - T_v$ is not a happy component, and so in order for the connectivity requirement of i to be satisfied, she must be connected to some tree T' of OPT, as desired. ∎

Let i be a player that paid $x < c(e)$ for her upper level incident edge $e = (i, j)$ when the algorithm asked her to make a payment for e. Then, by Lemma 4, the connected component C of $X_i(p_i, e)$ that contains i, either contains all the players of T or another tree T' of OPT. If C contains all the players of T then $X_i(p_i, e)$ would be a graph cheaper than OPT and satisfying the connectivity requirements of all the players, which would be a contradiction. Assume C does not contain all the players of T but another tree T' of OPT. Let S be a subset of the players in T_i such that for every $u \in S$, u paid strictly less than $c(f)$ for her upper level incident edge f when the algorithm asked her to pay for f. The following lemma states that we can obtain a graph G' cheaper than OPT that satisfies the connectivity requirements of all the players in T_i, by replacing the strategies p_u of some elements u of S with their respective deviations $\chi_u(\overline{p_u}, f)$, none of which use any of the edges of $T - T_i$. In other words, if a player i cannot pay the whole cost of her upper level incident edge then there is a way to satisfy the connectivity requirements of all the players in T_i without increasing their cost and without relying on the payments of the players in $T - T_i$.

Lemma 5 *Let i be a player that could not pay the whole cost of her upper level incident edge e and the connected component of $X_i(p_i, e)$ that contains i does not contain all the players of T. Then, one can obtain a graph G' cheaper than OPT that satisfies the connectivity requirements of all the players in T_i, by replacing the strategies p_u of some elements u of S with their respective deviations $\chi_u(\overline{p_u}, f)$, none of which use any of the edges of $T - T_u$.*

Proof. We will prove the lemma by induction on the number of nodes in T_i. Consider the case where i is a leaf node as the base case. If player i cannot pay $c(e)$ but some amount $x < c(e)$ when the algorithm asks her to pay for her upper level incident edge e, then she has a deviation $\chi_i(\overline{p_i}, e)$ such that $|\chi_i(\overline{p_i}, e)| = x < c(e)$. Observe that the set of edges bought in $X_i(p_i, e)$ cannot be connecting i to $T - T_i$ since e is the cheapest path between i and $T - T_i$. Therefore, i will be connected to a different tree of OPT without connecting to $T - T_i$ if she plays $\chi_i(\overline{p_i}, e)$ as her strategy. Since i is the only deviating player, her connectivity requirements are trivially satisfied in $X_i(p_i, e)$ and since there is no other player in T_i, the lemma holds.

Consider the case where i is not a leaf-node. If player i cannot pay $c(e)$ but some amount $x < c(e)$ when the algorithm asks her to pay for her upper level incident edge e, then she has a deviation $\chi_i(\overline{p_i}, e)$ such that $|\chi_i(\overline{p_i}, e)| = |p_i + x| < |p_i| + c(e)$ where p_i is the strategy of player i right before she pays for e and therefore, $X_i(p_i, e)$ is strictly cheaper than OPT. By Lemma 4, the connected component C of $X_i(p_i, e)$ that contains i, either contains all terminals of T or a different tree T' of OPT. As pointed out above, C cannot contain all the terminals in T since otherwise $X_i(p_i, e)$ would be a graph cheaper than OPT that satisfies the connectivity requirements of all the players. Therefore, C contains a different tree T' of OPT. Observe that i is not connected to the terminals of $T - T_i$ in $X_i(p_i, e)$ since the shortest path between i and $T - T_i$ is e and therefore, $\chi_i(\overline{p_i}, e)$ would not be a best deviation of player i otherwise. The set of players in the subtrees of T_i that are connected to i in $X_i(p_i, e)$ (included in C) are connected to T' as well, and so their connectivity requirements are also satisfied in $X_i(p_i, e)$. Let $i_1, i_2, \ldots, i_k$ be the children of i in T such that they are not connected to i in $X_i(p_i, e)$, i.e., C does not contain them. By the inductive hypothesis, for every subtree T_{i_j}, there is a set of players S_j in T_{i_j} such that by replacing the strategies of the players in S_j with their respective best deviations, none of which uses any of the edges of $T - T_{i_j}$, we can obtain a graph cheaper than OPT where the connectivity requirements of all the players in T_{i_j} are satisfied.

Consider the graph G' that results by replacing the strategies of the players in $\cup_j S_j \cup \{i\}$ with their respective best deviations. Notice that all the players in each subtree T_{i_j} are connected to a tree of OPT other than T in G', since the best deviations of none of the players in S_j uses any edges of $T - T_{i_j}$, and so the fact that other players $(\cup_{l \neq j} S_l \cup \{i\})$ deviate and possibly remove their payments from some edges of $T - T_{i_j}$ does not affect them. The players connected to i in $X_i(p_i, e)$ are still connected to a tree T', since all the edges of the connected component C of $X_i(p_i, e)$ are also in G'. Therefore, all the players of T_i are connected to trees of OPT other than T. Since the happiness function F is monotone, this means that all the connectivity requirements of all players in T_i are satisfied. ■

We are now ready to prove that Algorithm 1 never executes the 'break' command, and thus pays for all the edges of OPT. For the purpose of contradiction, assume Algorithm 1 executed the 'break' command right after a player i is asked to pay for an incident edge $e = (v, i)$. Since the algorithm only executes the 'break' command if the sum of payments of the adjacent players v and i does not cover the total cost of the edge and the higher level incident player is always asked later than the lower level incident player, e is the lower level incident edge of player i. Observe that e is witnessed from 2-sides since otherwise v would have already paid for the total cost of e by Lemma 3. Recall that since i cannot pay for the remaining cost $d(e)$ of e but some amount $x < d(e)$ when she is asked to pay for e without violating the invariant, she does have a deviation $\chi_i(\overline{p_i}, e)$ such that $|\chi_i(\overline{p_i}, e)| = |p_i| + x$ where p_i is the strategy of player i right before she pays for e. Recall that the connected component C of $X_i(p_i, e)$ that contains i, either contains all terminals of T or a different tree T' of OPT by Lemma 4. However, C cannot contain all terminals of T since otherwise $X_i(p_i, e)$, which is cheaper than OPT, would be a network satisfying the connectivity requirements of all the players. Therefore, C contains a different tree T' of OPT, as well as $T - T_i$ (since i has not paid for any of the edges of $T - T_i$) and possibly some but not all subtrees of T_i. So, $X_i(p_i, e)$ satisfies the connectivity requirements of all the terminals except the ones that are in the subtrees of T_i which are not in C.

Let $T_{i_1}, T_{i_2}, \ldots, T_{i_k}$ be the subtrees of T_i that are not contained in C and let $f_1, f_2, \ldots, f_k$ be the upper level incident edges of the roots of these subtrees in OPT. Observe that none of the players at the roots of $T_{i_1}, T_{i_2}, \ldots, T_{i_k}$ have paid for the entire cost of their respective upper level incident edges $f_1, f_2, \ldots, f_k$ when the algorithm asked, since otherwise player i would contribute nothing to these edges, and so the subtrees $T_{i_1}, T_{i_2}, \ldots, T_{i_k}$ would be in C. To complete the proof all we need to prove is that we can modify $X_i(p_i, e)$ without increasing its cost such that we connect the players in $T_{i_1}, T_{i_2}, \ldots, T_{i_k}$ to happy connected components. This is exactly what Lemma 5 proves.

4 Good Equilibria in the General Game

In Section 3, we saw that a good equilibrium always exists when all nodes are terminals. In this section, we consider the general Group Network Formation Game, and show that there always exists a 2-approximate Nash equilibrium that is as cheap as the centralized optimum. By a 2-approximate Nash equilibrium, we mean a strategy profile $p = (p_1, p_2, \ldots, p_n)$ such that no player i can reduce her cost by more than a factor of 2 by unilaterally deviating from p_i to $\chi_i(p_{-i})$, i.e., $|\chi_i(p_{-i})| \geq |p_i|/2$ for all players i. To prove this, we first look at an important special case that we call the *Group Network Formation of Couples Game* or *GNFCG*. This game is exactly the same as the Group Network Formation Game, except that every player node has at least two players located at that node (although not all nodes need to be player nodes).

Theorem 1 *If the price of stability for the GNFCG is* 1 *then there exists a* 2*-approximate Nash Equilibrium for the Group Network Formation Game that costs as much as OPT.*

Proof. Assume we are given an instance $\Im_1 = (N_1, G, T, F)$ of a Group Network Formation Game, i.e., we are given a set of players $N_1 = \{1, 2, \ldots, n\}$, a graph $G = (V, E)$ such that each edge $e \in E$ is associated with a nonnegative cost $c(e)$, a set of terminal nodes $T \subseteq V$ such that each player $i \in N_1$ is located at a terminal node $u \in T$ and a monotone happiness function $F : 2^T \to \{0, 1\}$. All we need to show is that $\Im_1$ has a 2-approximate Nash equilibrium as cheap as OPT assuming price of stability for the Group Network Formation of Couples Game is 1.

We will first define an instance $\Im_2 = (N_2, G, T, F)$ of the Group Network Formation of Couples Game. Observe that the graph G, the set of terminal nodes T and the monotone happiness function F for both $\Im_1$ and $\Im_2$ are the same; however, the number of players of $\Im_2$ is twice as much as the number of players of $\Im_1$, i.e., $N_2 = \{1, 2, \ldots, 2n\}$. For each player $i \in N_1$ of $\Im_1$ located at $u \in T$ there are 2 corresponding players $i, (n + i) \in N_2$ of $\Im_2$ located at u.

First observe that the socially optimal network for both games $\Im_1$ and $\Im_2$ is the same since any network satisfying the players of $\Im_1$ also satisfies the players of $\Im_2$ and vice versa. Let OPT denote the socially optimal network of both of the games. As in Section 3, let p^* denote the strategy vector that buys all the edges of OPT, i.e., $p^*(e) = c(e)$ if e is in OPT and $p^*(e) = 0$ otherwise.

Since we have assumed that the price of stability for Group Network Formation of Couples Game is 1, we know that there exists a stable strategy profile $p = (p_1, p_2, \ldots, p_{2n})$ for the players N_2 of $\Im_2$ that buys OPT. That is, for each player $i \in N_2$, $|\chi_i(p^* - p_i)| = |p_i|$. Furthermore, since player i and player $(n+i)$ sit at the same terminal node for $i \le n$, the stable strategy p_i of player i is also a stable strategy for player $(n+i)$, i.e., if all the players except player $(n+i)$ pays $p^* - p_i$ then the best response of player $(n+i)$ is p_i.

To complete the proof, all we need to do is to give a strategy profile $p' = (p'_1, p'_2, \ldots, p'_n)$ for the players of $\Im_1$ such that $\sum_i p'_i = p^*$ and for each $i \in N_1$, $|p'_i| \le 2\,|\chi_i(p^* - p'_i)|$. We define p' as follows. For each player $i \in N_1$, $p'_i = p_i + p_{(n+i)}$. Since p_i and $p_{(n+i)}$ are stable strategies for players i and $(n+i)$ of $\Im_2$ respectively, we have that $|p_i| = |\chi_i(p^* - p_i)|$ and $|p_{(n+i)}| = \left|\chi_i\left(p^* - p_{(n+i)}\right)\right|$. Since $|\chi_i(p^* - p_i)|, \left|\chi_i\left(p^* - p_{(n+i)}\right)\right| \le |\chi_i(p^* - p'_i)|$, we have $|p'_i| = |p_i| + |p_{(n+i)}| \le 2\,|\chi_i(p^* - p'_i)|$. ■

Because of Theorem 1, we will focus on the GNFCG in the rest of the section and prove the existence of a Nash equilibrium as cheap as OPT. This result is interesting in its own right, since it states that to form an equilibrium that is as good as the optimum solution, it is enough to double the number of players. Such results are already known for many variants of congestion games and selfish routing [3,24], but as Theorem 2 shows, we can also prove such results for games with arbitrary sharing.

We use the same notation as in Section 3, including the definitions of p^*, $\overline{p_i}$, and χ_i. Given a set of bought edges T, a strategy profile p is a Nash equilibrium when $|\chi_i(p_{-i})| = |p_i|$ for all players i. To prove that price of stability is 1 for GNFCG, we give an algorithm that forms such a strategy profile on the edges of OPT. Recall that the payment strategies of all the players have to be stable when the algorithm terminates. As in Section 3, to have an easier analysis we not only want our algorithm to ensure stability at termination but also at each intermediate step. Specifically, we give an algorithm such that:

- The invariant $|p_i| \le |\chi_i(\overline{p_i})|$ holds at every step of our algorithm for all players i.
- When the algorithm terminates, all the edges of OPT are bought by the players.

The second property guarantees that the invariant is exactly the Nash equilibrium condition, since when all the edges of OPT are bought, then $p_{-i} = \overline{p_i}$. Therefore, the above two conditions imply that the price of stability is 1. In the rest of the section we prove our main theorem for the GNFCG.

Theorem 2 *For GNFCG, there exists a Nash equilibrium as cheap as the socially optimal network, i.e., the price of stability is* 1*.*

For ease of explanation, we will first consider the case where all the edges of OPT are witnessed from two sides and later illustrate how our algorithm can be modified for the case where some of the edges are witnessed from one side only. We start by rooting each connected component of OPT arbitrarily by a high degree non-player node. Throughout the paper, the term *high degree node* refers to the nodes with degree 3 or more. On each connected component T of OPT, we run a 2-phase algorithm. In the first phase of the algorithm, we assign players to make payments to the edges of T in a bottom-up manner, i.e., we start from a lowest level edge e of T and pick a player i to make some payment for e and continue with the next edge in the reverse BFS order. In the first phase of the algorithm, we ask a player i to contribute only for the cost of edges on the unique path between her and the root and furthermore, the payment for each edge is made by only one player.

Algorithm (Phase 1) For an arbitrary edge $e = (u, v)$ where u is the lower level incident node of e, the assignment of the player to pay for e is as follows. If u is a terminal node, we ask a player i located at node u to make maximum amount of payment on e that will not make p_i violate the invariant, i.e., we set $p_i(e) = \min\{\chi_i(\overline{p_i}, e) - |p_i|, c(e)\}$. This will not violate the invariant by the same reasoning as in Section 3 (i.e., Lemma 1 still holds). If u is a degree 2 nonterminal node then we ask the player who has completely bought the other incident edge of u, i.e., made a payment equal to the cost of that edge, to make maximum amount of payment on e that will not make her strategy unstable (i.e., violate the invariant) as shown on the left of Figure 1(A). Note that it may be the case that no player has bought the other incident edge of u in which case we don't ask any player to pay for e and the payment for e will be postponed to the second phase of the algorithm. If u is a high degree nonterminal then the selection of the player to pay for e is based on the

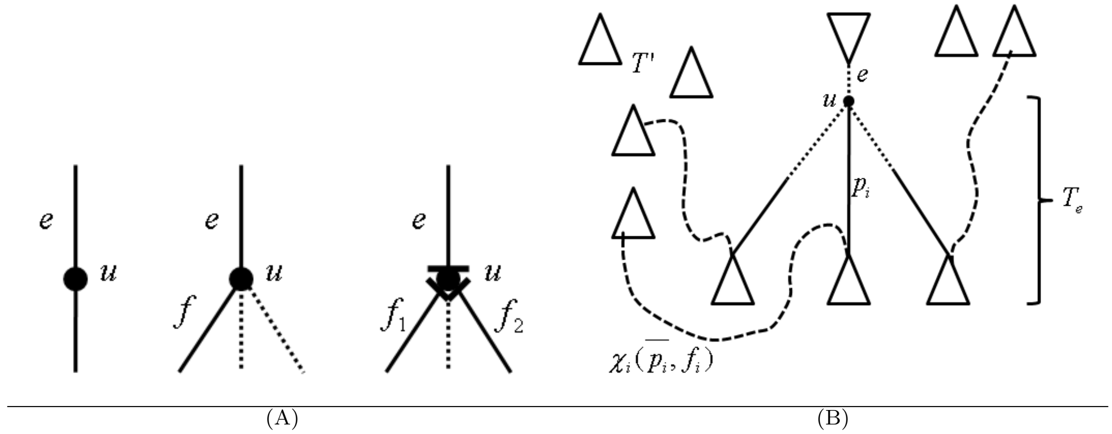

Fig. 1 (A) Illustrates the assignment of the player to pay for the cost of e. (B) Shows how to construct a cheap network that satisfies all the players in T_e by using the deviations of a subset S of them.

number of lower level incident edges of u that are bought in the previous iterations of the algorithm. If none of the lower level incident edges of u are bought then we postpone the payment on e to the second phase of the algorithm. If exactly one of the lower level incident edges of u, namely f, is bought then we ask the player who bought f to make maximum amount of payment on e that will not make her strategy unstable (i.e., violate the invariant) as shown in the middle of Figure 1(A). If 2 or more of the lower level incident edges of u are already bought, namely $f_1, f_2, \ldots, f_l$, then we fix the strategies of the players $i_1, i_2, \ldots, i_l$ that bought those edges, i.e., the players $i_1, i_2, \ldots, i_l$ are not going to pay any more and therefore the strategies of those players that will be returned at the end of the algorithm are already determined. Since there are two players located at every terminal, pick an arbitrary player located at the same terminal as one of $i_1, i_2, \ldots, i_l$ that has not made any payments yet, and assign her to make maximum amount of payment for e that will not make her strategy unstable as shown on the right of Figure 1(A). We later prove that such a player always exists, i.e., not all of $i_1, i_2, \ldots, i_l$ are the last players to make payment at their respective terminal nodes.

Notation We now define some helpful notation in order to prove some lemmas about the first phase of the algorithm which is fully specified above. When we are talking about a player i, let T denote the connected component of OPT containing i and let T' denote the set of other connected components of OPT. The strategy of a player is denoted by p_i which is a vector of length m, the total number of edges in G, where each entry of p_i indicates the payment player i is making for the corresponding edge. Recall that we use p^* for the strategy vector that buys OPT, i.e., $p^*(e) = c(e)$ if e is an edge of OPT and $p^*(e) = 0$ otherwise. Observe that when the algorithm terminates it should be that $\sum_i p_i = p^*$. We use the notation $G(p)$ for the subgraph of bought edges by strategy p, i.e., the subgraph composed of the edges for which $p(e) = c(e)$. For instance, $G(p^*)$ denotes OPT, $G(p^* - p_i)$ denotes the subgraph composed of edges bought by players other than i and $G(p^* - p_i + \chi_i(\overline{p_i}, e))$ denotes the subgraph of bought edges if player i deviates from her strategy p_i to her best deviation that does not use e, $\chi_i(\overline{p_i}, e)$. Finally, for an arbitrary edge e of a rooted tree T, we use T_e in order to refer to the subtree of T below e and $\overline{T_e}$ to refer to the rest of the tree $T - T_e$. The notation for the subtree of T rooted at a node u is analogously T_u.

We now present the analysis of the first phase of the algorithm by giving a series of lemmas that successively proves the following. For every edge e that could not be bought in the first phase of the algorithm by the assigned player to make payment for it, we can connect all the terminal nodes in T_e to the connected components of T' *without using any of the edges of* $\overline{T_e}$ by simply setting $p_i = \chi_i(\overline{p_i})$ for a subset S of players in T_e. The deviations of the subset S of the players are depicted in Figure 1(B). The condition that no edges of $\overline{T_e}$ are used by the deviations is crucial, since that is what allows us to have a set of players all deviate at once and still be satisfied afterwards. The fact that such a "re-wiring" exists allows us to argue in our proofs that at least one of the incident edges of the root of T will be bought during the first phase of the algorithm. In all the lemmas below, the payment p will refer to the payment at the end of Phase 1 of the algorithm.

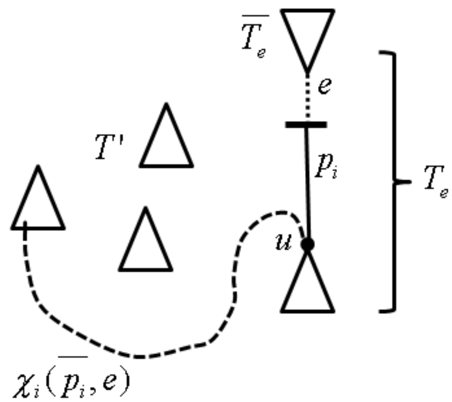

Fig. 2 Shows the deviation $\chi_i(p_i)$ of a player i that is located at terminal u and could not buy the edge e.

Lemma 6 *Let u be an arbitrary terminal node and i be the first player to make payments that is located at u. Let e be the first edge between u and the earliest ancestor of u which is either a terminal or a high degree nonterminal node such that i could not buy all of e without violating the invariant. Then all the players in T_u will be satisfied in the subgraph $G\left(p^* - p_i + \chi_i(\overline{p_i}, e)\right) - \overline{T_e}$.*

Proof. Since every edge f between u and e is bought by player i, i.e., $p^*(f) = p_i(f) = c(f)$ and player i did not pay for any other edge in G, then the subgraph $G(p^* - p_i)$ consists of the connected components T', T_u and $\overline{T_e}$. Observe that the players in T_u, including i, are not satisfied in $G(p^* - p_i)$ since according to our assumption, any edge of OPT is witnessed from two sides, and so any subset of T_e cannot be a happy component. Since player i is satisfied in $G\left(p^* - p_i + \chi_i(\overline{p_i}, e)\right)$ and not satisfied in $G(p^* - p_i)$, then the subgraph $G\left(p^* - p_i + \chi_i(\overline{p_i}, e)\right) - G(p^* - p_i)$ must include a path P_u between T_u and one of the connected components in T' or $\overline{T_e}$.

For the purpose of contradiction, assume P_u is a path between T_u and $\overline{T_e}$. Since $T - T_u - \overline{T_e}$ is a path of degree 2 nonterminal nodes between T_u and $\overline{T_e}$, then all the players of T will be in the same connected component of $G\left(p^* - p_i + \chi_i(\overline{p_i}, e)\right)$ and therefore the subgraph $G\left(p^* - p_i + \chi_i(\overline{p_i}, e)\right)$ satisfies all the players. This is because i must be satisfied in $G\left(p^* - p_i + \chi_i(\overline{p_i}, e)\right)$, and so it is in a happy component. Observe that $G\left(p^* - p_i\right)$ has the same set of edges as OPT except e and the edges bought by player i. Since the cost of $\chi_i(\overline{p_i}, e)$ is equal to the cost of p_i by Lemma 2 (which still holds), then $G\left(p^* - p_i + \chi_i(\overline{p_i}, e)\right)$ is a subgraph satisfying all the players and cheaper than OPT. More precisely, the cost of $G\left(p^* - p_i + \chi_i(\overline{p_i}, e)\right)$ is less than the cost of OPT by an amount $c(e) - p_i(e)$. Since OPT is the cheapest network satisfying all the players, this is a contradiction and therefore there cannot exist a path between T_u and $\overline{T_e}$ in $G\left(p^* - p_i + \chi_i(\overline{p_i}, e)\right)$. Since the players in T_u are in a connected component that is disjoint from all the nodes and the edges of $\overline{T_e}$ in $G\left(p^* - p_i + \chi_i(\overline{p_i}, e)\right)$, then they are all satisfied in $G\left(p^* - p_i + \chi_i(\overline{p_i}, e)\right) - \overline{T_e}$ as shown in Figure 2. ∎

Lemma 7 *Let i be a player such that i did not buy e, i.e., $p_i(e) < c(e)$, even though the algorithm assigned i to make payment for e. Then there exists a subset of players $S = (s_1, \ldots, s_k)$ in T_e with deviations $\chi_1(\overline{p_1}, f_1), \ldots, \chi_k(\overline{p_k}, f_k)$, where f_l is the edge player l could not fully buy, from their respective strategies $p_1, \ldots, p_k$ such that all the players in T_e will be satisfied in the subgraph $G\left(p^* - \sum_{l \in S} p_l + \sum_{l \in S} \chi_l(\overline{p_l}, f_l)\right) - \overline{T_e}$.*

Proof. We will prove this result by induction on the number of nodes in T_e. If T_e has only 1 node u, then the lemma holds by Lemma 6. Below we will use the notation $R(S)$ to denote $G(p^* - \sum_{l \in S} p_l + \sum_{l \in S} \chi_l(\overline{p_l}, f_l))$ and $R(S, e)$ to denote $R(S) - \overline{T_e}$.

Let us assume that the lemma holds for all instances such that the number of nodes in T_e is at most k and let's prove that the lemma also holds when the number of nodes is $k+1$. Let Γ be the set composed of connected components of $G\left(\sum_{i \in T_e} p_i\right)$ that involves at least one terminal node and let C be the connected component involving i. Observe that C is the highest level connected component, i.e., the one that is adjacent to e.

Let u be the highest level terminal or high degree nonterminal node in C. If u is a terminal node then the result directly follows from Lemma 6, with $S = \{i\}$, since the terminals in T_u are the same as in T_e. Therefore, assume u is a high-degree nonterminal node. Observe that at least one of the lower level incident edges of u is a bought edge since otherwise C would not be a connected component of $G\left(\sum_{i \in T_e} p_i\right)$ that involves at least one terminal node. Recall that if at least 2 of the lower level incident edges of u are bought, i is only assigned to make payment for the edges above u, i.e., player i has not made any payment on the edges below u. Therefore, all we need to do is to exactly repeat the proof of Lemma 6, once again giving us the desired result with $S = \{i\}$. Let us now consider the final case, which is depicted in Figure 1(B), where u is a high-degree nonterminal node such that exactly one lower level incident edge of u is bought.

In this case, let $C_1, C_2, \ldots, C_k$ be the elements of Γ that are one level lower than C and let $e_1, e_2, \ldots, e_k$ be the immediate higher level unpaid edges of $C_1, C_2, \ldots, C_k$ respectively. By the inductive hypothesis, each of these edges e_j already has a desired set of players S_j in T_{e_j}.

The connected component containing i in the subgraph $G\left(p^* - p_i + \chi_i(\overline{p_i}, e)\right)$ may also contain a player j in T_{e_j}. Since player i did not make any payment for the edges in T_{e_j}, then all the edges of T_{e_j} are also part of $G\left(p^* - p_i + \chi_i(\overline{p_i}, e)\right)$ and therefore all the players in T_{e_j} are in the same connected component with i in $G\left(p^* - p_i + \chi_i(\overline{p_i}, e)\right)$. Let Λ be the edges of $e_1, e_2, \ldots, e_k$ such that T_{e_j} is *not* in the same connected component of $G(p^* - p_i + \chi_i(\overline{p_i}, e))$ as i. Then, we set the set S to be $\cup_{e_j \in \Lambda} S_j \cup \{i\}$.

We must now prove that all the players in T_e are satisfied in the subgraph $R(S, e) = G(p^* - \sum_{l \in S} p_l + \sum_{l \in S} \chi_l(\overline{p_l}, f_l)) - \overline{T_e}$. First, we prove this for the players in T_{e_j} for $e_j \in \Lambda$. By the inductive hypothesis, they are all satisfied in the subgraph $R(S_j, e_j) = G\left(p^* - \sum_{l \in S_j} p_l + \sum_{l \in S_j} \chi_l(\overline{p_l}, f_l)\right) - \overline{T_{e_j}}$. Let g be an arbitrary edge in a connected component of a player in T_{e_j} in the subgraph $R(S_j, e_j)$. This edge cannot be in $\overline{T_e}$, since $\overline{T_e} \subseteq \overline{T_{e_j}}$. This edge cannot be paid for by a player outside T_{e_j}, since those players only pay for edges in $\overline{T_{e_j}}$. Therefore, g must still be present in $R(S, e)$, and so all players in T_{e_j} are still satisfied in $R(S, e)$.

Now consider the players in C and in T_{e_j} for $e_j \notin \Lambda$. They are satisfied in $G(p^* - p_i + \chi(\overline{p_i}, e))$ since i is satisfied and they are in the same connected component. Let g be an arbitrary edge in the connected component of $G(p^* - p_i + \chi(\overline{p_i}, e))$ containing player i. These players are satisfied in $R(S) = G\left(p^* - \sum_{l \in S} p_l + \sum_{l \in S} \chi_l(\overline{p_l}, f_l)\right)$, since g is still present in $R(S)$. This is because if g were being paid for by a player j, then it would be part of some subtree T_{e_j} with $e_j \notin \Lambda$, and so $j \notin S$, and those payments on g would remain unchanged. We need to prove that players in C and in T_{e_j} for $e_j \notin \Lambda$ are satisfied in $R(S, e)$, and so it is enough to show that $g \notin \overline{T_e}$. If this were not the case, then $\overline{T_e}$ is in the same connected component of $R(S)$ as i. Since i is satisfied in $R(S)$, then so are all the players in $\overline{T_e}$. We already proved that all players in all subtrees T_{e_j} are satisfied in $R(S)$, and so $R(S)$ is a feasible solution (all the players in the graph are satisfied). Notice, however, that the solution $R(S)$ is cheaper than OPT, by the same argument as in Lemma 6, and so this is not possible.

Therefore, all the players in T_e are satisfied in $R(S, e)$, as desired. ∎

Lemma 8 *Let $C_1, C_2, \ldots, C_k$ be the highest level connected components of the subgraph $G\left(\sum_i p_i\right)$ that include at least one terminal node, and suppose that these do not include the root of T. Let $e_1, e_2, \ldots, e_l$ be the immediately above incident edges of $C_1, C_2, \ldots, C_k$ respectively. Let $\overline{T_{e_1, e_2, \ldots, e_l}} = T - T_{e_1} - T_{e_2} - \ldots - T_{e_k}$. Then for each C_j there exists a corresponding set S_j of players in T_{e_j} such that all players in T are satisfied in $G\left(p^* - \sum_{l=1}^{k}\left(\sum_{j \in S_l} p_j - \sum_{j \in S_l} \chi_j(\overline{p_j}, f_j)\right)\right) - \overline{T_{e_1, e_2, \ldots, e_k}}$, where f_j is the edge player j could not fully buy.*

Proof. Let i be an arbitrary player in T. Without loss of generality assume it is in T_{e_j}. There is a set S_j such that player i is satisfied in $G\left(p^* - \sum_{j \in S_j} p_j - \sum_{j \in S_j} \chi_j(\overline{p_j}, f_j)\right) - \overline{T_{e_j}}$ due to Lemma 7. To prove the lemma, all we need to show is that every edge of $G\left(p^* - \sum_{j \in S_j} p_j + \sum_{j \in S_j} \chi_j(\overline{p_j}, f_j)\right) - \overline{T_{e_j}}$ is also an edge of $G\left(p^* - \sum_{l=1}^{k}\left(\sum_{j \in S_l} p_j - \sum_{j \in S_l} \chi_j(\overline{p_j}, f_j)\right)\right) - \overline{T_{e_1, e_2, \ldots, e_k}}$.

First let us consider the edges of T. Observe that none of the edges in $\overline{T_{e_j}}$ are in $G(p^* - \sum_{j \in S_j} p_j + \sum_{j \in S_j} \chi_j(\overline{p_j}, e)) - \overline{T_{e_j}}$. Therefore all we need is to check the edges in T_{e_j}. Since the algorithm never assigns a player i to pay for the edges that are not on the unique path between the terminal i and the root of T, none of the the players in $T - T_{e_j}$ made payment for any edge in T_{e_j}. Since the players in $\bigcup_{l \neq j} S_l$ did not make payment on the edges of T_{e_j}, then an edge of T_{e_j} that is in $G\left(p^* - \sum_{j \in S_j} p_j + \sum_{j \in S_j} \chi_j(\overline{p_j}, f_j)\right) - \overline{T_{e_j}}$ is also an edge of $G\left(p^* - \sum_{l=1}^{k}\left(\sum_{j \in S_l} p_j - \sum_{j \in S_l} \chi_j(\overline{p_j}, f_j)\right)\right) - \overline{T_{e_1,e_2,\ldots,e_k}}$.

Now let us consider the edges of $\overline{T}$, i.e., the edges outside of T. Since the algorithm never asks the players to pay for the edges outside of T, an edge not in T is in $G\left(p^* - \sum_{j \in S_j} p_j + \sum_{j \in S_j} \chi_j(\overline{p_j}, f_j)\right) - \overline{T_{e_j}}$ if and only if it is also in $G\left(\sum_{j \in S_j} \chi_j(\overline{p_j}, f_j)\right)$. Therefore, any edge of $\overline{T}$ that is in $G\left(p^* - \sum_{j \in S_j} p_j + \sum_{j \in S_j} \chi_j(\overline{p_j}, f_j)\right) - \overline{T_{e_j}}$ is in $G\left(p^* - \sum_{k=1}^{l}\left(\sum_{j \in S_k} p_j - \sum_{j \in S_k} \chi_j(\overline{p_j}, f_j)\right)\right) - \overline{T_{e_1,e_2,\ldots,e_l}}$ as well. ■

Lemma 9 *At least one of the incident edges of the root of T will be bought at the end of the first phase of the algorithm.*

Proof. For the purpose of contradiction assume none of the incident edges of the root of T is bought at the first phase of the algorithm and let's obtain a contradiction by constructing a subgraph that is cheaper than OPT and satisfies all the players. According to Lemma 8, the graph

$$R = G(p^* - \sum_{l=1}^{k}\left(\sum_{j \in S_l} p_j - \sum_{j \in S_l} \chi_j(\overline{p_j}, f_j)\right)) - \overline{T_{e_1,e_2,\ldots,e_k}}$$

satisfies all the players. Therefore, all we need to show is that the cost of R is less than the cost of OPT. Since the cost of $\chi_i(\overline{p_i}, f_i)$ is equal to the cost of p_i by Lemma 2, then we know by the same argument as in the proof of Lemma 6 that R is strictly cheaper than OPT. ■

Algorithm (Phase 2) In the second phase of the algorithm, we ask the players that have not made any payments yet to make stable payments for the remaining edges and buy them. Let Γ be the set composed of connected components of $G(p) - T'$ that include at least one terminal node. In other words, Γ consists of connected components of the edges in T purchased so far by the algorithm (a single terminal node with no adjacent bought edges would also be a component in Γ). We call a connected component $C_1 \in \Gamma$ *immediately below* a connected component $C \in \Gamma$ if after contracting the components in Γ, C is above C_1 in the resulting tree and there are no other components of Γ between them. In the second phase of the algorithm, we form payments on the edges in a top-down manner as we explain next. We start from the connected component $C \in \Gamma$ that includes the root of T (this must exist by Lemma 9) and assign a player i in C that has not made any payments yet to buy *all* the edges between C and the connected components that are immediately below C. The set of edges i should buy are shown in Figure 3. We prove that such a player i always exists in Lemma 10. Observe that once i buys all the edges between C and the connected components $C_1, C_2, \ldots, C_k$ that are immediately below C, all these $k+1$ connected components form a single connected C that contains the root. We repeat this procedure, i.e., pick a player i in the top-most connected component C that has not made a payment yet to buy all the edges between C and the connected components that are immediately below C, until all the players in T are in the same connected component and all of T is paid for.

Proof of Theorem 2. To show that our algorithm forms an equilibrium payment, we need to prove that all of OPT is paid for when it terminates, and that the invariant is never violated. It is clear that all of OPT is fully paid for, since Phase 2 of the algorithms pays for every edge of OPT that was not paid for in Phase 1. It is also clear that the invariant is never violated during the first phase by construction, and so the final payments of players used in Phase 1 are stable. To finish the proof, we need to show that a strategy p_i that buys all the edges between a connected component C and the connected components $C_1, C_2, \ldots, C_k$ that are immediately below C is a stable strategy for any player in C, which we show in Lemma 11.

This concludes the proof of Theorem 2. Recall that for ease of explanation, we only considered the case where all edges of OPT are witnessed from two sides until now. In Lemma 12, we modify this algorithm to return a Nash equilibrium that purchases OPT even if some of the edges of OPT are witnessed from one side. ■

Lemma 10 *At any stage of the algorithm, each connected component of Γ has a player i such that $p_i(e) = 0$ for all edges e.*

Proof. The statement is trivially true at the start of the algorithm. In the first phase, consider a time when we ask a player i, which is not the first to pay among the players with the same terminal node of i, to make payment for some edges. This only occurs when at least 2 of the lower level incident edges of a high-degree nonterminal u are bought. But then 2 or more connected components of Γ merge. Since each of these connected components had at least one player who hasn't made any payment yet, and only one of them is being asked to pay at this moment, then every component of Γ still has at least one player that has not made any payments.

In the second phase of the algorithm, one player i buys all the edges between a connected component C and the connected components $C_1, C_2, \ldots, C_k$ that are immediately below C as shown in Figure 3. Similar to the above case, after the player i makes her payment, at least 2 connected components of Γ merge. Therefore, the lemma holds at the end of the second phase of the algorithm. ■

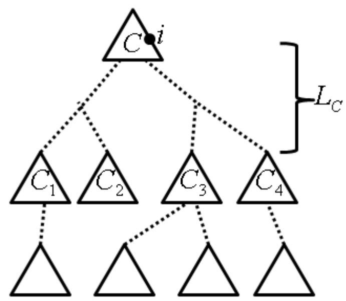

Fig. 3 Illustrates the set of edges to be bought by a player i located at the connected component that contains the root.

Lemma 11 *Let $C \in \Gamma$ be a connected component of bought edges containing the root and let $C_1, C_2, \ldots, C_k \in \Gamma$ be the connected components of bought edges that are immediately below C. Then for any player i in C the strategy p_i that buys all the edges between C and $C_1, C_2, \ldots, C_k$ is stable, i.e., $|p_i| \leq |\chi_i(\overline{p_i})|$.*

Proof. Let L_C denote the set of edges between C and $C_1, C_2, \ldots, C_k$. Observe that even though all the edges in L_C are not bought, a player j may have made a payment $p_j(e) < c(e)$ for some edge $e \in L_C$ in the first phase of the algorithm. Therefore, the cost of the strategy p_i of player $i \in C$ that buys all the edges of L_C, which we denote by l_i, may be less than $\sum_{e \in L_C} c(e)$. More precisely, $l_i = \sum_{e \in L_C} \Big(c(e) - \sum_j p_j(e)\Big)$.

We claim that a strategy p_i of a player $i \in C$ that buys all the edges in L_C is stable. For the purpose of contradiction, assume p_i is not a stable strategy, i.e, i could not pay the remaining cost of all the edges in L_C. Then, player i has a deviation $\chi_i(\overline{p_i})$ from p_i such that the cost of $\chi_i(\overline{p_i})$ is strictly less than l_i, and therefore the subgraph $G\left(p^* - p_i + \chi_i(\overline{p_i})\right)$ is cheaper than OPT. Since player i did not make any payment for the edges in C, all the players in C are in the same connected component of $G\left(p^* - p_i + \chi_i(\overline{p_i})\right)$ as i and therefore are satisfied. If the players in $C, C_1, C_2, \ldots, C_k$ are also in the same connected component with i in $G\left(p^* - p_i + \chi_i(\overline{p_i})\right)$, then $G\left(p^* - p_i + \chi_i(\overline{p_i})\right)$ satisfies all the players and is cheaper than OPT. Therefore, the players in some of the connected components $C_1, C_2, \ldots, C_k$ are not in the same connected component with i in $G\left(p^* - p_i + \chi_i(\overline{p_i})\right)$.

Let K be the subset of connected components $C_1, C_2, \ldots, C_k$ the players of which are not in the same connected component with i in $G\left(p^* - p_i + \chi_i(\overline{p_i})\right)$. Let $e_1, \ldots, e_d$ be the edges that were unpaid for in the first phase of the algorithm directly above the components in K. Then by Lemma 8, there is a set of players S such that all the players in $T_{e_1}, \ldots, T_{e_d}$ are satisfied in the graph $G(p^* - \sum_{l \in S} p_l + \sum_{l \in S} \chi_l(\overline{p_l}, f_l)) - \overline{T_{e_1, e_2, \ldots, e_d}}$. We claim that the subgraph $R = G(p^* - \sum_{j \in S \cup i} p_j + \sum_{j \in S \cup i} \chi_j(\overline{p_j}, f_j))$ satisfies all the players in T and is cheaper than OPT. The latter is clear since $|\chi_i(\overline{p_i})| < l_i = |p_i|$. All players in C and in the subtrees below $C_j \notin K$ are satisfied in R, since they are satisfied in $G\left(p^* - p_i + \chi_i(\overline{p_i})\right)$, and by construction, the payments

p_j for $j \in T_{e_1}, \ldots, T_{e_d}$ do not contain edges of $\overline{T_{e_1,e_2,\ldots,e_d}}$. All players in components of K are satisfied in R as well, since the only edges missing from R that were in $G(p^* - \sum_{j \in S} p_j + \sum_{j \in S} \chi_j(\overline{p_j}, f_j)) - \overline{T_{e_1,e_2,\ldots,e_d}}$ are edges of p_i, which are all edges of $\overline{T_{e_1,e_2,\ldots,e_d}}$. Therefore, all the edges are still there that are needed to make the players in K (and the subtrees below them) be satisfied. Since we constructed a subgraph that satisfies all players and is cheaper than OPT, we have a contradiction. ■

Lemma 12 *Price of stability is* 1 *for the Group Network Formation of Couples Game even if some of the edges of OPT are witnessed from one side.*

Proof. We will show the result by slightly modifying the first phase of the algorithm. Recall that by Corollary 1, the edges witnessed from 2-sides form a connected component of T, which we will refer to as D, and so we root the tree T at a node in D. Observe that there exists a subset S of nodes of D such that all the edges that are witnessed from 1-side are subtrees of T rooted at the nodes of S. If OPT has edges witnessed from 1-side, i.e., not all nodes of S are leaf nodes, we ask the players to buy the edges witnessed from 1-side first. Specifically, for each $u \in S$, we ask the players in the subtree of 1-sided edges rooted at u to buy all the edges in their subtree, using the algorithm from [4] for the Single Source Connection Game. However, we ask only one player per terminal node to form payments in this algorithm. The proof that this set of players can indeed buy all the edges of this subtree using stable payments is exactly the same as the proof of the Single Source Connection Game payment algorithm so we will not repeat it here. Specifically, we can reduce this problem to the Single Source Connection Game by contracting $\overline{T_u}$ and T' into a single node, since every player in T_u must connect either to $\overline{T_u}$ or T' in order to be satisfied.

Once we have formed the payment on the edges witnessed from 1-side, we use our payment algorithm. If u is a terminal node, one of the players at u pays for the higher level incident edge of u in D. Let us now consider the case where u is a high degree nonterminal node. If the subtree composed of edges that are witnessed from 1-side has more than one terminal nodes, we contract the subtree and ask a second player j in one of these terminal nodes to pay for the higher level incident edge of u. The key observation here is that there are at least 2 players in the subtree (one per each terminal node) that did not make any payment yet, and so Lemma 10 still holds right after player j makes her payments. If the subtree has only one terminal node then the payment for the higher level incident edge depends on the number of lower level incident edges of u that are bought. If only one of the lower level incident edges of u is bought, the one that belongs to the subtree of edges witnessed from 1-side, then we ask this player that bought all the edges of the subtree to pay for the higher level incident edge of u. If at least 2 of the lower level incident edges of u are bought then 2 connected components of bought edges merge at u and therefore we ask a player that has not made payment yet to pay for the higher level incident edge of u. ■

5 Computing Equilibria in Polynomial Time

The proof of our 2-approximate Nash equilibrium result suggests an algorithm which forms a cheaper network whenever a 2-approximate Nash equilibrium cannot be found. Using techniques similar to [4], this allows us to form efficient algorithms to compute approximate equilibria:

Theorem 3 *Suppose we are given a feasible solution G_α whose cost is within a factor α of OPT. Then for any $\epsilon > 0$, there is a polynomial time algorithm which returns a $3.1(1+\epsilon)$-approximate Nash equilibrium on a feasible graph G', where $cost\,(G') \leq cost\,(G_\alpha)$. Furthermore, if all the nodes are player nodes, there is a polynomial time algorithm which returns a $(1+\epsilon)$-approximate Nash equilibrium on a feasible graph G', where $cost\,(G') \leq cost\,(G_\alpha)$.*

Proof. We will first prove that given an α-approximation to the socially optimal graph G_α for an instance of the Group Network Formation Game where all nodes are player nodes and any $\epsilon > 0$, there is a polynomial time algorithm which returns a $(1+\epsilon)$-approximate Nash equilibrium on a feasible graph G', where $cost\,(G') \leq cost\,(G_\alpha)$.

To define our algorithm, recall that the proof in Section 3 followed by constructing a network cheaper than the given one and the proof ended up with contradiction since the network at hand was optimal. The proof for obtaining a $(1+\epsilon)$-approximate Nash equilibrium in polynomial time on a given α-approximate socially optimal network G_α is based on following this suggested algorithm to obtain a cheaper network whenever a Nash equilibrium cannot be found. However, the improvements we consider should be substantial enough to ensure the time-bound, while they should be small enough to ensure the approximation ratio.

To find a $(1+\epsilon)$-approximate Nash equilibrium, i.e., a solution where no player can reduce its cost by more than a factor of $(1+\epsilon)$ by taking any deviation, we start by defining $\gamma = \frac{c(G_\alpha)\epsilon}{\alpha(1+\epsilon)m}$, where m is the total number of edges of the graph G. We now use our payment algorithms to pay for all but γ of each edge in G_α. Since G_α is not optimal, it is possible that even with the γ reduction in price, a player may not pay for the cost of an incident one-sided edge or remaining cost of all her lower level incident edges, i.e., that Algorithm 1 could execute the 'break' command. However, the proofs of Lemma 4 and Lemma 3 indicate how we can rearrange G_α to reduce its cost. If we modify G_α in this manner, it is easy to show that we have reduced the cost by at least γ.

Observe that each call to our payment algorithm takes polynomial time if we can compute the best deviation of a player $\chi_i(\overline{p_i}, e)$ in polynomial time. Recall that $\chi_i(\overline{p_i}, e)$ is the cheapest set of edges (using modified costs) that fulfills i's connectivity requirements (see the beginning of Section 3 for a discussion of modified costs). By Lemma 4, $\chi_i(\overline{p_i}, e)$ is the cheapest set of edges (using modified costs) that connects player i to either to a different connected component T' of G_α or to all other terminals of T. To find the best deviation of player i, all we need to do is to find the cheapest deviation that connects i to a different connected component T' of G_α and the cheapest deviation of player i than connects i to all other terminals of T separately, and then take the cheaper one. The former one is essentially computing the shortest path from player i to any of the different connected components T' (using modified costs for the edges) and can be done in polynomial time. Computing the cheapest deviation of player i that connects i to all other terminals of T we do the following. We obtain a new graph by merging the terminal i with all other trees T' of G_α. Computing the cheapest deviation of player i that connects i to all other terminals of T corresponds to connecting all connected components of this new graph, since every node of the graph is a terminal, which can be done in polynomial time since this problem is the minimum spanning tree problem.

Thus we know that the Algorithm 1 can be made to run in poly-time, and that at every call to this algorithm it either forms a Nash equilibrium, or returns a solution that is cheaper by at least γ. Since each call which fails to form a Nash equilibrium reduces the cost by γ, we can have at most $\frac{\alpha(1+\epsilon)m}{\epsilon}$ calls. And since each call to our payment algorithm can be made to run in polynomial time, we obtained a network G' with $c(G') \leq c(G_\alpha)$ such that we have a Nash equilibrium on G' if the cost of its edges were decreased by γ in time polynomial in m and ϵ^{-1}.

For all payment strategies p_i and for each edge e in G', we now increase $p_i(e)$ in proportion to $|p_i|$ so that e is now fully paid for. Now clearly G' is fully paid for. To show that we obtained a $(1+\epsilon)$-approximate Nash equilibrium, all we need to show is that each stable strategy p_i became $(1+\epsilon)$-approximately stable after they are proportionally increased.

Observe that the payment player i makes is increased to $\frac{c(G')|p_i|}{c(G')-m'\gamma}$, where m' denotes the number of edges in G'. To see that this is $(1+\epsilon)$-approximate Nash equilibrium, note that p_i was a stable payment before it was increased and therefore $\chi_i(\overline{p_i})$, deviation of player i with respect to p_i, was as expensive as p_i. Therefore, by deviating with respect to p_i, player i can gain at most a factor of

$$\frac{c(G')}{c(G') - m'\gamma} \leq \frac{c(G')}{c(G') - \frac{m' c(G_\alpha)\epsilon}{\alpha(1+\epsilon)m}} \leq \frac{c(G')}{c(G') - \frac{c(G')\epsilon}{(1+\epsilon)}} = (1+\epsilon).$$

We have given a polynomial-time algorithm which returns a $(1+\epsilon)$-approximate Nash equilibrium on a feasible graph G', where $cost(G') \leq cost(G_\alpha)$ for the Group Network Formation Game where all nodes are terminals.

The same algorithm and the proof holds for the Group Network Formation of Couples Game except that computing the cheapest deviation does not reduce to minimum spanning tree but to minimum Steiner tree problem and therefore, the best deviation of a player cannot be computed in polynomial time. But since the Steiner tree problem can be provably approximated within a factor of $1 + \frac{\ln 3}{2} \approx 1.55$ by a polynomial algorithm[22], we will use the output of this algorithm instead of the cheapest deviation to decide the payment on the edges. Note that we obtain a network G' with $c(G') \leq c(G_\alpha)$ such that we have a 1.55-approximate Nash equilibrium on G' if the cost of its edges were decreased by γ in time polynomial in m and ϵ^{-1}. Similarly to the case when all nodes are terminals, the payment of each player will be increased by a factor

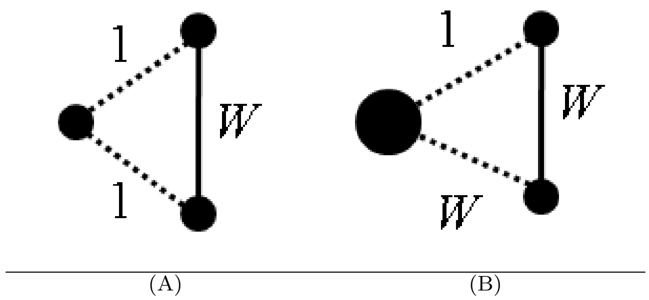

Fig. 4 (A) An instance of a game where there is no approximate Nash equilibria if the happiness function is not monotone. (B) An instance of a Group Network Formation Game where there is no approximate Nash equilibrium on OPT if the fair sharing cost-sharing mechanism is used.

of at most $(1+\epsilon)$ after the payments on the edges are raised proportionally to cover the whole cost of the edges of G'. Therefore, we have given a polynomial-time algorithm which returns a $(1.55+\epsilon)$-approximate Nash equilibrium on a feasible graph G', where $cost\,(G') \leq cost\,(G_\alpha)$ for the Group Network Formation of Couples Game.

Given an instance $\Im_1 = (N_1, G, T, F)$ of Group Network Formation Game, we can obtain an instance $\Im_2 = (N_2, G, T, F)$ of Group Network Formation of Couples Game that has twice as many players on the same graph G, with the same set of terminals T and the same happiness function F such that each player $i \in N_1$ has 2 corresponding players $j, k \in N_2$ as illustrated in the proof of Theorem 1. The strategy $p_i = p_j + p_k$ is $3.1(1+\epsilon)$-approximately stable for player i since the costs of both p_j and p_k are within a factor of $(1.55+\epsilon)$ of the cost of $\chi_i(p_i)$. ∎

Since for all monotone functions F, finding OPT is a constrained forest problem [14], then Theorem 3 gives us a poly-time algorithm for $\alpha = 2$. In other words, we can find a $(3.1+\epsilon)$-approximate Nash equilibrium with cost at most $2 \cdot OPT$ in polynomial time.

6 Inapproximability Results and Terminal Backup

Recall that in this paper, we consider games where the happiness functions are monotone. Theorem 4 shows that this property of happiness functions is critical for even approximate stability.

Theorem 4 *For the Group Network Formation Game where the happiness functions may not be monotone, there is no α-approximate Nash equilibrium for any α.*

Proof. To prove that there is no approximate Nash equilibrium for the Group Network Formation Game we give such an instance of the problem as shown in Figure 4(A). All nodes are player nodes. We define a component to be happy if all the players in it are happy according to the following connectivity requirements: The player on the left is happy if it is connected to at most one other terminal. The 2 players on the right are happy if and only if they are connected to exactly 1 other terminal. The cost of each edge is as given in Figure 4(A), where W is very large. Observe that there is only one feasible solution of the game, i.e., all the players are happy and where only the edge whose cost is W is purchased. The player on the left cannot contribute to the cost of the edge in any approximate Nash equilibrium since it is already happy and therefore has a deviation of cost 0. Since the other 2 players have to be sharing the cost of W, at least one of them should be paying at least $W/2$. Since that player has a deviation of cost 1, at least one of the players can reduce its cost by a factor of $W/2$ by deviating and therefore there is no α-approximate Nash equilibrium for this game where $\alpha < W/2$. Since W can be arbitrarily large and it is independent of the number of players, there is no approximate Nash equilibrium for that instance of the general Group Network Formation Game. ∎

Recall that congestion games, including our game with fair sharing, are guaranteed to have Nash equilibria, although they may be expensive. The following theorem studies the quality (cost) of approximate Nash equilibrium and shows that there may not be any approximately stable solution that is as cheap as the socially optimal network.

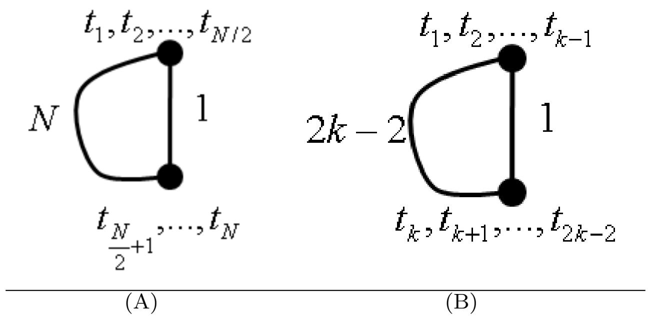

Fig. 5 (A) An instance of a Group Network Formation Game that has a Nash equilibrium whose cost is N times more than the socially optimal network. (B) An instance of Terminal Backup problem that has a Nash equilibrium whose cost is $2k-2$ times more than the socially optimal network.

Theorem 5 *For the Group Network Formation Game, there may not be any approximate Nash equilibrium whose cost is as much as OPT if the fair cost-sharing mechanism is used.*

Proof. In Figure 4(B), all nodes are player nodes. We define a component to be happy if all the players in it are happy according to the following connectivity requirements: The large node on the left is happy always, and the 2 nodes on the right want to connect to at least one other terminal. In OPT, the 2 players on the right would have to buy the edge between them which has a cost of W. When the fair sharing cost scheme is used, both of the nodes have to make a payment of $W/2$ on that edge even though the top player has a deviation of cost 1. Since the top player can reduce its cost by a factor of $W/2$ by deviating, there is no α-approximate Nash equilibrium for this game where $\alpha < W/2$. Since W can be arbitrarily large and it is independent of the number of players, there is no approximate Nash equilibrium on OPT. ■

Because of its applications to multi-homing [5,25], we are especially interested in the behavior of Terminal Backup connectivity requirements, i.e., when a player node desires to connect to at least $k-1$ other player nodes for a fixed k.

Theorem 6 *For the Group Network Formation Game and the Terminal Backup problem, the Price of Anarchy is n and $2k-2$ respectively. Furthermore, these bounds are tight.*

Proof. The *price of anarchy* refers to the ratio of the worst (most expensive) Nash equilibrium and the optimal centralized solution. In the Group Network Formation Game, the price of anarchy is at most N, the number of players. This is simply because if the worst Nash equilibrium p costs more than N times OPT, the cost of the optimal solution, then there must be a player whose payments in p are strictly more than OPT, so he could deviate by purchasing the entire optimal solution by himself, and form a connected component that makes her happy with smaller payments than before. More importantly, there are cases when the price of anarchy actually equals N. This is demonstrated with the example in Figure 5(A).

Suppose there are N players, and G consists of 2 nodes which are joined by 2 disjoint paths, one of cost 1 and and one of cost N. Half of the players have their terminal at one node and the other half of the players have their terminal at the other node. The only happy components must include both nodes. This corresponds to Terminal backup requirements where every player wants to be connected to at least $N/2+1$ terminals. Then, the worst Nash equilibrium has each player contributing 1 to the long path, and has a cost of N. The optimal solution here has a cost of only 1, so the price of anarchy is N. Therefore, the price of anarchy could be very high in the Group Network Formation Game.

We are also interested in the price of anarchy of the Terminal Backup problem which is a special case of Group Network Formation Games. In Figure 5(B), we give an example where price of anarchy is $2k-2$ and we prove that this bound is tight below. In this case, k is the connectivity requirement: components are happy if and only if they contain at least k terminal nodes.

To prove the result all we need to do is to show that no equilibrium will cost more than $2k-2$ times OPT since Figure 5(B) shows an instance of a Terminal Backup problem with an equilibrium whose cost is exactly $2k-2$ times the cost of OPT.

For the purpose of contradiction, assume there is an instance of a Terminal Backup problem that has an equilibrium whose cost is more than $2k-2$ times the cost of OPT. Let EEQ denote this expensive equilibrium solution. Observe that there must exist a connected component T of OPT such that the total payments of the players of T in the expensive equilibrium solution EEQ is more than $2k-2$ times the cost of T. Since EEQ is a Nash equilibrium, no player in T pays more than the cost of T. Therefore, T includes more than $2k-2$ terminals. However, as we have shown in the proof of Theorem 7, there exists a set of edges C_i for every player i such that $C_i \subseteq T$, C_i contains at least k terminals including i, and every edge of T is contained in no more than $2k-2$ sets C_i.

We define α_i be the cost of C_i, which is at least the cost of connecting a player i in T to at least $k-1$ other players in T. Since no edge of T is used by more than $2k-2$ sets C_i, we know that $\sum_{i\in T}\alpha_i \leq (2k-2)\,c(T)$ where $c(T)$ denotes the cost of the component T.

However, in EEQ, the total payment of all the players in T is more than $2k-2$ times the cost of T, i.e., $\sum_{i\in T}|p_i| > (2k-2)\,c(T)$. Therefore, $\sum_{i\in T}|p_i| > \sum_{i\in T}\alpha_i$ which implies that there exists a player i in T such that $|p_i| > \alpha_i$. Therefore, EEQ cannot be a Nash equilibrium since player i can reduce its cost to α_i by deviating.

The lower bounds for Terminal Backup also hold for the general Group Network Formation Game, showing that while the price of stability may be low, the price of anarchy can be as high as the number of players. ■

Theorem 7 *For the Terminal Backup problem, with the Shapley cost-sharing (fair sharing) payment scheme, the price of stability is at most $H(2k-2)$ where $H(2k-2)$ denotes the $(2k-2)^{nd}$ harmonic number.*

Proof. The Terminal Backup problem, under the Shapley cost-sharing model (or fair sharing), falls into a class of games called congestion games. In this model, the strategy of a player i is just a set of edges S_i that contains at least k terminals including i, and the edges that are built are $\cup_i S_i$. The players split the cost of every edge evenly, i.e., a player i using edge $e \in S_i$ must pay $c(e)/x_e$, where $x_e = |\{S_i|e \in S_i\}|$. [3] showed that in a network design game with the Shapley-cost sharing mechanism, the cost of the Nash equilibrium obtained by the best-response dynamics starting from OPT is at most $H(n)$ times more expensive than OPT, where $H(n)$ is the n^{th} harmonic number and n is the maximum number of players using a single edge on OPT for their connectivity requirements. Therefore, to obtain a price of stability bound for the Terminal Backup problem, all we need to do is to select sets S_i for the players in OPT while ensuring that no edge appears in too many sets S_i.

To prove the result, we need to show that we can select sets S_i for the players on OPT such that no edge is used by more than $2k-2$ players. We first start by replacing all the edges of OPT by 2 directed edges. We can now form an Euler tour in each connected component of OPT since every vertex has an equal number of incoming and outgoing edges incident to it. Then, we ask each player i to follow the Euler tour in the clockwise direction until it encounters $k-1$ other distinct terminals. We set S_i to be this set of edges. Observe that each directed edge appears in at most $k-1$ sets S_i belonging to distinct closest players in the counterclockwise direction. Since each undirected edge of OPT had been replaced with 2 directed edges, each edge of OPT is used by at most $2k-2$ sets S_i. ■

References

1. E. Anshelevich and B. Caskurlu. Exact and Approximate Equilibria for Optimal Group Network Formation. In *Proc. 17th Annual European Symposium on Algorithms* (ESA 2009).
2. E. Anshelevich and B. Caskurlu. Price of Stability in Survivable Network Design. In *Proc. 2nd Internationonal Symposium on Algorithmic Game Theory* (SAGT 2009).
3. E. Anshelevich, A. Dasgupta, J. Kleinberg, É. Tardos, T. Wexler, T. Roughgarden. The Price of Stability for Network Design with Fair Cost Allocation. *SIAM Journal on Computing,* Volume 38, Issue 4 (November 2008), pp. 1602-1623.
4. E. Anshelevich, A. Dasgupta, É. Tardos, T. Wexler. Near-Optimal Network Design with Selfish Agents. In *Theory of Computing*, Volume 4 (2008), pp. 77-109.
5. E. Anshelevich and A. Karagiozova. Terminal Backup, 3D Matching, and Covering Cubic Graphs. In *Proc. 39th ACM Symposium on Theory of Computing* (STOC 2007).
6. C. Chekuri, J. Chuzhoy, L. Lewin-Eytan, J. Naor, and A. Orda. Non-cooperative multicast and facility location games. Proceedings of the 7th ACM Conference on Electronic Commerce (EC), Ann Arbor, Michigan (2006), pp. 72-81.
7. H. Chen, T. Roughgarden. Network Design with Weighted Players. In *SPAA* 2006.
8. H. Chen, T. Roughgarden, and G. Valiant. Designing Networks with Good Equilibria. *SODA 2008.*

9. G. Christodoulou and E. Koutsoupias. On the price of anarchy and stability of correlated equilibria of linear congestion games. ESA, 2005.
10. A. Epstein, M. Feldman, and Y. Mansour. Strong Equilibrium in Cost-Sharing Connection Games. *EC* 2007.
11. A. Fabrikant, C. Papadimitriou, K. Talwar. The complexity of pure Nash equilibria. In *STOC*, 2004.
12. A. Fiat, H. Kaplan, M. Levy, S. Olonetsky, R. Shabo. On the Price of Stability for Designing Undirected Networks with Fair Cost Allocations. *Proceedings of ICALP* 2006, pp. 608–618.
13. N. Garg, G. Konjevod, R. Ravi. A polylogarithmic approximation algorithm for the group Steiner tree problem. *SODA 2000*.
14. M. X. Goemans and D. P. Williamson. The primal-dual method for approximation algorithms and its application to network design problems. In *Approximation Algorithms for NP-Hard Problems*, edited by D. S. Hochbaum, PWS Publishing Company (1997).
15. M. Hoefer. Non-cooperative Facility Location and Covering Games. In *ISAAC* 2006.
16. M. Hoefer. Non-cooperative Tree Creation. *Algorithmica* 53(1), pp. 104–131, 2009.
17. M. Hoefer, P. Krysta. Geometric Network Design with Selfish Agents. *COCOON* 2005.
18. R. Holzman, N. Law-Yone. Strong Equilibrium in congestion games. *Games and Economic Behavior,* 21, 1997.
19. M. Jackson, A survey of models of network formation: stability and efficiency. *Group Formation in Economics: Networks, Clubs and Coalitions*, eds. G. Demange and M.Wooders, Cambridge Univ. Press.
20. D. Monderer, L. Shapley. Potential Games. *Games and Economic Behavior* 14(1996), 124–143.
21. N. Nisan, T. Roughgarden, É. Tardos, and V. V. Vazirani (eds.), *Algorithmic Game Theory,* Cambridge University Press.
22. G. Robins and A. Zelikovsky. Tighter bounds for graph Steiner tree approximation. *SIAM Journal on Discrete Mathematics*, 19(1), pp. 122–134, 2006.
23. R. W. Rosenthal. The network equilibrium problem in integers. *Networks*, 3:53-59, 1973.
24. T. Roughgarden. *Selfish Routing and the Price of Anarchy,* MIT Press.
25. D. Xu, E. Anshelevich, and M. Chiang. On Survivable Access Network Design: Complexity and Algorithms. *INFOCOM 2008.*