Social Distancing Network Creation

Tobias Friedrich ✉
Hasso Plattner Institute, University of Potsdam, Germany

Hans Gawendowicz ✉
Hasso Plattner Institute, University of Potsdam, Germany

Pascal Lenzner ✉
Hasso Plattner Institute, University of Potsdam, Germany

Anna Melnichenko ✉
Hasso Plattner Institute, University of Potsdam, Germany

──── **Abstract** ────

During a pandemic people have to find a trade-off between meeting others and staying safely at home. While meeting others is pleasant, it also increases the risk of infection. We consider this dilemma by introducing a game-theoretic network creation model in which selfish agents can form bilateral connections. They benefit from network neighbors, but at the same time, they want to maximize their distance to all other agents. This models the inherent conflict that social distancing rules impose on the behavior of selfish agents in a social network. Besides addressing this familiar issue, our model can be seen as the inverse to the well-studied Network Creation Game by Fabrikant et al. [PODC 2003] where agents aim at being as central as possible in the created network. Thus, our work is in-line with studies that compare minimization problems with their maximization versions.

We look at two variants of network creation governed by social distancing. In the first variant, there are no restrictions on the connections being formed. We characterize optimal and equilibrium networks, and we derive asymptotically tight bounds on the Price of Anarchy and Price of Stability. The second variant is the model's generalization that allows restrictions on the connections that can be formed. As our main result, we prove that Swap-Maximal Routing-Cost Spanning Trees, an efficiently computable weaker variant of Maximum Routing-Cost Spanning Trees, actually resemble equilibria for a significant range of the parameter space. Moreover, we give almost tight bounds on the Price of Anarchy and Price of Stability. These results imply that, compared the well-studied inverse models, under social distancing the agents' selfish behavior has a significantly stronger impact on the quality of the equilibria, i.e., allowing socially much worse stable states.

1 Introduction

Network Design is a core topic in Theoretical Computer Science and Operations Research. Many classical combinatorial optimization problems, inspired by real world applications, have been formulated and analyzed, such as the MINIMUM SPANNING TREE problem [28], the NETWORK DESIGN problem [34, 40] and finding geometric spanners [14, 45]. Typically, a network having certain properties must be found by a centralized algorithm. However, in many settings, the desired network is not created by a central authority but by individually acting agents, e.g., people or institutions, controlling a local part of the network. Prominent examples are the Internet, road networks, and, most relevant for our work, social networks.

Especially in settings with little coordination, these individual agents tend to selfishly optimize their own utility without taking the impact of their actions on the efficiency of the

whole network into account. To better understand the dynamics arising in these decentralized settings and the network structures resulting from them, many influential game-theoretic network formation models have been introduced in the last decades [32, 9, 23, 7, 8]. The main research questions are: Do equilibrium networks, i.e., stable networks where no agent can improve by performing a local change, exist? What properties do these networks have? And how efficient are they compared to centrally computed optimal solutions?

All of the above mentioned influential game-theoretic network formation models assume that the creation of an edge is costly but the agents benefit from having small distances to other agents in the network. However, departing from this standard assumption in the field, there are real-world settings that should better be modeled via an inverted utility function: neighbors yield benefit but being close to many agents is costly as it yields an increased risk. One example for this choice are financial networks. There, financial institutions benefit from working together but suffer from risks arising from one of them failing[1]. Another example, that is the main motivation of our work, came up with the current COVID-19 pandemic and is described by the now commonly used term *social distancing.* It refers to reducing social contacts in order to contain the spread of a contagious virus in the population. While often mandated by the government, social distancing was performed by many people voluntarily. One of the main reasons is quite simple: While reducing social contacts is a restriction of the quality of life, it also reduces the probability of getting infected. Hence, the network of social interactions between people was sparsified by individual strategic decisions.

In this work we introduce a novel game-theoretic network formation model in which selfish agents strategically form a social network under the influence of social distancing. Agents benefit from direct connections to other agents, modeling the positive effects of social contacts on their social life. However, at the same time they want to maximize their distances to all other agents in the network in order to reduce their risk of getting infected via an increased reaction time in case a contagious disease starts spreading in the network. Here we assume that a random network node becomes infected and that it is beneficial to be far away from the source of infection in order to gain valuable time for setting up counter-measures.

The agents in our model act according to an inverted utility function, compared to the famous models by Jackson and Wolinsky [32] and Fabrikant et al. [23]. Thus, to the best of our knowledge, this is one of the rare cases of a game-theoretic model where both minimizing and maximizing the utility function has a natural interpretation. Another similar well-known example is the contrast between the Network Design Game with fair cost sharing by Anshelevich et al. [7] and the Selfish Routing model by Roughgarden and Tardos [46]. In both models the agents select paths in a given network but in the former sharing an edge is beneficial for the involved agents whereas in the latter edge sharing is detrimental. This difference yields vastly different behavior in terms of the quality of the equilibria. However, this is not obvious, as can also be seen by comparing classical minimization and maximization variants of optimization problems, e.g., MINIMUM SPANNING TREE versus MAXIMUM SPANNING TREE or SHORTEST PATH versus LONGEST PATH. Sometimes, as with spanning trees, the inverse problems are almost identical, whereas sometimes, as with the path problems, the inverse problems may have completely opposite behavior. We set out to explore this comparison for the natural inverse counter-part to the well-known Network Creation Game by Fabrikant et al. [23]. Along the way, we will uncover a connection to the MAXIMUM ROUTING-COST SPANNING TREE problem that is inverse to the well-studied

---

[1] The financial crisis in the late 2000s was mainly driven by contagious network effects of failing banks.

MINIMUM ROUTING-COST SPANNING TREE problem [30][2].

1.1 Model and Notation

Before we start with the model definition, we introduce some notation regarding networks. A *network* is a tuple $G \coloneqq (V, E)$ where V is the set of *nodes* and E is the set of *edges*. An *edge* is represented by a set containing both incident nodes. If we do not give the tuple defining G explicitly, we denote the set of nodes of G as V_G and the set edges of G as E_G. We only consider unweighted undirected networks. For addition and removal of a single edge e, we write $G + e \coloneqq (V, E \cup \{e\})$ and $G - e \coloneqq (V, E \setminus \{e\})$. A network G' with $V_{G'} \subseteq V$ and $E_{G'} \subseteq E$ is called a *subnetwork* of G and denoted as $G' \leq G$. If G' is connected and $V_{G'} = V$, G' is a *spanning subnetwork* of G. Let $n \in \mathbb{N}$ denote the number of nodes. The set of all connected networks containing exactly n nodes will be referred to as $\mathcal{G}_n$.

For two nodes $v, x \in V$, we define $d_G(v, x)$ as the *distance* between v and x in network G, that is, the number of edges on a shortest path from v to x in G. For convenience, we extend the definition of d_G to sets of nodes: Let $v \in V$ be a node and $M, N \subseteq V$ be sets of nodes. Then $d_G(v, M) \coloneqq \sum_{x \in M} d_G(v, x)$ and $d_G(M, N) \coloneqq \sum_{x \in M, y \in N} d_G(x, y)$. We call the special case $d_G(v, V)$ the *distances from/for v* and $d_G(V, V)$ the *total/summed distances* or *routing costs* of G. The *degree* of v in the network G is the number of edges that are incident to v and is denoted as $\deg_G(v)$. We call a tree which is a spanning subnetwork of G a *spanning tree* of G. A spanning tree of G with routing costs at least as high as the routing costs of any other spanning tree of G will be called a *Maximum Routing-Cost Spanning Tree* (MRCST). A spanning tree of G with routing costs that cannot be increased by swapping one edge is a *Swap-Maximal Routing-Cost Spanning Tree* (SMRCST).

Now, we can define the game-theoretic model. Let $H = (V, E)$ be a connected network. We call H the *host network* and its nodes *agents*. A *state* of the game $G \leq H$ is a spanning subnetwork of H. We only consider connected networks as host networks and states.

Each agent $v \in V$ selfishly tries to maximize its utility in state G given by

$$u_v(G) \coloneqq \alpha \deg_G(v) + d_G(v, V)$$

where $\alpha \in \mathbb{R}_{>0}$ is a global parameter. We will call $\alpha \deg_G(v)$ the *edge utility* and $d_G(v, V)$ the *distance utility* of v. Note that α is a parameter of the game, i.e., equal for all agents, that allows to adjust the agents' trade-offs between edge utility and distance utility. Here α is the benefit of a single edge, i.e., the benefit for each direct neighbor in the network.

For measuring the efficiency of the network G, we use the *social welfare* defined as $\mathrm{SW}(G) \coloneqq \sum_{v \in V_G} u_v(G) = 2\alpha |E_G| + d_G(V, V)$. This quantifies the well-being of the society of all agents. We call a network maximizing the social welfare for the host network H a *social optimum* and denote it as OPT_H.

Agents are allowed to form connections bilaterally. More specifically, each agent can unilaterally remove any incident edge if it does not disconnect the network, and two agents together can form an edge between them if it is contained in the host network. If removing an edge strictly increases the utility of one of its incident nodes or adding an edge strictly increases the utility of both incident nodes, we call this an *improving move*. A network without improving moves is referred to as *pairwise stable* [32] or *stable* for short[3]. If there

---

[2] This problem is also known as the OPTIMUM COMMUNICATION SPANNING TREE problem.
[3] As shown by Corbo and Parkes [18] for bilateral Network Creation Games, pairwise stability is equivalent to pairwise Nash stability, which is a refinement of the Nash equilibrium: it must be stable against

are no improving edge additions or removals we call the network *stable against edge addition* and *stable against edge removal*, respectively.

For a host network H, we define $\mathcal{S}(H)$ as the set of all pairwise stable states. For measuring the efficiency lost by letting agents form the network selfishly, we use the *Price of Anarchy* (PoA) [37] and *Price of Stability* (PoS) [7] defined as

$$\text{PoA}_n := \max_{H \in \mathcal{G}_n} \max_{G \in \mathcal{S}(H)} \frac{\text{SW(OPT}_H)}{\text{SW}(G)} \quad \text{and} \quad \text{PoS}_n := \max_{H \in \mathcal{G}_n} \min_{G \in \mathcal{S}(H)} \frac{\text{SW(OPT}_H)}{\text{SW}(G)}.$$

We will call this model *Social Distancing Network Creation Game* (SDNCG). In Section 2 we will restrict the host networks to complete networks K_n. We will call this restricted variant *complete Social Distancing Network Creation Game* (K-SDNCG).

1.2 Related Work

Variants of game-theoretic network formation models have been studied extensively for decades and we refer to Jackson [31] for an overview.

Closest to our work is the literature on the Network Creation Game (NCG) by Fabrikant et al. [23]. This influential model can be seen as the unilateral inverted variant of the K-SDNCG. There, an agent can buy any incident edge without the consent of the other endpoint for the price of $\alpha > 0$. Each agent aims at minimizing its cost, which is defined as the sum of α times the number of bought edges and the sum of hop-distances to all agents. The authors of [23] show that Nash equilibria always exist, i.e., complete networks are stable for $\alpha \leq 2$ and stars are stable for $\alpha \geq 2$. However, besides these generic examples finding Nash equilibria is challenging since the NCG and many of its variants do not belong to the class of potential games [38, 36]. Besides finding equilibria, also computing a best possible strategy is challenging, since this problem was shown to be NP-hard in [23]. However, such strategies can be efficiently approximated with greedy strategy changes [39]. Regarding the quality of equilibrium states the authors of [23] show that the PoA is in $\mathcal{O}(\sqrt{\alpha})$, that the PoA for tree Nash equilibria is constant, and that the PoS is at most $\frac{4}{3}$. Later, a series of papers [2, 20, 42, 41, 4, 12, 5] improved the PoA bounds, with the best general upper bound of $2^{\mathcal{O}\sqrt{\log n}}$ by Demaine et al. [20]. The latter also proved that the PoA is constant for $\alpha \in \mathcal{O}(n^{1-\varepsilon})$ for any fixed $\varepsilon > \frac{1}{\log n}$. For large α, it was shown by Bilò and Lenzner [12] that for $\alpha > 4n - 13$ all Nash equilibria must be trees and this bound was recently improved by Dippel and Vetta [21] to $\alpha > 3n - 3$. This implies a constant PoA for $\alpha > 3n - 3$. Finally, Álvarez and Messegué [5] established a constant PoA for $\alpha > n(1 + \varepsilon)$, for any $\varepsilon > 0$.

The NCG was generalized by Demaine et al. [19] by introducing a host network that specifies which edges can be bought. They show that the PoA deteriorates by providing a lower bound of $\Omega(\min\{\alpha/n, n^2/\alpha\})$ and an upper bound of $\mathcal{O}(\sqrt{\alpha})$, for $\alpha < n$, and $\mathcal{O}(\min\{\sqrt{n}, n^2/\alpha\})$, for $\alpha \geq n$. Interestingly, no results on the existence of equilibria are known. Recently, a further generalization that allows weighted host networks was proposed by Bilò et al. [11]. This variant has a tight PoA of $(\alpha + 2)/2$ for metric weights. Later a tight bound of $\Theta(\alpha)$ was shown for arbitrary weights [24]. Also a bilateral variant of the NCG was studied by Corbo and Parkes [18]. There, similar to our model, edges can only be established by bilateral consent of the involved nodes and both nodes have to pay α. The

---

unilateral deviations and it must be stable against joint strategy changes by coalitions of agents of size two. The strategy space of any agent $i \in V$ is the power set of $V \setminus i$. An edge $\{u, v\}$ is formed if and only if v is in agent u's strategy and u is in agent v's strategy.

authors of [18] prove existence of pairwise stable networks, i.e., complete networks are stable for $\alpha \leq 1$ and stars are stable for $\alpha \geq 1$, they give a tight PoA bound of $\Theta(\min\{\sqrt{\alpha}, n/\sqrt{\alpha}\})$, and they show that the PoS is 1. To the best of our knowledge, the bilateral variant with a given host network has not yet been studied. Recently, also a bilateral variant modeling the formation of social networks was introduced [10].

The idea of a game-theoretic model of network formation in a context of spreading risk is not new. Goyal et al. [27] study a setting where a node is attacked and this attack spreads to all vulnerable neighbors. Agents strategically create edges and immunize themselves to maximize their connected component post attack. For this model, also the efficient computation of best strategies [25] and a variant with probabilistic spread [17] was studied. Moreover, there has been much research in the context of financial contagion, where agents benefit from collaborating, but also suffer from the risk of cascading failure arising with the collaboration [3, 29, 15, 1]. In particular, Blume et al. [13] developed an elegant model where nodes form a network and then some randomly chosen nodes fail and this failure then spreads with some probability via the edges. The utility is a linear combination of the node degree and the risk of failing in the second phase. The virtue of this model is that utilities are based on a random process that realistically models the spread of a contagious infection. However, the major downside of this model is that the computation of the random process is #P-complete. Thus, this model does not yield a realistic prediction of real-world behavior.

While analyzing our model for general host networks, we consider Maximum Routing-Cost Spanning Trees. Routing costs have been studied much in mathematics, mostly under the name of the *Wiener index* [47]. Trees were of special interest and there has been much research on the Wiener index of trees with different properties. But although spanning trees minimizing the Wiener index were studied extensively, the concept of spanning trees maximizing the Wiener index received little attention [22, 48]. However, it was shown that finding or even approximating a tree maximizing the Wiener index is NP-hard [16, 26].

1.3 Our Contribution

We introduce the Social Distancing Network Creation Game (SDNCG), a game-theoretic model in which selfish agents try to maximize their utility by strategically connecting to other agents and thereby creating a network. Each agent values direct connections to other agents but at the same time wants to maximize the distances to all other agents in order to lower their exposure and increase their reaction time to risks appearing in the network. In contrast to the similar model by Blume et al. [13], our model, while not modeling a perfectly realistic spread of the infection, has the advantage of an efficiently computable utility function. By using the distance to the other agents as part of the utility, it also accounts for reaction time: If an infection breaks out far away, an agent has more time to prepare or react to it. Another virtue of our model is that it is the inverse to the well-known Network Creation Game [23] and its bilateral variant [18]. Hence, we can study and compare the game-theoretic properties of the inverted models. To the best of our knowledge, this is one of the rare cases where both the minimization and the maximization of a utility function have a natural interpretation.

Our results and the comparison with the inverted models are summarized in Table 1. We analyze two variants of the SDNCG. For the K-SDNCG, where we assume a complete host network, we characterize optimal and several stable networks and show that the PoS is 1. We provide an improving response cycle, which implies that equilibrium existence for the (K-)SDNCG cannot be derived from potential function arguments. Finally, derive several bounds for the PoA which are tight for $\alpha \geq \frac{n}{2}$, asymptotically tight for $\alpha \leq \sqrt{n}$, and asymptotically tight up to a log-factor for $\alpha \leq \frac{n}{6} - 3$.

	Optimum	Equilibria	PoA	PoS
NCG [23]	$\alpha \leq 2: K_n$ [23] $\alpha \geq 2: S_n$ [23]	$\alpha \leq 1: K_n$ [23] $\alpha \geq 1: S_n$ [23]	$2^{\mathcal{O}\left(\sqrt{\log n}\right)}$ [20] $\alpha \in \mathcal{O}\left(n^{1-\varepsilon}\right): \Theta(1)$ [20] $\alpha > n(1+\varepsilon): \Theta(1)$ [5]	$\alpha \leq 1: 1$ [23] $1<\alpha<2: \leq \frac{4}{3}$ [23] $\alpha \geq 2: 1$ [23]
BNCG [18]	$\alpha < 1: K_n$ [18] $\alpha > 1: S_n$ [18]	$\alpha < 1: K_n$ [18] $\alpha > 1: S_n, \ldots$ [18]	$\Theta\left(\min\left\{\sqrt{\alpha}, \frac{n}{\sqrt{\alpha}}\right\}\right)$ [18] $\alpha < 1: 1$ [18]	1 [18]
K-SDNCG	$\alpha < \frac{n}{3}: P_n$ [T. 1] $\alpha > \frac{n}{3}: K_n$ [T. 1]	$\alpha \leq 1:$ trees [T. 2] $1 \leq \alpha \leq \frac{n}{2}:$ $P_n, K_n, \ldots$ [T. 2] $\alpha \geq \frac{n}{2}: K_n$ [T. 2]	$\mathcal{O}(n)$ [T. 5] $\alpha \leq \sqrt{n}: \Theta(n)$ [T. 5] $\alpha \leq \frac{n}{6} - 3: \Omega\left(\frac{n}{\log n}\right)$ [T. 5] $\alpha \leq \lfloor \frac{n}{2} \rfloor - 2: \Omega\left(\sqrt{n}\right)$ [T. 5] $\alpha \geq \frac{n}{2}: 1$ [T. 5]	1 [T. 6]
H-NCG [19]	open	open	$\alpha < n: \mathcal{O}\left(\sqrt{\alpha}\right)$ [19] $\alpha \geq n: \min\left\{\mathcal{O}\left(\sqrt{n}, \frac{n^2}{\alpha}\right)\right\}$ [19] $\Omega\left(\min\left\{\frac{\alpha}{n}, \frac{n^2}{\alpha}\right\}\right)$ [19]	open
SDNCG	$\alpha \leq 1: \text{MRCST}$ [T. 7] $\alpha > N_3: H$ [T. 7]	$\alpha \leq 1:$ trees [T. 9] $1 \leq \alpha \leq \frac{n}{3}:$ SMRCST [T. 10] $\alpha \geq N_2: H$ [T. 9]	$\mathcal{O}(n)$ [C. 14] $\alpha \leq n: \Theta(n)$ [T.14] $\alpha \leq N_2: \Omega\left(\frac{n^2}{\alpha}\right)$ [T. 14] $N_2 < \alpha \leq N_3: \Theta(1)$ [T. 14] $\alpha \geq N_3: 1$ [T. 14]	$\alpha \leq 1: 1$ [T. 15] $\alpha < \frac{n}{3}: \mathcal{O}\left(\sqrt{n}\right)$ [T. 15] $N_2<\alpha\leq N_3: \Theta(1)$ [T. 15] $\alpha \geq N_3: 1$ [T. 15]

■ **Table 1** An overview of our results (yellow) and a comparison with the results for the inverted models (white). BNCG abbreviates the bilateral NCG by Corbo and Parkes [18] whereas H-NCG denotes the NCG on a host network by Demaine et al. [19]. $N_2 := \frac{(n-1)^2}{4}$, $N_3 := \frac{(n-2)n(n+2)}{24}$, H denotes the host network, P_n, K_n, S_n are the path, clique, and star networks on n nodes, respectively.

For the SDNCG on arbitrary host networks we utilize Maximum Routing-Cost Spanning Trees for characterizing optimal networks for $\alpha \leq 1$. As our main result, we show that their locally optimal variant, the Swap-Maximal Routing-Cost Spanning Trees, and hence also Maximum Routing-Cost Spanning Trees, are pairwise stable for $\alpha \leq \frac{n}{3}$. We prove that computing the MRCST is NP-hard, while the SMRCST can be constructed efficiently. Thus, for the significant range of $1 \leq \alpha \leq \frac{n}{3}$, we not only have guaranteed equilibrium existence on any host graph, but we can compute stable states efficiently. This is in stark contrast to what is known for the inverse model studied by Demaine et al. [19]. Additionally, we approximate optimal networks and we derive several (tight) bounds on the PoA and the PoS.

Compared with the NCG [23] and the bilateral NCG [18], we find that the results for the K-SDNCG regarding optimal and stable networks are analogous but reversed, with the spanning path taking over the role of the spanning star. Moreover, our PoA results for both the K-SDNCG and the SDNCG show that our maximization variant has a significantly worse PoA that is linear or almost linear in n, compared to the PoA upper bounds of $o(n^\varepsilon)$ and $\mathcal{O}(\sqrt{\alpha}, n/\sqrt{\alpha})$ for the NCG and the bilateral NCG, respectively. As main take away from our paper, this implies that under social distancing the agents' selfish behavior has significantly more impact on the quality of the equilibria. This calls for strong coordination mechanisms governing the network formation to avoid detrimental stable states.

2 Complete Host Networks

We analyze the properties of the K-SDNCG, i.e., the SDNCG on complete host networks. First, we characterize optimal networks and give some examples for stable networks, dependent on the relation between n and α. After that, we show several bounds on the PoA and PoS.

2.1 Stable and Optimal Networks

Intuitively, for small α the distance utility dominates the social welfare. Hence, the path should be the optimum since it maximizes the total distances. For large α, the edge utility dominates, which leads to the clique being optimal since it maximizes the number of edges. Now we show that this intuition is indeed true. Moreover, the optimal construction is unique.

▶ **Theorem 1.** *For $\alpha < \frac{n}{3}$, the unique social optimum is the path. For $\alpha > \frac{n}{3}$, the unique optimum is the clique. For $\alpha = \frac{n}{3}$, the clique and the path are the only social optima.*

Proof. Šoltés and Ľubomír [49] showed that for a fixed number of nodes and edges, the network maximizing the summed distances is unique and contains a clique and a path with at least two edges between one endpoint of the path and the clique. We call this a *PathClique*. (Note that the clique can be empty, resulting in just a path) For a visualization, we refer to Figure 1. Note, that the social optimum has to be such a network, since for every other
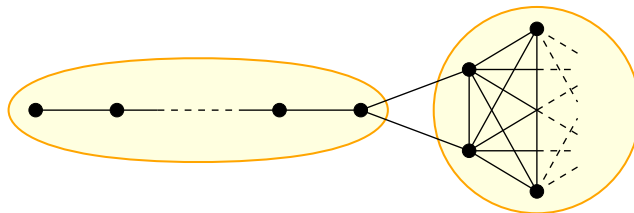
■ **Figure 1** This figure shows a PathClique. It consists of a path (left) and a clique (right), which are connected by at least two edges between one endpoint of the path and some nodes of the clique.

network, there is a PathClique with the same number of edges but larger summed distances and therefore a larger social welfare.

Let G be a PathClique with n nodes having a clique containing k nodes. Then the corresponding path contains $n - k$ nodes. First, we show that, unless G is a path or a clique, we get a socially better network by adding or removing edges.

Let G be neither a clique nor a path and let v be the endpoint of the path that is connected to the clique. Observe that removing an edge between v and the clique results in a PathClique. (This is still true if there are only two edges connecting v to the clique: Removing one of these edges makes the remaining neighbor of v in the clique the new endpoint of the path and reduces the size of the clique by 1.) Therefore, this is the socially best way of removing an edge from G. Similarly, the best way of adding an edge to G is adding it between v and the clique unless v is already fully connected to the clique in which case it is best to add an edge between the neighbor of v on the path and the clique. We now make a case distinction.

If v is fully connected to the clique, adding an edge decreases distances by $2(n - k - 1)$ and deleting an edge increases distances by $2(n - k)$. This means, that G can only be optimal if $\alpha \leq 2(n - k - 1)$ and $\alpha \geq 2(n - k)$, which is a contradiction.

If v is not fully connected to the clique, adding an edge decreases distances by $2(n - k)$ and deleting an edge increases distances by $2(n - k)$. Thus, $\alpha = 2(n - k)$ is necessary for G

to be socially optimal. But when $\alpha = 2(n-k)$, adding and deleting edges between v and the clique does not change the social welfare. Let G' be the network obtained by fully connecting v to the clique. Then $\text{SW}(G) = \text{SW}(G')$. Additionally, G' cannot be an optimum since it fulfills the conditions of the first case. Therefore, G cannot be socially optimal, too.

So, the social optimum network must be the path P_n or the clique K_n. We have

$$\text{SW}(P_n) = 2\alpha(n-1) + 2\sum_{i=1}^{n-1} i(n-i) = 2\alpha(n-1) + \frac{1}{3}(n-1)n(n+1),$$

$$\text{SW}(K_n) = 2\alpha\binom{n}{2} + 2\binom{n}{2} = n(n-1)(\alpha+1).$$

For $\alpha = \frac{n}{3}$, we see that

$$\text{SW}(P_n) = \frac{2}{3}n(n-1) + \frac{1}{3}(n-1)n(n+1) = n(n-1)\left(\frac{n}{3}+1\right) = \text{SW}(K_n),$$

for $\alpha < \frac{n}{3}$ we have $\text{SW}(P_n) > \text{SW}(K_n)$, and $\alpha > \frac{n}{3}$ yields $\text{SW}(K_n) > \text{SW}(P_n)$. ◀

Next, we have a look at the existence of pairwise stable networks. Similar to the social optimum, for small α, agents prefer large distances over many incident edges and therefore should remove as many edges as possible, leading to only trees being stable. Interestingly, the restrictions of pairwise stability lead to all trees being stable for small α, even if the distances are very small (like in a star). This is shown by the next theorem.

▶ **Theorem 2** (Stable Networks).
(1) *For $\alpha \leq 1$, every tree is pairwise stable. For $\alpha < 1$, any pairwise stable network is a tree.*
(2) *For $\alpha \geq 1$, the clique is pairwise stable.*
(3) *For $\alpha \leq \frac{n-1}{2}$, the path is pairwise stable.*
(4) *For $\alpha > \frac{n}{2}$, the clique is the only pairwise stable network.*

Proof of (1). Let G be a tree. Since removing an edge from G would lead to G being disconnected, we only have to consider adding an edge. This would shorten the distances for both endpoints by at least 1. Since $\alpha \leq 1$, this is not an improvement for the agents.

Let G be a network. If G contains a cycle, we can remove an edge without disconnecting the network. By doing this, the distances for both endpoints increase by at least 1. Since $\alpha < 1$, this is an improvement for both agents. ◀

Proof of (2). Let G be a clique. Since adding an edge is not possible, we only look at removing an edge. This would result in a distance increase of 1 for both endpoints. Since $\alpha \geq 1$, this is not an improvement for either of the two incident agents. ◀

Proof of (3). Let G be the path consisting of n nodes $v_1, \ldots, v_n$, in that order. Since G is a tree, no edge can be removed without disconnecting the network. Therefore, the only possible move is adding an edge. Let $1 \leq i < j \leq n$ be such that v_i and v_j are not adjacent (i.e., $j - i \geq 2$). If $j \leq \frac{n}{2}$, adding the edge $e := \{v_i, v_j\}$ shortens distances from v_i to at least $\frac{n}{2}$ nodes. The same holds for $i \geq \frac{n}{2}$ regarding node v_j. It is easy to see that the distance decrease is minimal when $j - i = 2$. Intuitively, edge e has to be as central as possible. According to this, choosing $i = \lfloor \frac{n}{2} \rfloor$ and $j = i + 2$ yields

$$d_G(v_i, V) - d_{G+e}(v_i, V) = n - j + 1 = n - i - 1 \geq \frac{n-2}{2} \text{ and}$$

$$d_G(v_j, V) - d_{G+e}(v_j, V) = i \geq \frac{n-1}{2}.$$

Thus, if $\alpha \leq \frac{n-1}{2}$, adding edge e is not an improving move and G is pairwise stable. ◀

Proof of (4). Let V be a set of n agents and $G = (V, E)$ be a stable network. Let $v \in V$ be a node having minimum total distances, i.e., for all $v' \in V$, we have $d_G(v, V) \leq d_G(v', V)$. Let $N_G[v]$ denote the closed neighborhood of v.

Now suppose, the network induced by $N_G[v]$ is not a clique. Then there are two neighbors x, y of v with $\{x, y\} \notin E$. We observe that for each node $z \in V$, the distances $d_G(v, z)$ and $d_G(x, z)$ can only differ by 1, since v and x are neighbors. By choice of v, there are at least as many nodes that are closer to v than nodes that are closer to x. Therefore, there are at most $\frac{n}{2}$ many nodes that are closer to x. Adding an edge between x and y can, for node y, only shorten distances to nodes which are closer to x than to v. Thus, this edge shortens the distances from y by at most $\frac{n}{2}$. The same holds for node x. Therefore, for $\alpha > \frac{n}{2}$, this edge would improve the utility of agents x and y and, thus, G would not be stable. This contradicts our assumption. Thus, $N_G[v]$ must induce a clique.

Now let x be a neighbor of v. Since x is connected to all neighbors of v, we have $d_G(x, V) \leq d_G(v, V)$, i.e., also x minimizes pairwise distances. Hence, $N_G[x]$ also induces a clique, leading to $N_G[v] = N_G[x]$. By induction, since G is connected, it must be a clique.   ◄

Theorem 2 implies that socially optimal networks are also stable. In fact, they are stable for a wide range of α-values. The clique is stable for $\alpha \geq 1$, meeting the bound below which only trees are stable. Similarly, the path is stable for $\alpha \leq \frac{n-1}{2}$, almost meeting the lower bound for only the clique being stable. Additionally, we observe that we only need two networks (path and clique) to provide pairwise stable networks for all possible values of α.

For further constructions, we need the following definition. Let G be a network. We call G' a *clique network* of G, if it can be obtained by replacing each node of G by a clique of size at least 2 and for each edge of G connect the two corresponding cliques fully bipartite. By using only constant-size cliques, some properties of G (density, length of shortest paths) are preserved while the network is more stable against edge removal.

▶ **Theorem 3.** *Let G be a clique network. For $\alpha \geq 1$, G is stable against edge removal.*

Proof. Removing any edge from G only effects the distances between its endpoints. These distances increase by 1.   ◄

Finally, we show that stable states may not be found by simply letting agents iteratively play improving moves, i.e., via a sequential process of improving strategy changes. Figure 2 provides an example of a cyclic sequence of improving moves. This also implies that both the K-SDNCG and the SDNCG do not belong to the class of potential games [44], i.e., the existence of equilibria cannot be proven via potential function arguments.

▶ **Theorem 4.** *The Social Distancing Network Creation Game is not a potential game.*

Proof. This is shown by the existence of improving cycles. See Figure 2 for an example.   ◄
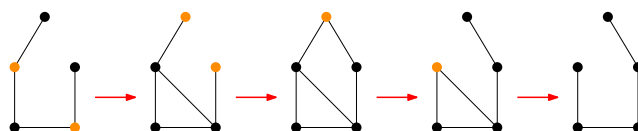
■ **Figure 2** This figure shows a cyclic sequence of improving moves performed by $n = 5$ agents for $\alpha = 2.5$. In each step, the nodes responsible for the next change are highlighted in orange. Note that the last step is isomorphic to the first step.

2.2   Price of Anarchy and Price of Stability

In this section, we give a series of bounds for the Price of Anarchy and the Price of Stability.

▶ **Theorem 5** (Price of Anarchy).
(1) *The Price of Anarchy is in $\mathcal{O}(n)$.*
(2) *For $\alpha \leq 1$, the Price of Anarchy is in $\Theta(n)$.*
(3) *For $1 < \alpha \leq \sqrt{n}$, the Price of Anarchy is in $\Theta(n)$.*
(4) *For $\sqrt{n} \leq \alpha \leq \frac{n}{6} - 3$, the Price of Anarchy is in $\Omega\left(\frac{n}{\log n}\right)$.*
(5) *For $\frac{n}{6} - 3 < \alpha \leq \left\lfloor \frac{n}{2} \right\rfloor - 2$, the Price of Anarchy is in $\Omega\left(\sqrt{n}\right)$.*
(6) *For $\alpha \geq \frac{n}{2}$, the Price of Anarchy is 1.*

Proof of (1). Let $\alpha \in \mathbb{R}_{>0}$. Every connected network has at least $n - 1 \in \Omega(n)$ and at most $\binom{n}{2} \in \mathcal{O}(n^2)$ edges. Furthermore, the summed distances are at least $n(n-1) \in \Omega(n^2)$, since every node has at least distance 1 to every other node, and at most n^3, since every node has at most distance n to every other node. With this, we get the trivial bound of

$$\mathrm{PoA}_n \in \mathcal{O}\left(\frac{\alpha n^2 + n^3}{\alpha n + n^2}\right) = \mathcal{O}(n). \qquad \blacktriangleleft$$

Proof of (2). For $\alpha \leq 1$, the social optimum is the path as shown in Theorem 1. It has social welfare of

$$\alpha(n-1) + \frac{1}{3}(n-1)n(n+1).$$

Because of Theorem 2 and the star being a tree, it is pairwise stable for $\alpha \leq 1$. It has social welfare of

$$\alpha(n-1) + 2(n-1) + 2(n-1)(n-2) = \alpha(n-1) + 2(n-1)^2.$$

Together with (1), this yields

$$\mathrm{PoA} \geq \frac{\alpha(n-1) + \frac{1}{3}(n-1)n(n+1)}{\alpha(n-1) + 2(n-1)^2} = \frac{\alpha + \frac{1}{3}n(n+1)}{\alpha + 2(n-1)} \in \Theta(n). \qquad \blacktriangleleft$$

Proof of (3). We construct a star-like network with cliques as leaves in the following way. Let $c := \lceil \alpha \rceil + 2$. Additionally, let $K_1, \ldots, K_d$ be $d := \left\lfloor \frac{n-2}{c} \right\rfloor$ cliques containing $c - 2$ nodes and $v_1, v_1', v_2, v_2', \ldots, v_d, v_d'$ be $2d$ nodes. Let furthermore M be a clique of size $n - cd$. We now define our network G as

$$V_G := \bigcup_{i=1}^{d} V_{K_i} \cup \bigcup_{i=1}^{d} \{v_i, v_i'\} \cup V_M$$

$$E_G := \bigcup_{i=1}^{d} E_{K_i} \cup E_M \cup \bigcup_{i=1}^{d} \{\{v_i, v_i'\}\}$$

$$\cup \bigcup_{i=1}^{d} \bigcup_{v \in K_i} \{\{v, v_i\}, \{v, v_i'\}\} \cup \bigcup_{i=1}^{d} \bigcup_{v \in M} \{\{v, v_i\}, \{v, v_i'\}\}.$$

We essentially connect the outer cliques $K_1, \ldots, K_d$ to the center clique M via d 2-cliques and each connection is fully bipartite (see Figure 3). Since $n = |V_G|$, G is a network of the desired size.
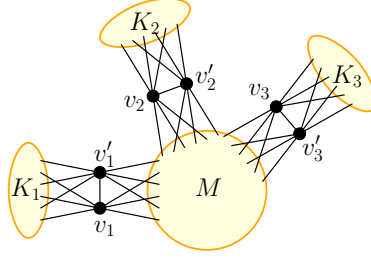
Figure 3 The figure shows a star-like clique network, where the center is formed by a clique M and each ray consists of two nodes v_i, v_i' and a clique K_i.

We now show that G is pairwise stable. We see that G is a clique network. Because of Theorem 3 and $\alpha > 1$, G is stable against edge removal. On the other hand, adding an edge shortens distances to at least $|K_i| = c - 2 \geq \alpha$ nodes which means a distance decrease of at least α for the two incident nodes. This also does not increase their utility. Therefore, G is pairwise stable.

For the center clique M, we see that $|V_M| = n - cd = n - c \lfloor \frac{n-2}{c} \rfloor$ and therefore $2 \leq |V_M| < n - (n - 2 - c) = c + 2$. With this and $1 < \alpha \leq \sqrt{n}$, we obtain

$$|E_G| = d\binom{c-2}{2} + \binom{|V_M|}{2} + 2d + d(c-2)2 + d|V_M|2$$
$$= \left\lfloor \frac{n-2}{\lceil \alpha \rceil + 2} \right\rfloor \left(\frac{\lceil \alpha \rceil(\lceil \alpha \rceil - 1)}{2} + 2 + 2\lceil \alpha \rceil + 2|V_M| \right) + \frac{|V_M|(|V_M| - 1)}{2}$$
$$\in \Theta(\alpha n)$$

and $d_G(V_G, V_G) \in \Theta(n^2)$.

For $\alpha < \sqrt{n}$, the socially optimal network is the path. With the previous calculations, we can now bound the Price of Anarchy as

$$\text{PoA} \geq \frac{2\alpha(n-1) + \Theta(n^3)}{2\alpha\Theta(\alpha n) + \Theta(n^2)} = \frac{\Theta(n^3)}{\Theta(n^2)} \in \Omega(n).$$

From (1), we have $\text{PoA} \in \mathcal{O}(n)$ and therefore $\text{PoA} \in \Theta(n)$. ◀

Proof of (4). Let $d = \lfloor \log n \rfloor - 1$. Then, the d-dimensional hypercube is represented by G_H with $V_{G_H} = \{0,1\}^d$ and $E_{G_H} = \{\{v,x\} \mid v, x \in V \wedge d_H(v,x) = 1\}$ where $d_H(v,x)$ denotes the Hamming Distance between v and x. Let G be a clique network for G_H with $|V_G| = n$ such that the sizes of the cliques replacing the nodes of G_H differ by at most 1. Observe, that each clique is of size 2 or 3 if $2 \cdot 2^d \leq n < 3 \cdot 2^d$ and of size 3 or 4 if $3 \cdot 2^d \leq n < 4 \cdot 2^d$. By Theorem 3 and since $\alpha \geq 1$, we know that G is stable against edge removal. We now show that adding an edge shortens the total distances for the incident nodes by at least $\frac{n}{6} - 3$.

Let $v, x \in V_G$ such that $e := \{v, x\} \notin E_G$ and let $v', x' \in V_{G_H}$ be the nodes corresponding to the cliques that contain v and x, respectively. Therefore, $e' := \{v', x'\} \notin E_{G_H}$, which implies $d_H(v', x') \geq 2$. By symmetry of the hypercube, we can assume w.l.o.g. that

$$v' = \underbrace{00\ldots0}_{d_H(v,x)} \underbrace{0\ldots00}_{d - d_H(v,x)} \qquad \text{and} \qquad x' = \underbrace{11\ldots1}_{d_H(v,x)} \underbrace{0\ldots00}_{d - d_H(v,x)} .$$

Adding e' to G_H decreases the distances from v' to another node $y' \in V_{G_H}$ if and only if $d_H(v', y') \geq d_H(x, y) + 2$. The difference in distance can only come from the first $d_H(v', x')$

bits of the label since the remaining bits are equal for v' and x'. Let ℓ be the number of the first $d_H(v',x')$ bits of y' equal to 1. Then, $d_H(v',x') - \ell$ is the number of the first $d_H(v',x')$ bits of y' equal to 0. We obtain $d_H(v',y') - d_H(x',y') = \ell - (d_H(v',x') - \ell) = 2\ell - d_H(v',x')$. Thus, adding e' to G_H shortens the distance from v' to y' by $2\ell - d_H(v',x') - 1$.

The number of nodes where exactly ℓ of the first $d_H(v',x')$ bits are equal to 1 is $\binom{d_H(v',x')}{\ell} \cdot 2^{d - d_H(v',x')}$. Therefore, we get a distance decrease for v' of

$$
\begin{aligned}
&d_{G_H}(v', V_{G_H}) - d_{G_H+e'}(v', V_{G_H}) \\
&= \sum_{\ell = \left\lceil \frac{d_H(v',x')}{2} \right\rceil + 1}^{d_H(v',x')} \binom{d_H(v',x')}{l} \cdot 2^{d - d_H(v',x')} \cdot (2\ell - d_H(v',x') - 1) \\
&= 2^{d - d_H(v',x')} \sum_{\ell = 0}^{\left\lfloor \frac{d_H(v',x')}{2} \right\rfloor - 1} \binom{d_H(v',x')}{\ell} \cdot (d_H(v',x') - 2\ell - 1) \\
&\geq 2^{d - d_H(v',x')} \sum_{\ell = 0}^{\left\lfloor \frac{d_H(v',x')}{2} \right\rfloor - 1} \binom{d_H(v',x')}{\ell} \\
&\geq 2^{d - d_H(v',x')} \frac{1}{2} \left(\sum_{\ell = 0}^{d_H(v',x')} \binom{d_H(v',x')}{\ell} - \binom{d_H(v',x')}{\left\lfloor \frac{d_H(v',x')}{2} \right\rfloor} \right) \\
&= 2^{d - d_H(v',x')} \frac{1}{2} \left(2^{d_H(v',x')} - \binom{d_H(v',x')}{\left\lfloor \frac{d_H(v',x')}{2} \right\rfloor} \right) \\
&\geq 2^{d - d_H(v',x')} \frac{1}{2} 2^{d_H(v',x') - 1} \\
&= \frac{2^d}{4}.
\end{aligned}
$$

Observe that the distances from v to all nodes in other cliques in G are exactly the same as the distances from v' to all other nodes in G_H. The same holds for $G + e$ and $G_H + e'$, with the exception of the distances from v to the (at most 3) nodes in the same clique as x. We distinguish two cases:

If $2 \cdot 2^d \leq n < 3 \cdot 2^d$, each clique consists of 2 or 3 nodes. Therefore, we have a distance decrease of at least

$$
d_G(v, V_G) - d_{G+e}(v, V_G) \geq 2(d_{G_H}(v', V_{G_H}) - d_{G_H+e'}(v', V_{G_H})) - 3 \geq \frac{2^d}{2} - 3 \geq \frac{n}{6} - 3.
$$

If $3 \cdot 2^d \leq n < 4 \cdot 2^d$, each clique consists of 3 or 4 nodes. This means, we have a distance decrease of at least

$$
d_G(v, V_G) - d_{G+e}(v, V_G) \geq 3(d_{G_H}(v', V_{G_H}) - d_{G_H+e'}(v', V_{G_H})) - 3 \geq 3\frac{2^d}{4} - 3 \geq \frac{n}{6} - 3.
$$

Thus, edge additions are not beneficial for the incident agents if $\alpha \leq \frac{n}{6} - 3$ and we conclude that the constructed network is stable for $1 \leq \alpha \leq \frac{n}{6} - 3$. We also see that the number of edges m is in $\Theta(n \log n)$ and the distance $d(V_G, V_G)$ is in $\Theta(n^2 \log n)$. Since the social optimum for $1 \leq \alpha \leq \frac{n}{6} - 3$ is the path P_n, we get for the Price of Anarchy:

$$
\text{PoA}_n \geq \frac{SW(P_n)}{SW(G)} = \frac{\alpha(n-1) + \Theta(n^3)}{\alpha\Theta(n \log n) + \Theta(n^2 \log n)} \in \Omega\left(\frac{n}{\log n} \right). \qquad \blacktriangleleft
$$

Proof of (5). We construct a path of cliques in the following way. Let $2 \le d \le \frac{n-6}{2}$ be some even number and $c = \left\lfloor \frac{n-6}{d} \right\rfloor$. Furthermore, let $K_1, \ldots, K_d$ be d cliques consisting of c or $c+1$ nodes, such that $\sum_{i=1}^{d} |V_{K_i}| = n - 6$ and $\sum_{i=1}^{\frac{d}{2}} |V_{K_i}| = \left\lceil \frac{n-6}{2} \right\rceil$ and $\sum_{i=\frac{d}{2}+1}^{d} |V_{K_i}| = \left\lfloor \frac{n-6}{2} \right\rfloor$, and $v_1, v_1', v_2, v_2', v_3, v_3'$ be 6 more nodes. We now define the network G as

$$V_G := \bigcup_{i=1}^{d} V_{K_i} \cup \{v_1, v_1', v_2, v_2', v_3, v_3'\},$$

$$E_G := \bigcup_{i=1}^{d} E_{K_i} \cup \{\{v_1, v_1'\}, \{v_2, v_2'\}, \{v_3, v_3'\}\} \cup \{\{v, x\} \mid v \in \{v_2, v_2'\} \wedge x \in \{v_1, v_1', v_3, v_3'\}\}$$

$$\cup \bigcup_{i=1}^{\frac{d}{2}-1} \{\{v, x\} \mid v \in K_i \wedge x \in K_{i+1}\} \cup \bigcup_{v \in K_{\frac{d}{2}}} \{\{v, v_1\}, \{v, v_1'\}\}$$

$$\cup \bigcup_{i=\frac{d}{2}+1}^{d-1} \{\{v, x\} \mid v \in K_i \wedge x \in K_{i+1}\} \cup \bigcup_{v \in K_{\frac{d}{2}+1}} \{\{v, v_3\}, \{v, v_3'\}\}.$$

Figure 4 shows a sketch of G.
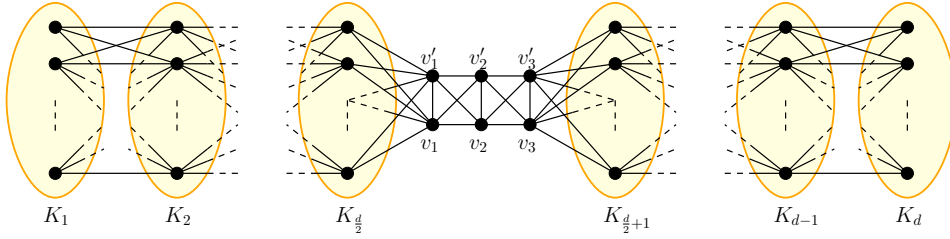
■ **Figure 4** The figure shows clique network for a path consisting of d cliques $K_1, \ldots, K_d$ highlighted in yellow with 6 additional nodes in the middle. Note, that edges inside the cliques are not shown in this figure.

We observe that G is stable against edge removal because of Theorem 3, since $\alpha > \frac{n}{6} - 3 \ge 1$ and G being a clique network for the path. We now show that adding an edge is also not an improving move.

We quickly see that, for a node v, adding an edge into the 2-neighborhood always shortens distances the least. We therefore only have to consider these edges. We observe that adding an edge between v_1 and v_3 (or because of symmetry, v_1' or v_3') decreases distances from v_1 to v_3 and all nodes in $K_{\frac{d}{2}+1}, \ldots, K_d$ and decreases distances from v_3 to v_1 and all nodes in $K_1, \ldots, K_{\frac{d}{2}}$ by exactly 1. This means, we get

$$d_G(v_1, V) - d_{G+\{v_1, v_3\}}(v_1, V) = 1 + \sum_{i=\frac{d}{2}+1}^{d} |V_{K_i}| = \left\lfloor \frac{n}{2} \right\rfloor - 2 \quad \text{and}$$

$$d_G(v_3, V) - d_{G+\{v_1, v_3\}}(v_3, V) = 1 + \sum_{i=1}^{\frac{d}{2}} |V_{K_i}| = \left\lceil \frac{n}{2} \right\rceil - 2.$$

Every other edge we could add decreases distances to all the cliques of one side of the path, resulting in larger distance decreases. This means that adding an edge is not an improving move for $\alpha \le \left\lfloor \frac{n}{2} \right\rfloor - 2$. Therefore, G is pairwise stable for the desired values of α.

We now evaluate the number of edges. We have $|E_{K_i}| \in \Theta(c^2)$. The number of edges between two neighboring cliques is also in $\Theta(c^2)$. This means that the total number of edges is $|E_G| \in \Theta(dc^2)$. We also see that the diameter of G is d and therefore $d_G(V, V) \in \mathcal{O}(dn^2)$. If we choose $d = 2 \left\lfloor \frac{\sqrt{n}}{2} \right\rfloor$, we have $d \in \Theta(\sqrt{n})$ and $c \in \Theta(\sqrt{n})$. Since $\alpha \in \Theta(n)$, we get for the Price of Anarchy

$$\mathrm{PoA}_n \geq \frac{\alpha(n-1) + \Theta(n^3)}{\alpha\Theta(dc^2) + \mathcal{O}(dn^2)} \in \Omega\left(\frac{n^3}{n^{\frac{5}{2}} + n^{\frac{5}{2}}}\right) = \Omega\left(\sqrt{n}\right). \qquad \blacktriangleleft$$

Proof of (6). This follows directly from the clique being socially optimal (see Theorem 1) and the only pairwise stable network (see Theorem 2). $\blacktriangleleft$

We have established that the Price of Anarchy is relatively high for $\alpha \leq \frac{n}{2}$. It even meets the trivial upper bound of $\mathcal{O}(n)$ for a large range of α. In contrast to the high PoA values, we observe that the Price of Stability is independent of α and best possible.

▶ **Theorem 6.** *The Price of Stability is 1.*

Proof. This follows directly from the path being stable and socially optimal for $\alpha \leq \frac{n}{3}$ and the clique being stable and socially optimal for $\alpha \geq \frac{n}{3}$ (see Theorem 1 and Theorem 2). $\blacktriangleleft$

From an efficiency point-of-view, the huge gap between the PoA and the PoS suggests that having an outside force assigning an initial strategy to all players is beneficial. That way, stability and optimal social welfare can be guaranteed. Without such coordination, the players could end up in socially bad equilibria or in a cyclic sequence of improving moves.

3   General Host Networks

We now analyze the SDNCG on arbitrary connected but not necessarily complete host networks. First, we analyze socially optimal networks and then we investigate the existence of pairwise stable networks. We prove our main result that establishes equilibrium existence on any connected host network for a wide parameter range of α. Finally, we derive bounds on the Price of Anarchy and the Price of Stability. Additionally, we show that computing the social optimum and the Maximum Routing-Cost Spanning Tree is NP-hard while computing a Swap-Maximal Routing-Cost Spanning Tree can be done in polynomial time.

3.1   Stable and Optimal Networks

While for the K-SDNCG, we only have two possible social optima (dependent on α), this gets more complicated for general host networks. Of course, if they exist on general host networks, then the optima for the K-SDNCG are still the most efficient networks. Intuitively, if the host network does not contain a Hamilton path, then the social optimum should be a tree if α is small enough. Since all trees have the same number of edges, the social welfare of a tree is only influenced by the total distances. Remember that the spanning tree maximizing the total distances is by definition the Maximum Routing-Cost Spanning Tree (MRCST). We now show, that this intuition is indeed correct.

▶ **Theorem 7** (Social Optimum). *Let H be a connected host network containing n nodes.*
(1) *If H contains a Hamilton path, then this path is the social optimum for $\alpha \leq \frac{n}{3}$. The Hamilton path is the unique social optimum if $\alpha < \frac{n}{3}$.*
(2) *For $\alpha \leq 1$, the MRCST of H is socially optimal.*

(3) *For $\alpha > \frac{1}{24}(n-2)n(n+2)$, H itself is the unique social optimum.*

Proof of (1). This follows directly from Theorem 1. ◀

Proof of (2). Let T_{opt} be the MRCST of H and G some state of H, that is, a spanning subnetwork of H. Furthermore, let T be a spanning tree of G. Then $\mathrm{SW}(G) \leq \mathrm{SW}(T)$, since we can construct G by adding edges to T and for every edge added, the social welfare goes up by $\alpha \leq 1$ and down by at least 1, because of distances decreasing. Since T_{opt} maximizes the total distances, we have $\mathrm{SW}(T) \leq \mathrm{SW}(T_{opt})$ and therefore $\mathrm{SW}(G) \leq \mathrm{SW}(T_{opt})$. ◀

Proof of (3). We show that any edge added to any network shortens the total distances by at most $\frac{1}{24}(n-1)n(n+1)$. It is easy to see that adding an edge between the two endpoints of a Hamilton path maximizes the distance decrease. This means, if α is larger than that, it is always socially better to add more edges to the network, resulting in H itself to be optimal.

We already know the social welfare of a Hamilton path P_n from Section 2. After adding an edge between the two endpoints of P_n, we get a cycle C_n. In this cycle, we see that for each node there are exactly two nodes for every possible distance $1 \leq d < \frac{n}{2}$ and an additional node at distance $\frac{n}{2}$, if n is even. This yields

$$\mathrm{SW}(P_n) = 2\alpha(n-1) + \frac{1}{3}(n-1)n(n+1),$$

$$\mathrm{SW}(C_n) = 2\alpha n + n\sum_{i=1}^{\frac{n-1}{2}} 2i = 2\alpha n + \frac{1}{4}(n-1)n(n+1) \qquad \text{for } n \text{ odd,}$$

$$\mathrm{SW}(C_n) = 2\alpha n + n\left(\sum_{i=1}^{\frac{n}{2}-1} 2i + \frac{n}{2}\right) = 2\alpha n + \frac{1}{4}(n-2)n^2 + \frac{n^2}{2} \qquad \text{for } n \text{ even.}$$

We then have $\mathrm{SW}(C_n) > \mathrm{SW}(P_n)$ if and only if

$$\alpha > \frac{1}{24}(n-1)n(n+1) \qquad\qquad \text{for } n \text{ odd,}$$

$$\alpha > \frac{1}{24}(n-2)n(n+2) > \frac{1}{24}(n-1)n(n+1) \qquad\qquad \text{for } n \text{ even.}$$

This shows the claim. ◀

Contrasting statement (3) from Theorem 7, we observe that for $\alpha < \frac{1}{24}(n-2)n(n+2)$, the host network is not necessarily the social optimum. Consider the host network $H := C_n$ for even n, i.e., an even cycle with n nodes. In the proof of (3), we see that $\mathrm{SW}(P_n) > \mathrm{SW}(C_n)$, implying that C_n cannot be the social optimum. In fact, in this example, P_n is the optimum since there are only two possible states (up to isomorphism): P_n and C_n itself. This is in stark contrast to the K-SDNCG, where the host network is optimal for $\alpha \geq \frac{n}{3}$.

Since finding a MRCST is NP-hard [16], finding the social optimum for a given host network must also be NP-hard.

▶ **Theorem 8** (Computational Hardness). *Computing the social optimum for a connected host network H is NP-hard.*

Proof. This follows directly from the MRCST being hard to compute [16] and the unique social optimum for $\alpha < 1$ (see Theorem 7). ◀

Next, we discuss stable networks. In contrast to the K-SDCNG, it is not obvious that pairwise stable networks are guaranteed to exist for any connected host network. However, we can directly transfer the result that spanning trees are stable for small α. For large α, similar to the clique being the unique stable network for $\alpha > \frac{n}{2}$ for complete host networks, as shown in Theorem 2, we show that the whole host network is pairwise stable. However, in contrast to the K-SDNCG, this is true only for much larger values of α.

▶ **Theorem 9** (Stable Networks). *Let H be a connected host network containing n nodes.*
(1) *For $\alpha \leq 1$, every spanning tree of H is pairwise stable. For $\alpha < 1$, spanning trees are the only pairwise stable networks.*
(2) *For $\alpha > \frac{1}{4}(n-1)^2$, H is the only pairwise stable network.*

Proof of (1). The proof is exactly the same as for (1) of Theorem 2. ◀

Proof of (2). Consider a network G on a host network $H = (V, E_H)$. The largest distance decrease a node $v \in V$ can suffer when forming an edge $e \in E_H$ is when G is a path and v one of its endpoints connecting to the other endpoint $x \in V$. This move decreases the distances by

$$\Delta d := d_G(v, V) - d_{G+e}(v, V) = d_G(x, V) - d_{G+e}(x, V)$$

$$= \sum_{i=1}^{\frac{n-1}{2}} (n - 2i) = \frac{1}{4}(n-1)^2 \qquad \text{for } n \text{ odd,}$$

$$\Delta d = \sum_{i=1}^{\frac{n}{2}-1} (n - 2i) = \frac{1}{4}(n-2)n < \frac{1}{4}(n-1)^2 \qquad \text{for } n \text{ even.}$$

Thus, since $\alpha > \frac{1}{4}(n-1)^2$, forming edges is always beneficial for the incident nodes. Similarly, edge removal always decreases the utility of the incident nodes. Therefore, the host network H is the only pairwise stable network. ◀

Contrasting statement (2) of Theorem 9, using $H := C_n$ for odd n and $\alpha < \frac{1}{4}(n-1)^2$ shows that the host network is not necessarily pairwise stable. This example also shows that the optimum is not necessarily stable: For $\alpha \geq \frac{1}{4}(n-1)^2$ and $H := C_n$ as the host network, C_n is the only pairwise stable network but it is not the optimum for $\alpha < \frac{1}{24}(n-2)n(n+2)$. This is another significant difference to the K-SDNCG.

Now that we characterized stable networks for extreme α-values, the question remains whether stable states also exist for in-between values. For the K-SDNCG, the path is stable up to $\alpha < \frac{n-1}{2}$. This is, of course, still true for non-complete host networks if they contain a Hamilton path. Since a Hamilton path (if it exists) is the MRCST, it is natural to suspect that the MRCST properties at least partially ensure stability for some $\alpha \geq 1$. However, even if true, the MRCST is still NP-hard to compute. Hence, in quest of an efficiently computable stable network, we introduce a less strict variant of MRCSTs which is only locally optimal: Swap-Maximal Routing-Cost Spanning Trees. Remember, a SMRCST is a spanning tree whose summed distances cannot be increased by removing one edge and adding another edge.

As our main result, we now show that SMRCSTs (and therefore MRCSTs, too) are indeed stable beyond $\alpha \leq 1$. Note, that for the inverse model of the NCG on an arbitrary host network [19], so far no equilibrium existence statement is known.

▶ **Theorem 10.** *Let H be a connected host network containing n nodes. Then for $\alpha \leq \frac{n}{3}$, any Swap-Maximal Routing-Cost Spanning Tree is pairwise stable.*
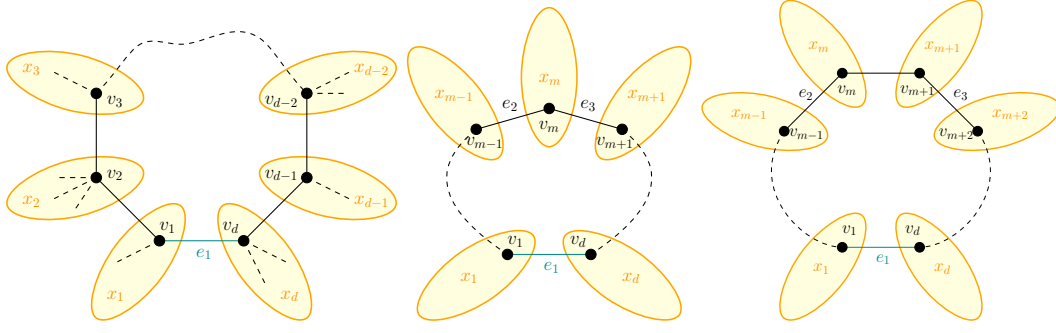
■ **Figure 5** This figure shows the cycle formed by adding e_1 to the SMRCST. The cycle is of length d and contains the nodes $v_1, \ldots, v_d$. Every other node is contained in one of the subtrees rooted in one of the nodes on the cycle. These subtrees are represented in yellow. The number of nodes contained in the subtree rooted in v_i is x_i. Middle and right: the cycle for d being odd or even, respectively, and the two special edges e_2 and e_3.

Proof. Let G with $V_G = V_H$ and $E_G \subseteq E_H$ be a SMRCST. Since G is a tree, we only have to consider edge additions. We show that adding any edge decreases the summed distances for at least one of the edges endpoints by at least $\frac{n}{3}$. This is sufficient to show the claim.

Let $e_1 \in E_H \setminus E_G$ be an edge not part of the SMRCST. Adding e_1 would form a cycle of length $d \in \mathbb{N}$ consisting of nodes $v_1, \ldots, v_d \in V$ with v_1 and v_d being the nodes incident to e_1. Let E_C be the set of all edges on this cycle. Removing all edges in E_C from G would create d trees rooted in $v_1, \ldots, v_d$ respectively. Let furthermore $x_1, \ldots, x_d$ be the number of nodes in each of the d trees. See Figure 5 for an illustration.

Since G is a tree, there is exactly one path between each pair of nodes (which is also the shortest). For each edge $e \in E_G$, we define $d_G(e)$ as the number of paths between two nodes in G which include e. We then can express the total distances as

$$d_G(V, V) = 2 \sum_{e \in E_G} d_G(e).$$

Note, that each path between two nodes contributes twice to the total distances (one for each node), which leads to the factor of 2.

Let $x := (x_1, \ldots, x_d)$. We now define for each edge $e \in E_C$ on the cycle

$$c_e(x) := \sum_{e' \in E_C} d_{G+e_1-e}(e') = \sum_{i=1}^{d-1} \sum_{j=i+1}^{d} x_i x_j d_{G+e_1-e}(v_i, v_j).$$

This is the contribution of all the edges on the cycle to the total distances if we add e_1 to it and instead remove e from it. Note that c_{e_1} is the value for the original network since $G + e_1 - e_1 = G$. We see that c_e does not depent on the structure of the subtrees rooted in the v_i but only on the number of nodes in each subtree. Since the number of paths going over an edge that is not on the cycle does not change when we add e_1 and remove e, we have

$$d_G(V, V) - d_{G+e-e_1}(V, V)$$
$$= 2 \sum_{e' \in E} (d_G(e') - d_{G+e-e_1}(e'))$$
$$= 2 \sum_{e' \in E_C} (d_G(e') - d_{G+e-e_1}(e')) + 2 \sum_{e' \in E_G \setminus E_C} (d_G(e') - d_{G+e-e_1}(e'))$$

$$= 2c_{e_1}(x) - 2c_e(x).$$

We know that G is a SMRCST of H. This means that $d_G(V, V) \geq d_{G'}(V, V)$ for any other spanning tree G' which can be obtained from G by a swap of one edge. We therefore also have

$$\forall e \in E_C \colon c_{e_1}(x) \geq c_e(x). \tag{1}$$

Now, we use the previous observations to formulate and solve a minimization problem which yields the desired bound. We start with some definitions.

We call $x = (x_1, \ldots, x_d) \in \mathbb{N}^d$ with $x_i \geq 1$ and $\sum_{i=1}^d x_i = n$ a *node distribution*. For each edge $e \in E_C$, we call $c_e(x)$ (defined above) the *cost* of e. And lastly, we define the *distance decrease* Δd as

$$\Delta d(x) := \max \left\{ \sum_{i=1}^{\lfloor \frac{d-1}{2} \rfloor} (d - 2i)x_i, \ \sum_{i=1}^{\lfloor \frac{d-1}{2} \rfloor} (d - 2i)x_{d-i+1} \right\}. \tag{2}$$

The goal then is: Find a node distribution x that fulfills 1 and minimizes $\Delta d(x)$. Observe that this indeed yields a lower bound for the distance decrease when adding e to G. If we show that this is at least $\frac{n}{3}$, we proved the statement.

Let $x = (x_1, \ldots, x_d) \in \mathbb{N}^d$ be a node distribution minimizing $\Delta d(x)$. We first show the claim for $d = 3$ and $d = 4$.

For $d = 3$, we have $x_2 \leq x_1$ and $x_2 \leq x_3$ from 1 and $\Delta d(x) = \max\{x_1, x_3\}$. Since $x_1 + x_2 + x_3 = n$, this yields $\Delta d(x) \geq \frac{n}{3}$.

For $d = 4$, we have $c_{e_1}(x) - c_{\{x_1,x_2\}}(x) = 2x_1x_4 - 2x_1x_2$, and therefore $c_{e_1} \geq c_{\{x_1,x_2\}}$ if and only if $x_4 \geq x_2$. Similarly, we get $x_1 \geq x_3$ from $\{x_3, x_4\}$. Together with $x_1 + x_2 + x_3 + x_4 = n$, we see that $\max\{x_1, x_4\} \geq \frac{n}{4}$. We conclude that $\Delta d(x) = 2\max\{x_1, x_4\} \geq \frac{n}{2} > \frac{n}{3}$.

For $d > 4$, we make a case distinction between d being odd and d being even and simplify the problem by doing several relaxation steps. For $d \leq 4$, it is easy to show that $\Delta d(x) \geq \frac{n}{3}$. For further steps, we allow $x \in \mathbb{R}_{\geq 1}^d$. Note, that this only allows for smaller minima and therefore still yields a lower bound for the original problem.

The high level idea of the following steps is that we can redistribute weights of the node distribution x without changing $\Delta(x)$ or violating 1 and thereby reducing the number of variables contained in x by setting most x_i to 1. We now make a case distinction.

Case d odd: Let $m = \frac{d+1}{2}$ and $e_2 := \{v_{m-1}, v_m\}$ and $e_3 := \{v_m, v_{m+1}\}$. Thus, v_m is the node equidistant from v_1 and v_d in C and e_2 and e_3 are the edges on C incident to v_m. (see Figure 5 (middle)) We will only consider the two constraints

$$c_{e_1}(x) \geq c_{e_2}(x) \qquad \text{and} \qquad c_{e_1}(x) \geq c_{e_3}(x), \tag{3}$$

where

$$c_{e_1}(x) = \sum_{i=1}^{d-1} \sum_{j=i+1}^{d} (j - i)x_i x_j,$$

$$c_{e_2}(x) = \sum_{i=1}^{m-2} \sum_{j=i+1}^{m-1} (j - i)x_i x_j + \sum_{i=m}^{d-1} \sum_{j=i+1}^{d} (j - i)x_i x_j + \sum_{i=1}^{m-1} \sum_{j=m}^{d} (i + d - j)x_i x_j,$$

$$c_{e_3}(x) = \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} (j - i)x_i x_j + \sum_{i=m+1}^{d-1} \sum_{j=i+1}^{d} (j - i)x_i x_j + \sum_{i=1}^{m} \sum_{j=m+1}^{d} (i + d - j)x_i x_j.$$

This still yields a lower bound for the original problem since the constraints from 3 are a subset of the constraints from 1.

We observe that the claim is trivially true for $\frac{n}{3} \leq \frac{d+3}{2}$ since $\Delta d(x) \geq \frac{d+3}{2}$ (tight for $n = 5$, $d = 5$, $x = (1,1,1,1,1)$). This means, we can assume

$$n > \frac{3(d+3)}{2}, \qquad \text{or rather} \qquad n > 3(m+1). \tag{4}$$

In the following, we perform a series of relaxations. For that, we define the node distributions $x^{(1)}, x^{(2)}, x^{(3)}$ such that for all $1 \leq i \leq d$

$$x_i^{(1)} := \begin{cases} x_1 + \sum_{p=2}^{m-2} \frac{m-1-p}{m-2}(x_p - 1) & \text{if } i = 1, \\ x_{m-1} + \sum_{p=2}^{m-2} \frac{p-1}{m-2}(x_p - 1) & \text{if } i = m-1, \\ x_m & \text{if } i = m, \\ x_{m+1} + \sum_{p=m+2}^{d-1} \frac{d-p}{m-2}(x_p - 1) & \text{if } i = m+1, \\ x_d + \sum_{p=m+2}^{d-1} \frac{p-m-1}{m-2}(x_p - 1) & \text{if } i = d, \\ 1 & \text{else}, \end{cases}$$

(Without loss of generality, we assume $x_1^{(1)} \geq x_d^{(1)}$

and $(2m-3)x_1^{(1)} + x_{m-1}^{(1)} = (2m-3)x_d^{(1)} + x_{m+1}^{(1)}$.)

$$x_i^{(2)} := \begin{cases} x_1^{(1)} & \text{if } i = 1 \text{ or } i = d, \\ x_{m-1}^{(1)} & \text{if } i = m-1 \text{ or } i = m+1, \\ x_m^{(1)} + (2m-4)\left(x_1^{(1)} - x_d^{(1)}\right) & \text{if } i = m, \\ 1 & \text{else}, \end{cases}$$

$$x_i^{(3)} := \begin{cases} x_m^{(2)} + 2\frac{2m-4}{2m-3}\left(x_{m-1}^{(2)} - 1\right) & \text{if } i = m, \\ x_1^{(2)} + \frac{1}{2m-3}\left(x_{m-1}^{(2)} - 1\right) & \text{if } i = 1 \text{ or } i = d, \cdot \\ 1 & \text{else}. \end{cases}$$

It is easy to see that $\Delta d(x) = \Delta d\left(x^{(1)}\right) = \Delta d\left(x^{(2)}\right) = \Delta d\left(x^{(3)}\right)$. We show that $c_{e_1}\left(x^{(i)}\right) \geq c_{e_2}\left(x^{(i)}\right)$ and $c_{e_1}\left(x^{(i)}\right) \geq c_{e_3}\left(x^{(i)}\right)$ for $1 \leq i \leq 3$. This means, $x^{(3)}$ is also a solution of the minimization problem.

First, we show this for $x^{(1)}$. Let $1 < p < m-1$. Consider x^*, a modification of x where the weight of x_p is distributed among x_1 and x_{m-1} as follows:

$$\forall 1 \leq i \leq d \colon x_i^* = \begin{cases} x_1 + \frac{m-1-p}{m-2}(x_p - 1) & \text{if } i = 1, \\ 1 & \text{if } i = p, \\ x_{m-1} + \frac{p-1}{m-2}(x_p - 1) & \text{if } i = m-1, \\ x_i & \text{else}. \end{cases}$$

To show that x^* also fulfills (3), we show that $c_{e_2}(x) - c_{e_1}(x) = c_{e_2}(x^*) - c_{e_1}(x^*)$ and $c_{e_3}(x) - c_{e_1}(x) = c_{e_3}(x^*) - c_{e_1}(x^*)$. We have

$$c_{e_2}(x) - c_{e_1}(x) = \sum_{i=1}^{m-1} \sum_{j=m}^{d} (i + d - j - (j - i))x_i x_j = \sum_{i=1}^{m-1} \sum_{j=m}^{d} (2i + d - 2j)x_i x_j$$

and

$$c_{e_2}(x^*) - c_{e_1}(x^*)$$

$$= \sum_{i=1}^{m-1} \sum_{j=m}^{d} (2i + d - 2j) x_i^* x_j^*$$

$$= \sum_{i=1}^{m-1} \sum_{j=m}^{d} (2i + d - 2j) x_i x_j + \frac{m-1-p}{m-2}(x_p - 1) \sum_{j=m}^{d} (2 + d - 2j) x_j$$

$$- (x_p - 1) \sum_{j=m}^{d} (2p + d - 2j) x_j + \frac{p-1}{m-2}(x_p - 1) \sum_{j=m}^{d} (2(m-1) + d - 2j) x_j$$

$$= \sum_{i=1}^{m-1} \sum_{j=m}^{d} (2i + d - 2j) x_i x_j + \frac{m-1-p}{m-2}(x_p - 1)(d - m + 1)2$$

$$- (x_p - 1)(d - m + 1)2p + \frac{p-1}{m-2}(x_p - 1)(d - m + 1)2(m-1)$$

$$= \sum_{i=1}^{m-1} \sum_{j=m}^{d} (2i + d - 2j) x_i x_j$$

$$+ (x_p - 1)(d - m + 1)\left(-2p + \frac{2}{m-2}((m-1-p) + (p-1)(m-1))\right)$$

$$= \sum_{i=1}^{m-1} \sum_{j=m}^{d} (2i + d - 2j) x_i x_j + (x_p - 1)(d - m + 1)\left(-2p + \frac{2}{m-2}(m-2)p\right)$$

$$= c_{e_2}(x) - c_{e_1}(x).$$

The calculations for $c_{e_3}(x) - c_{e_1}(x) = c_{e_3}(x^*) - c_{e_1}(x^*)$ are exactly the same with the only difference being the sum indices (first sum goes to m and second sum starts at $m + 1$).

Because of symmetry, for $m + 1 < p < d$, we can similarly distribute weights from x_p to x_{m+1} and x_d. Using this for all $1 < p < m - 1$ and $m + 1 < p < d$ iteratively, we get exactly $x^{(1)}$, which therefore still fulfills (3).

Next, we show that $x^{(2)}$ fulfills (3), too. We have

$$\begin{aligned}
c_{e_1}(x^{(1)}) = {} & (m-2)x_1^{(1)}x_{m-1}^{(1)} + (m-1)x_1^{(1)}x_m^{(1)} + mx_1^{(1)}x_{m+1}^{(1)} + (2m-2)x_1^{(1)}x_d^{(1)} \\
& + x_{m-1}^{(1)}x_m^{(1)} + 2x_{m-1}^{(1)}x_{m+1}^{(1)} + mx_{m-1}^{(1)}x_d^{(1)} + x_m^{(1)}x_{m+1}^{(1)} + (m-1)x_m^{(1)}x_d^{(1)} \\
& + (m-2)x_{m+1}^{(1)}x_d^{(1)} + x_1^{(1)}(m-3)(2m-2) + x_d^{(1)}(m-3)(2m-2) \\
& + x_{m-1}^{(1)}(m-3)m + x_{m+1}^{(1)}(m-3)m + x_m^{(1)}(m-3)m \\
& + \frac{1}{3}(m-4)(m-3)(m-2) + (m-3)(m-3)m, \\
c_{e_2}(x^{(1)}) = {} & (m-2)x_1^{(1)}x_{m-1}^{(1)} + mx_1^{(1)}x_m^{(1)} + (m-1)x_1^{(1)}x_{m+1}^{(1)} + x_1^{(1)}x_d^{(1)} \\
& + (2m-2)x_{m-1}^{(1)}x_m^{(1)} + (2m-3)x_{m-1}^{(1)}x_{m+1}^{(1)} + (m-1)x_{m-1}^{(1)}x_d^{(1)} + x_m^{(1)}x_{m+1}^{(1)} \\
& + (m-1)x_m^{(1)}x_d^{(1)} + (m-2)x_{m+1}^{(1)}x_d^{(1)} \\
& + x_1^{(1)}(m-3)(m-1) + x_d^{(1)}(m-3)(m-1) + x_{m-1}^{(1)}(m-3)(2m-3) \\
& + x_{m+1}^{(1)}(m-3)(2m-3) + x_m^{(1)}(m-3)(2m-1) \\
& + \frac{1}{3}(m-4)(m-3)(m-2) + (m-3)(m-3)(m-1) \text{ and}
\end{aligned}$$

$$c_{e_3}(x^{(1)}) = (m-2)x_1^{(1)}x_{m-1}^{(1)} + (m-1)x_1^{(1)}x_m^{(1)} + (m-1)x_1^{(1)}x_{m+1}^{(1)} + x_1^{(1)}x_d^{(1)}$$

$$+ x_{m-1}^{(1)}x_m^{(1)} + (2m-3)x_{m-1}^{(1)}x_{m+1}^{(1)} + (m-1)x_{m-1}^{(1)}x_d^{(1)} + (2m-2)x_m^{(1)}x_{m+1}^{(1)}$$

$$+ mx_m^{(1)}x_d^{(1)} + (m-2)x_{m+1}^{(1)}x_d^{(1)}$$

$$+ x_1^{(1)}(m-3)(m-1) + x_d^{(1)}(m-3)(m-1) + x_{m-1}^{(1)}(m-3)(2m-3)$$

$$+ x_{m+1}^{(1)}(m-3)(2m-3) + x_m^{(1)}(m-3)(2m-1)$$

$$+ \frac{1}{3}(m-4)(m-3)(m-2) + (m-3)(m-3)(m-1).$$

We can further assume that $(2m-3)x_1^{(1)} + x_{m-1}^{(1)} = (2m-3)x_d^{(1)} + x_{m+1}^{(1)}$. This is because if (without loss of generality) $(2m-3)x_1^{(1)} + x_{m-1}^{(1)} < (2m-3)x_d^{(1)} + x_{m+1}^{(1)}$ holds, we can move value from $x_{m-1}^{(1)}$ to $x_1^{(1)}$ without changing $\Delta d(x^{(1)})$. This increases $c_{e_1}(x^{(1)})$ more than $c_{e_2}(x^{(1)})$ and $c_{e_3}(x^{(1)})$.

Let (without loss of generality) $x_1^{(1)} \geq x_d^{(1)}$ and $y := x_1^{(1)} - x_d^{(1)}$. Then we have $x_d^{(1)} = x_1^{(1)} - y$ and $x_{m+1}^{(1)} = x_{m-1}^{(1)} + (2m-3)y$. We see, that

$$\forall 1 \leq i \leq d \colon x_i^{(2)} = \begin{cases} x_m^{(1)} + (2m-4)y & \text{if } i = m, \\ x_{m-1}^{(1)} & \text{if } i = m-1 \text{ or } i = m+1, \\ x_1^{(1)} & \text{if } i = 1 \text{ or } i = d, \\ 1 & \text{else.} \end{cases}$$

We show $c_{e_1}(x^{(2)}) \geq c_{e_3}(x^{(2)})$, by showing that $c_{e_1}(x^{(1)}) - c_{e_3}(x^{(1)}) - c_{e_1}(x^{(2)}) + c_{e_3}(x^{(2)}) > 0$. We have

$$\Delta x^{(1)} := c_{e_1}(x^{(1)}) - c_{e_3}(x^{(1)})$$

$$= x_1^{(1)}x_{m+1}^{(1)} + (2m-3)x_1^{(1)}x_d^{(1)} - (2m-5)x_{m-1}^{(1)}x_{m+1}^{(1)} + x_{m-1}^{(1)}x_d^{(1)}$$

$$- (2m-3)x_m^{(1)}x_{m+1}^{(1)} - x_m^{(1)}x_d^{(1)} + x_1^{(1)}(m-3)(m-1) + x_d^{(1)}(m-3)(m-1)$$

$$- x_{m-1}^{(1)}(m-3)(m-3) - x_{m+1}^{(1)}(m-3)(m-3) - x_m^{(1)}(m-3)(m-1)$$

$$+ (m-3)(m-3)$$

$$= x_1^{(1)}x_{m-1}^{(1)} + x_1^{(1)}(2m-3)y + (2m-3)x_1^{(1)}x_1^{(1)} - (2m-3)x_1^{(1)}y$$

$$- (2m-5)x_{m-1}^{(1)}x_{m-1}^{(1)} - (2m-5)(2m-3)x_{m-1}^{(1)}y + x_{m-1}^{(1)}x_1^{(1)} - x_{m-1}^{(1)}y$$

$$- (2m-3)x_m^{(1)}x_{m-1}^{(1)} - (2m-3)x_m^{(1)}(2m-3)y - x_m^{(1)}x_1^{(1)} + x_m^{(1)}y$$

$$+ x_1^{(1)}(m-3)(m-1) + x_1^{(1)}(m-3)(m-1) - y(m-3)(m-1)$$

$$- x_{m-1}^{(1)}(m-3)(m-3) - x_{m-1}^{(1)}(m-3)(m-3)$$

$$- (2m-3)y(m-3)(m-3) - x_m^{(1)}(m-3)(m-1) + (m-3)(m-3)$$

and

$$\Delta x^{(2)} := c_{e_1}(x^{(2)}) - c_{e_3}(x^{(2)})$$

$$= x_1^{(1)}x_{m-1}^{(1)} + (2m-3)x_1^{(1)}x_1^{(1)} - (2m-5)x_{m-1}^{(1)}x_{m-1}^{(1)} + x_{m-1}^{(1)}x_1^{(1)}$$

$$- (2m-3)x_m^{(1)}x_{m-1}^{(1)} - (2m-3)(2m-4)yx_{m-1}^{(1)} - x_m^{(1)}x_1^{(1)} - (2m-4)yx_1^{(1)}$$

$$+ x_1^{(1)}(m-3)(m-1) + x_1^{(1)}(m-3)(m-1) - x_{m-1}^{(1)}(m-3)(m-3)$$

$$- x_{m-1}^{(1)}(m-3)(m-3) - x_m^{(1)}(m-3)(m-1) - (2m-4)y(m-3)(m-1)$$

$$+ (m-3)(m-3),$$

and therefore

$$
\begin{aligned}
\Delta x^{(2)} - \Delta x^{(1)} &= (2m-5)(2m-3)x_{m-1}^{(1)}y + x_{m-1}^{(1)}y + (2m-3)x_m^{(1)}(2m-3)y \\
&\quad - (2m-3)(2m-4)yx_{m-1}^{(1)} - x_m^{(1)}y - (2m-4)yx_1^{(1)} + y(m-3)(m-1) \\
&\quad + (2m-3)y(m-3)(m-3) - (2m-4)y(m-3)(m-1) \\
&= x_{m-1}^{(1)}y((2m-5)(2m-3) + 1 - (2m-3)(2m-4)) \\
&\quad + x_m^{(1)}y((2m-3)(2m-3) - 1) - (2m-4)x_1^{(1)}y \\
&\quad + y(m-3)(m-1 + (2m-3)(m-3) - (2m-4)(m-1)) \\
&= -2x_{m-1}^{(1)}y(m-2) + 4x_m^{(1)}y(m-2)(m-1) - 2x_1^{(1)}y(m-2) \\
&\quad - 2y(m-3)(m-2) \\
&= 2y(m-2)(-x_{m-1}^{(1)} + 2x_m^{(1)}(m-1) - x_1^{(1)} - m + 3) \\
&= 2y(m-2)(x_m^{(1)} - x_1^{(1)} - x_{m-1}^{(1)} + 2x_m^{(1)}(m-1.5) - m + 3) \\
&\geq 0.
\end{aligned}
$$

The last step follows from $x_m^{(1)} \geq x_1^{(1)} + x_{m-1}^{(1)} \geq 1$, $m \geq 3$ and $y \geq 0$.

Since $x^{(1)}$ fulfills (3), we see that $c_{e_1}(x^{(2)}) \geq c_{e_3}(x^{(2)})$ holds, too. With $x^{(2)}$ being symmetric, $c_{e_1}(x^{(2)}) \geq c_{e_2}(x^{(2)})$ holds as well. This shows that $x^{(2)}$ also fulfills (3).

Now, we show that $x^{(3)}$ fulfills (3), too. Let $y := \frac{x_{m-1}^{(2)} - 1}{2m-3}$. We see that

$$
\forall 1 \leq i \leq d \colon x_i^{(3)} = \begin{cases} x_m^{(2)} + 2(2m-4)y & \text{if } i = m, \\ x_1^{(2)} + y & \text{if } i = 1 \text{ or } i = d, \\ 1 & \text{else.} \end{cases}
$$

We see that

$$
\begin{aligned}
\Delta x^{(2)} &= 2x_1^{(2)} + 2(2m-3)yx_1^{(2)} + (2m-3)x_1^{(2)}x_1^{(2)} - (2m-5)(1 + (2m-3)y)^2 \\
&\quad - (2m-3)x_m^{(2)} - (2m-3)x_m^{(2)}(2m-3)y - x_1^{(2)}x_m^{(2)} + 2x_1^{(2)}(m-3)(m-1) \\
&\quad - 2(m-3)(m-3) - 2(2m-3)y(m-3)(m-3) - x_m^{(2)}(m-3)(m-1) \\
&\quad + (m-3)(m-3) \\
&= 2x_1^{(2)} + (2m-3)x_1^{(2)}x_1^{(2)} - (2m-5) - (2m-3)x_m^{(2)} - x_1^{(2)}x_m^{(2)} \\
&\quad + 2x_1^{(2)}(m-3)(m-1) - 2(m-3)(m-3) - x_m^{(2)}(m-3)(m-1) \\
&\quad + (m-3)(m-3) \\
&\quad + y(2m-3)(2x_1^{(2)} - (2m-5)2 - (2m-3)x_m^{(2)} - 2(m-3)(m-3)) \\
&\quad - y^2(2m-3)^2(2m-5), \\
\Delta x^{(3)} &:= c_{e_1}(x^{(3)}) - c_{e_2}(x^{(3)}) \\
&= 2x_1^{(2)} + 2y + (2m-3)(x_1^{(2)} + y)^2 - (2m-5) - (2m-3)(x_m^{(2)} + 2(2m-4)y) \\
&\quad - (x_1^{(2)} + y)(x_m^{(2)} + 2(2m-4)y) + 2(x_1^{(2)} + y)(m-3)(m-1) \\
&\quad - 2(m-3)(m-3) - (x_m^{(2)} + 2(2m-4)y)(m-3)(m-1) \\
&\quad + (m-3)(m-3)
\end{aligned}
$$

$$\begin{aligned}
&= 2x_1^{(2)} + (2m-3)x_1^{(2)}x_1^{(2)} - (2m-5) - (2m-3)x_m^{(2)} - x_1^{(2)}x_m^{(2)} \\
&\quad + 2x_1^{(2)}(m-3)(m-1) - 2(m-3)(m-3) - x_m^{(2)}(m-3)(m-1) \\
&\quad + (m-3)(m-3) \\
&\quad + y(2 + (2m-3)2x_1^{(2)} - (2m-3)2(2m-4) - x_m^{(2)} - 2(2m-4)x_1^{(2)} \\
&\qquad + 2(m-3)(m-1) - 2(2m-4)(m-3)(m-1)) \\
&\quad + y^2((2m-3) - 2(2m-4))
\end{aligned}$$

and obtain

$$\begin{aligned}
\Delta x^{(3)} - \Delta x^{(2)} &= y(2 - 2(2m-3) + (2m-3)^2 x_m^{(2)} + 2(2m-3)(m-3)(m-3) - x_m^{(2)} \\
&\qquad - 2(2m-4)x_1^{(2)} + 2(m-3)(m-1) - 2(2m-4)(m-3)(m-1)) \\
&\quad + y^2((2m-3) - 2(2m-4) + (2m-3)^2(2m-5)) \\
&= y(m-2)(-4x_1^{(2)} + 4(m-1)x_m^{(2)} - 4(m-2)) \\
&\quad + 4y^2(2m-5)(m-1)(m-2) \\
&\geq 0.
\end{aligned}$$

The last step follows from $x_m^{(2)} \geq x_1^{(2)}$, $m \geq 3$ and $y \geq 0$.

Since $x^{(2)}$ fulfills (3), $x^{(3)}$ does, too. Because of $x^{(3)}$ being a node distribution with only two variables ($x_1^{(3)} = x_d^{(3)}$ and $x_m^{(3)}$), we can simplify $\Delta x^{(3)}$ and Δd to

$$\begin{aligned}
\Delta x^{(3)} &= c_{e_1}(x^{(3)}) - c_{e_2}(x^{(3)}) \\
&= 2x_1^{(3)} + (2m-3)x_1^{(3)}x_1^{(3)} - (2m-5) - (2m-3)x_m^{(3)} - x_1^{(3)}x_m^{(3)} \\
&\quad + 2x_1^{(3)}(m-3)(m-1) - 2(m-3)(m-3) - x_m^{(3)}(m-3)(m-1) \\
&\quad + (m-3)(m-3) \\
&= (2m-3)x_1^{(3)}x_1^{(3)} + 2x_1^{(3)}(m-2)^2 - x_1^{(3)}x_m^{(3)} - m(m-2)x_m^{(3)} - (m-2)^2
\end{aligned}$$

and

$$\Delta d(x^{(3)}) = (m-2)^2 + (2m-3)x_1^{(3)}.$$

We now prove the lower bound.

We see that $\Delta d(x^{(3)})$ is only dependent on $x_1^{(3)}$ which means we have to minimize $x_1^{(3)}$. Since $x^{(3)}$ sums to n, we have $n = 2x_1^{(3)} + x_m^{(3)} + 2m - 4$, and therefore $x_m^{(3)} = n - 2x_1^{(3)} - 2m + 4$. Substituting this into $\Delta x^{(3)}$ yields

$$\begin{aligned}
\Delta x^{(3)} &= (2m-3)x_1^{(3)}x_1^{(3)} + 2x_1^{(3)}(m-2)^2 - x_1^{(3)}(n - 2x_1^{(3)} - 2m + 4) \\
&\quad - m(m-2)(n - 2x_1^{(3)} - 2m + 4) - (m-2)^2 \\
&= x_1^{(3)}x_1^{(3)}(2m-1) + x_1^{(3)}(2(m-2)(2m-1) - n) + (2m-1)(m-2)^2 \\
&\quad - nm(m-2).
\end{aligned}$$

Observe that $x^{(3)}$ fulfills (3) if and only if $\Delta x^{(3)} \geq 0$. Solving the quadratic equation, we see that this is only the case for

$$\begin{aligned}
x_1^{(3)} \geq{}& -(m-2) + \frac{n}{2(2m-1)} \\
&+ \sqrt{\left(\frac{n}{2(2m-1)} - (m-2)\right)^2 + nm(m-2) - (2m-1)(m-2)^2}.
\end{aligned}$$

With (4), we see that the term under the root is larger than

$$0^2 + 3(m+1)m(m-2) - (2m-1)(m-2)^2$$
$$\geq (m+1)m(m-2)$$
$$\geq (m-2)^2.$$

This yields

$$x_1^{(3)} \geq -(m-2) + \frac{n}{2(2m-1)} + \sqrt{(m-2)^2} = \frac{n}{2(2m-1)},$$

and therefore

$$\Delta d(x^{(3)}) \geq (m-2)^2 + \frac{2m-3}{2m-1} \cdot \frac{n}{2} \geq \frac{n}{3}.$$

Case d even: Let $m = \frac{d}{2}$ and $e_2 := \{v_{m-1}, v_m\}$ and $e_3 := \{v_{m+1}, v_{m+2}\}$. (see Figure 5 (right)) Again, we will only consider the two constraints

$$c_{e_1}(x) \geq c_{e_2}(x) \qquad \text{and} \qquad c_{e_1}(x) \geq c_{e_3}(x), \tag{5}$$

where

$$c_{e_1} = \sum_{i=1}^{d-1} \sum_{j=i+1}^{d} (j-i)x_i x_j,$$

$$c_{e_2} = \sum_{i=1}^{m-2} \sum_{j=i+1}^{m-1} (j-i)x_i x_j + \sum_{i=m}^{d-1} \sum_{j=i+1}^{d} (j-i)x_i x_j + \sum_{i=1}^{m-1} \sum_{j=m}^{d} (i+d-j)x_i x_j,$$

$$c_{e_3} = \sum_{i=1}^{m} \sum_{j=i+1}^{m+1} (j-i)x_i x_j + \sum_{i=m+2}^{d-1} \sum_{j=i+1}^{d} (j-i)x_i x_j + \sum_{i=1}^{m+1} \sum_{j=m+2}^{d} (i+d-j)x_i x_j,$$

We observe that the claim is trivially true for $\frac{n}{3} \leq \frac{d+6}{2}$, since $\Delta d(x) \geq \frac{d+6}{2}$ (tight for $n = 6, d = 6, x = (1,1,1,1,1,1)$). This means we can assume

$$n > \frac{3(d+6)}{2}, \qquad \text{or rather} \qquad n > 3(m+3). \tag{6}$$

Similar to the odd case, we perform a series of relaxations. For that, we define the node distributions $x^{(1)}, x^{(2)}, x^{(3)}, x^{(4)}$ such that for all $1 \leq i \leq d$

$$x_i^{(1)} := \begin{cases} x_1 + \sum_{p=2}^{m-2} \frac{m-1-p}{m-2}(x_p - 1) & \text{if } i = 1, \\ x_{m-1} + \sum_{p=2}^{m-2} \frac{p-1}{m-2}(x_p - 1) & \text{if } i = m-1, \\ x_m & \text{if } i = m, \\ x_{m+1} & \text{if } i = m+1, \\ x_{m+2} + \sum_{p=m+3}^{d-1} \frac{d-p}{m-2}(x_p - 1) & \text{if } i = m+1, \\ x_d + \sum_{p=m+3}^{d-1} \frac{p-m-2}{m-2}(x_p - 1) & \text{if } i = d, \\ 1 & \text{else,} \end{cases}$$

(Without loss of generality, we assume $x_1^{(1)} \geq x_d^{(1)}$

and $(m-1)x_1^{(1)} + x_{m-1}^{(1)} = (m-1)x_d^{(1)} + x_{m+2}^{(1)}$.)

$$x_i^{(2)} := \begin{cases} x_1^{(1)} & \text{if } i = 1 \text{ or } i = d, \\ x_{m-1}^{(1)} & \text{if } i = m - 1 \text{ or } i = m + 2, \\ x_m^{(1)} & \text{if } i = m \\ x_{m+1}^{(1)} + (m - 2)\left(x_1^{(1)} - x_d^{(1)}\right) & \text{if } i = m + 1, \\ 1 & \text{else,} \end{cases}$$

$$x_i^{(3)} := \begin{cases} x_1^{(2)} & \text{if } i = 1 \text{ or } i = d, \\ x_{m-1}^{(2)} & \text{if } i = m - 1 \text{ or } i = m + 2, \\ \frac{x_m^{(2)} + x_{m+1}^{(2)}}{2} & \text{if } i = m \text{ or } i = m + 1, \\ 1 & \text{else,} \end{cases}$$

$$x_i^{(4)} := \begin{cases} x_1^{(3)} + \frac{1}{m-1}\left(x_{m-1}^{(3)} - 1\right) & \text{if } i = 1 \text{ or } i = d, \\ x_m^{(3)} + \frac{m-2}{m-1}\left(x_{m-1}^{(3)} - 1\right) & \text{if } i = m \text{ or } i = m + 1, \\ 1 & \text{else.} \end{cases}$$

Again, it is easy to see that $\Delta d(x) = \Delta d\left(x^{(1)}\right) = \cdots = \Delta d\left(x^{(4)}\right)$. We show $c_{e_1}\left(x^{(i)}\right) \geq c_{e_2}\left(x^{(i)}\right)$ and $c_{e_1}\left(x^{(i)}\right) \geq c_{e_3}\left(x^{(i)}\right)$, for $1 \leq i \leq 4$. Therefore, $x^{(4)}$ is a solution of the minimization problem.

First, we observe that $c_{e_2}(x) - c_{e_1}(x) = c_{e_2}(x^{(1)}) - c_{e_1}(x^{(1)})$ and $c_{e_3}(x) - c_{e_1}(x) = c_{e_3}(x^{(1)}) - c_{e_1}(x^{(1)})$ follow the same way as in the odd case (with slight adjustments to the sum indices). This means that $x^{(1)}$ fulfills (5).

Next, we show that $x^{(2)}$ fulfills (5), too. We have

$$
\begin{aligned}
c_{e_1}(x^{(1)}) = {} & (m-2)x_1^{(1)}x_{m-1}^{(1)} + (m-1)x_1^{(1)}x_m^{(1)} + mx_1^{(1)}x_{m+1}^{(1)} + (m+1)x_1^{(1)}x_{m+2}^{(1)} \\
& + (2m-1)x_1^{(1)}x_d^{(1)} + x_{m-1}^{(1)}x_m^{(1)} + 2x_{m-1}^{(1)}x_{m+1}^{(1)} + 3x_{m-1}^{(1)}x_{m+2}^{(1)} \\
& + (m+1)x_{m-1}^{(1)}x_d^{(1)} + x_m^{(1)}x_{m+1}^{(1)} + 2x_m^{(1)}x_{m+2}^{(1)} + mx_m^{(1)}x_d^{(1)} \\
& + x_{m+1}^{(1)}x_{m+2}^{(1)} + (m-1)x_{m+1}^{(1)}x_d^{(1)} + (m-2)x_{m+2}^{(1)}x_d^{(1)} \\
& + (m-3)(2m-1)x_1^{(1)} + (m-3)(m+1)x_{m-1}^{(1)} + (m-3)(m+1)x_m^{(1)} \\
& + (m-3)(m+1)x_{m+1}^{(1)} + (m-3)(m+1)x_{m+2}^{(1)} + (m-3)(2m-1)x_d^{(1)} \\
& + \frac{1}{3}(m-4)(m-3)(m-2) + (m-3)(m-3)(m+1),
\end{aligned}
$$

$$
\begin{aligned}
c_{e_2}(x^{(1)}) = {} & (m-2)x_1^{(1)}x_{m-1}^{(1)} + (m+1)x_1^{(1)}x_m^{(1)} + mx_1^{(1)}x_{m+1}^{(1)} + (m-1)x_1^{(1)}x_{m+2}^{(1)} \\
& + x_1^{(1)}x_d^{(1)} + (2m-1)x_{m-1}^{(1)}x_m^{(1)} + (2m-2)x_{m-1}^{(1)}x_{m+1}^{(1)} + (2m-3)x_{m-1}^{(1)}x_{m+2}^{(1)} \\
& + (m-1)x_{m-1}^{(1)}x_d^{(1)} + x_m^{(1)}x_{m+1}^{(1)} + 2x_m^{(1)}x_{m+2}^{(1)} + mx_m^{(1)}x_d^{(1)} \\
& + x_{m+1}^{(1)}x_{m+2}^{(1)} + (m-1)x_{m+1}^{(1)}x_d^{(1)} + (m-2)x_{m+2}^{(1)}x_d^{(1)} \\
& + (m-3)(m-1)x_1^{(1)} + (m-3)(2m-3)x_{m-1}^{(1)} + (m-3)(2m+1)x_m^{(1)} \\
& + (m-3)(2m-1)x_{m+1}^{(1)} + (m-3)(2m-3)x_{m+2}^{(1)} + (m-3)(m-1)x_d^{(1)} \\
& + \frac{1}{3}(m-4)(m-3)(m-2) + (m-3)(m-3)(m-1) \text{ and}
\end{aligned}
$$

$$
\begin{aligned}
c_{e_3}(x^{(1)}) = {} & (m-2)x_1^{(1)}x_{m-1}^{(1)} + (m-1)x_1^{(1)}x_m^{(1)} + mx_1^{(1)}x_{m+1}^{(1)} + (m-1)x_1^{(1)}x_{m+2}^{(1)} \\
& + x_1^{(1)}x_d^{(1)} + x_{m-1}^{(1)}x_m^{(1)} + 2x_{m-1}^{(1)}x_{m+1}^{(1)} + (2m-3)x_{m-1}^{(1)}x_{m+2}^{(1)} \\
& + (m-1)x_{m-1}^{(1)}x_d^{(1)} + x_m^{(1)}x_{m+1}^{(1)} + (2m-2)x_m^{(1)}x_{m+2}^{(1)} + mx_m^{(1)}x_d^{(1)} \\
& + (2m-1)x_{m+1}^{(1)}x_{m+2}^{(1)} + (m+1)x_{m+1}^{(1)}x_d^{(1)} + (m-2)x_{m+2}^{(1)}x_d^{(1)} \\
& + (m-3)(m-1)x_1^{(1)} + (m-3)(2m-3)x_{m-1}^{(1)} + (m-3)(2m-1)x_m^{(1)} \\
& + (m-3)(2m+1)x_{m+1}^{(1)} + (m-3)(2m-3)x_{m+2}^{(1)} + (m-3)(m-1)x_d^{(1)} \\
& + \frac{1}{3}(m-4)(m-3)(m-2) + (m-3)(m-3)(m-1).
\end{aligned}
$$

Similar to the odd case, we can further assume that $(m-1)x_1^{(1)} + x_{m-1}^{(1)} = (m-1)x_d^{(1)} + x_{m+2}^{(1)}$. If this is not the case and (without loss of generality) $(m-1)x_1^{(1)} + x_{m-1}^{(1)} > (m-1)x_d^{(1)} + x_{m+2}^{(1)}$, we could move weight from $x_{m+2}^{(1)}$ to $x_d^{(1)}$ without changing $\Delta d(x^{(1)})$. This would increase $c_{e_1}(x^{(1)})$ more than $c_{e_2}(x^{(1)})$ and $c_{e_3}(x^{(1)})$.

Let (without loss of generality) $x_1^{(1)} \geq x_d^{(1)}$ and $y := x_1^{(1)} - x_d^{(1)}$. Therefore, we have $x_d^{(1)} = x_1^{(1)} - y$ and $x_{m+2}^{(1)} = x_{m-1}^{(1)} + (m-1)y$. We see that

$$
\forall 1 \leq i \leq d\colon x_i^{(2)} = \begin{cases}
x_m^{(1)} & \text{if } i = m, \\
x_{m+1}^{(1)} + (m-2)y & \text{if } i = m+1, \\
x_{m-1}^{(1)} & \text{if } i = m-1 \text{ or } i = m+2, \\
x_1^{(1)} & \text{if } i = 1 \text{ or } i = d, \\
1 & \text{else.}
\end{cases}
$$

We now show that

$$
\Delta_2 x^{(2)} := c_{e_1}(x^{(2)}) - c_{e_2}(x^{(2)}) \geq c_{e_1}(x^{(1)}) - c_{e_2}(x^{(1)}) = \Delta_2(x^{(1)})
$$

and

$$
\Delta_3 x^{(2)} := c_{e_1}(x^{(2)}) - c_{e_3}(x^{(2)}) \geq c_{e_1}(x^{(1)}) - c_{e_3}(x^{(1)}) = \Delta_3(x^{(1)}).
$$

We have

$$
\begin{aligned}
\Delta_2 x^{(1)} = {} & c_{e_1}(x^{(1)}) - c_{e_2}(x^{(1)}) \\
= {} & -2x_1^{(1)}x_m^{(1)} + 2x_1^{(1)}x_{m+2}^{(1)} + (2m-2)x_1^{(1)}x_d^{(1)} - (2m-2)x_{m-1}^{(1)}x_m^{(1)} \\
& - (2m-4)x_{m-1}^{(1)}x_{m+1}^{(1)} - (2m-6)x_{m-1}^{(1)}x_{m+2}^{(1)} + 2x_{m-1}^{(1)}x_d^{(1)} \\
& + (m-3)mx_1^{(1)} - (m-3)(m-4)x_{m-1}^{(1)} - (m-3)mx_m^{(1)} \\
& - (m-3)(m-2)x_{m+1}^{(1)} - (m-3)(m-4)x_{m+2}^{(1)} + (m-3)mx_d^{(1)} \\
& + 2(m-3)(m-3) \\
= {} & -2x_1^{(1)}x_m^{(1)} + 2x_1^{(1)}x_{m-1}^{(1)} + 2(m-1)x_1^{(1)}y + (2m-2)x_1^{(1)}x_1^{(1)} - (2m-2)x_1^{(1)}y \\
& - (2m-2)x_{m-1}^{(1)}x_m^{(1)} - (2m-4)x_{m-1}^{(1)}x_{m+1}^{(1)} - (2m-6)x_{m-1}^{(1)}x_{m-1}^{(1)} \\
& - (2m-6)x_{m-1}^{(1)}(m-1)y + 2x_{m-1}^{(1)}x_1^{(1)} - 2x_{m-1}^{(1)}y + (m-3)mx_1^{(1)} \\
& - (m-3)(m-4)x_{m-1}^{(1)} - (m-3)mx_m^{(1)} - (m-3)(m-2)x_{m+1}^{(1)} \\
& - (m-3)(m-4)x_{m-1}^{(1)} - (m-3)(m-4)(m-1)y + (m-3)mx_1^{(1)} \\
& - (m-3)my + 2(m-3)(m-3)
\end{aligned}
$$

and

$$
\begin{aligned}
\Delta_2 x^{(2)} = &- 2x_1^{(1)}x_m^{(1)} + 2x_1^{(1)}x_{m-1}^{(1)} + (2m-2)x_1^{(1)}x_1^{(1)} - (2m-2)x_{m-1}^{(1)}x_m^{(1)} \\
&- (2m-4)x_{m-1}^{(1)}x_{m+1}^{(1)} - (2m-4)x_{m-1}^{(1)}(m-2)y - (2m-6)x_{m-1}^{(1)}x_{m-1}^{(1)} \\
&+ 2x_{m-1}^{(1)}x_1^{(1)} + (m-3)mx_1^{(1)} - (m-3)(m-4)x_{m-1}^{(1)} - (m-3)mx_m^{(1)} \\
&- (m-3)(m-2)x_{m+1}^{(1)} - (m-3)(m-2)(m-2)y \\
&- (m-3)(m-4)x_{m-1}^{(1)} + (m-3)mx_1^{(1)} + 2(m-3)(m-3),
\end{aligned}
$$

and therefore

$$
\begin{aligned}
\Delta_2 x^{(2)} - \Delta_2 x^{(1)} = &- (2m-4)x_{m-1}^{(1)}(m-2)y + (2m-6)x_{m-1}^{(1)}(m-1)y \\
&+ 2x_{m-1}^{(1)}y - (m-3)(m-2)(m-2)y \\
&+ (m-3)(m-4)(m-1)y + (m-3)my \\
= &\, x_{m-1}^{(1)}y(-(2m-4)(m-2) + (2m-6)(m-1) + 2) \\
&+ (m-3)y(-(m-2)(m-2) + (m-4)(m-1) + m) \\
= &\, x_{m-1}^{(1)}y \cdot 0 + (m-3)y \cdot 0 \\
= &\, 0.
\end{aligned}
$$

Furthermore, we see that

$$
\begin{aligned}
\Delta_3 x^{(1)} = &\, c_{e_1}(x^{(1)}) - c_{e_3}(x^{(1)}) \\
= &\, 2x_1^{(1)}x_{m+2}^{(1)} + (2m-2)x_1^{(1)}x_d^{(1)} - (2m-6)x_{m-1}^{(1)}x_{m+2}^{(1)} + 2x_{m-1}^{(1)}x_d^{(1)} \\
&- (2m-4)x_m^{(1)}x_{m+2}^{(1)} - (2m-2)x_{m+1}^{(1)}x_{m+2}^{(1)} - 2x_{m+1}^{(1)}x_d^{(1)} \\
&+ (m-3)mx_1^{(1)} - (m-3)(m-4)x_{m-1}^{(1)} - (m-3)(m-2)x_m^{(1)} \\
&- (m-3)mx_{m+1}^{(1)} - (m-3)(m-4)x_{m+2}^{(1)} + (m-3)mx_d^{(1)} \\
&+ 2(m-3)(m-3) \\
= &\, 2x_1^{(1)}x_{m-1}^{(1)} + 2x_1^{(1)}(m-1)y + (2m-2)x_1^{(1)}x_1^{(1)} - (2m-2)x_1^{(1)}y \\
&- (2m-6)x_{m-1}^{(1)}x_{m-1}^{(1)} - (2m-6)x_{m-1}^{(1)}(m-1)y + 2x_{m-1}^{(1)}x_1^{(1)} - 2x_{m-1}^{(1)}y \\
&- (2m-4)x_m^{(1)}x_{m-1}^{(1)} - (2m-4)x_m^{(1)}(m-1)y - (2m-2)x_{m+1}^{(1)}x_{m-1}^{(1)} \\
&- (2m-2)x_{m+1}^{(1)}(m-1)y - 2x_{m+1}^{(1)}x_1^{(1)} + 2x_{m+1}^{(1)}y \\
&+ (m-3)mx_1^{(1)} - (m-3)(m-4)x_{m-1}^{(1)} - (m-3)(m-2)x_m^{(1)} \\
&- (m-3)mx_{m+1}^{(1)} - (m-3)(m-4)x_{m-1}^{(1)} - (m-3)(m-4)(m-1)y \\
&+ (m-3)mx_1^{(1)} - (m-3)my + 2(m-3)(m-3)
\end{aligned}
$$

and

$$
\begin{aligned}
\Delta_3 x^{(2)} = &\, 2x_1^{(1)}x_{m-1}^{(1)} + (2m-2)x_1^{(1)}x_1^{(1)} - (2m-6)x_{m-1}^{(1)}x_{m-1}^{(1)} + 2x_{m-1}^{(1)}x_1^{(1)} \\
&- (2m-4)x_m^{(1)}x_{m-1}^{(1)} - (2m-2)x_{m+1}^{(1)}x_{m-1}^{(1)} - (2m-2)(m-2)yx_{m-1}^{(1)} \\
&- 2x_{m+1}^{(1)}x_1^{(1)} - 2(m-2)yx_1^{(1)} + (m-3)mx_1^{(1)} - (m-3)(m-4)x_{m-1}^{(1)} \\
&- (m-3)(m-2)x_m^{(1)} - (m-3)mx_{m+1}^{(1)} - (m-3)m(m-2)y \\
&- (m-3)(m-4)x_{m-1}^{(1)} + (m-3)mx_1^{(1)} + 2(m-3)(m-3),
\end{aligned}
$$

and therefore

$$
\begin{aligned}
\Delta_3 x^{(2)} - \Delta_3 x^{(1)} = {} & (2m-6)x_{m-1}^{(1)}(m-1)y + 2x_{m-1}^{(1)}y + (2m-4)x_m^{(1)}(m-1)y \\
& - (2m-2)(m-2)yx_{m-1}^{(1)} + (2m-2)x_{m+1}^{(1)}(m-1)y - 2(m-2)yx_1^{(1)} \\
& - 2x_{m+1}^{(1)}y - (m-3)m(m-2)y + (m-3)(m-4)(m-1)y \\
& + (m-3)my \\
= {} & -x_1^{(1)}y(2m-4) \\
& + x_{m-1}^{(1)}y((2m-6)(m-1) + 2 - (2m-2)(m-2)) \\
& + x_m^{(1)}y(2m-4)(m-1) \\
& + x_{m+1}^{(1)}y((2m-2)(m-1) - 2) \\
& + (m-3)y(-(m-2)m + (m-4)(m-1) + m) \\
= {} & 2(m-2)y(-x_1^{(1)} - x_{m-1}^{(1)} + (m-1)x_m^{(1)} + mx_{m+1}^{(1)} - (m-3)) \\
\geq {} & 0.
\end{aligned}
$$

The inequality in the last step comes from $m \geq 3$, $y \geq 0$, $x_1^{(1)} + x_{m-1}^{(1)} \leq \frac{n}{3}$ and $x_m^{(1)} + x_{m+1}^{(1)} \geq \frac{n}{3}$.

Since $x^{(1)}$ fulfills (5) and we have $\Delta_2 x^{(2)} - \Delta_2 x^{(1)} = 0$ and $\Delta_3 x^{(2)} - \Delta_3 x^{(1)} \geq 0$, the node distribution $x^{(2)}$ fulfills (5), too.

Now, we show that $x^{(3)}$ fulfills (5), as well. Without loss of generality, we can assume $x_{m+1}^{(2)} \geq x_m^{(2)}$.

We have

$$
\begin{aligned}
\Delta_3 x^{(2)} = {} & 4x_1^{(2)}x_{m-1}^{(2)} + (2m-2)x_1^{(2)}x_1^{(2)} - (2m-6)x_{m-1}^{(2)}x_{m-1}^{(2)} - (2m-4)x_m^{(2)}x_{m-1}^{(2)} \\
& - (2m-2)x_{m+1}^{(2)}x_{m-1}^{(2)} - 2x_{m+1}^{(2)}x_1^{(2)} + 2(m-3)mx_1^{(2)} - 2(m-3)(m-4)x_{m-1}^{(2)} \\
& - (m-3)(m-2)x_m^{(2)} - (m-3)mx_{m+1}^{(2)} + 2(m-3)(m-3)
\end{aligned}
$$

and

$$
\begin{aligned}
\Delta_3 x^{(3)} = {} & 4x_1^{(2)}x_{m-1}^{(2)} + (2m-2)x_1^{(2)}x_1^{(2)} - (2m-6)x_{m-1}^{(2)}x_{m-1}^{(2)} \\
& - (2m-4)x_{m-1}^{(2)}\frac{x_m^{(2)} + x_{m+1}^{(2)}}{2} - (2m-2)x_{m-1}^{(2)}\frac{x_m^{(2)} + x_{m+1}^{(2)}}{2} - 2x_1^{(2)}\frac{x_m^{(2)} + x_{m+1}^{(2)}}{2} \\
& + 2(m-3)mx_1^{(2)} - 2(m-3)(m-4)x_{m-1}^{(2)} - (m-3)(m-2)\frac{x_m^{(2)} + x_{m+1}^{(2)}}{2} \\
& - (m-3)m\frac{x_m^{(2)} + x_{m+1}^{(2)}}{2} + 2(m-3)(m-3) \\
= {} & 4x_1^{(2)}x_{m-1}^{(2)} + (2m-2)x_1^{(2)}x_1^{(2)} - (2m-6)x_{m-1}^{(2)}x_{m-1}^{(2)} - (2m-3)x_{m-1}^{(2)}x_m^{(2)} \\
& - (2m-3)x_{m-1}^{(2)}x_{m+1}^{(2)} - x_1^{(2)}x_m^{(2)} - x_1^{(2)}x_{m+1}^{(2)} + 2(m-3)mx_1^{(2)} \\
& - 2(m-3)(m-4)x_{m-1}^{(2)} - (m-3)(m-1)x_m^{(2)} - (m-3)(m-1)x_{m+1}^{(2)} \\
& + 2(m-3)(m-3),
\end{aligned}
$$

and therefore

$$
\begin{aligned}
\Delta_3 x^{(3)} - \Delta_3 x^{(2)} = {} & -x_{m-1}^{(2)}x_m^{(2)} + x_{m-1}^{(2)}x_{m+1}^{(2)} - x_1^{(2)}x_m^{(2)} + x_1^{(2)}x_{m+1}^{(2)} \\
& - (m-3)x_m^{(2)} + (m-3)x_{m+1}^{(2)}
\end{aligned}
$$

$$= (x_{m+1}^{(2)} - x_m^{(2)})(x_{m-1}^{(2)} + x_1^{(2)} + m - 3)$$
$$\geq 0.$$

We also see that $\Delta_2 x^{(3)} - \Delta_2 x^{(3)} \geq 0$ because $x^{(3)}$ is symmetric. Since $x^{(2)}$ fulfills (5), $x^{(3)}$ does, too. This also means that the two inequalities from (5) are equivalent.

Finally, we show that $x^{(4)}$ fulfills (5). Let $y := \frac{x_{m-1}^{(3)} - 1}{m-1}$. We see that

$$\forall 1 \leq i \leq d \colon x_i^{(4)} = \begin{cases} x_m^{(3)} + (m-2)y & \text{if } i = m \text{ or } i = m+1, \\ x_1^{(3)} + y & \text{if } i = 1 \text{ or } i = d, \\ 1 & \text{else.} \end{cases}$$

We have

$$\begin{aligned}
\Delta_3 x^{(3)} &= 4x_1^{(3)} x_{m-1}^{(3)} + (2m-2)x_1^{(3)} x_1^{(3)} - (2m-6)x_{m-1}^{(3)} x_{m-1}^{(3)} - (4m-6)x_{m-1}^{(3)} x_m^{(3)} \\
&\quad - 2x_1^{(3)} x_m^{(3)} + 2(m-3)mx_1^{(3)} - 2(m-3)(m-4)x_{m-1}^{(3)} - (m-3)(2m-2)x_m^{(3)} \\
&\quad + 2(m-3)(m-3) \\
&= 4x_1^{(3)} + 4x_1^{(3)}(m-1)y + (2m-2)x_1^{(3)} x_1^{(3)} - (2m-6)(1 + (m-1)y)^2 \\
&\quad - (4m-6)x_m^{(3)} - (4m-6)x_m^{(3)}(m-1)y - 2x_1^{(3)} x_m^{(3)} + 2(m-3)mx_1^{(3)} \\
&\quad - 2(m-3)(m-4) - 2(m-3)(m-4)(m-1)y - (m-3)(2m-2)x_m^{(3)} \\
&\quad + 2(m-3)(m-3) \\
&= 4x_1^{(3)} + (2m-2)x_1^{(3)} x_1^{(3)} - (2m-6) - (4m-6)x_m^{(3)} - 2x_1^{(3)} x_m^{(3)} + 2(m-3)mx_1^{(3)} \\
&\quad - 2(m-3)(m-4) - (m-3)(2m-2)x_m^{(3)} + 2(m-3)(m-3) \\
&\quad + y(m-1)(4x_1^{(3)} - 2(2m-6) - (4m-6)x_m^{(3)} - 2(m-3)(m-4)) \\
&\quad - y^2(m-1)^2(2m-6)
\end{aligned}$$

and

$$\begin{aligned}
\Delta_3 x^{(4)} &= 4x_1^{(3)} + 4y + (2m-2)(x_1^{(3)} + y)^2 - (2m-6) - (4m-6)x_m^{(3)} \\
&\quad - (4m-6)(m-2)y - 2(x_1^{(3)} + y)(x_m^{(3)} + (m-2)y) + 2(m-3)mx_1^{(3)} \\
&\quad + 2(m-3)my - 2(m-3)(m-4) - (m-3)(2m-2)x_m^{(3)} \\
&\quad - (m-3)(2m-2)(m-2)y + 2(m-3)(m-3) \\
&= 4x_1^{(3)} + (2m-2)x_1^{(3)} x_1^{(3)} - (2m-6) - (4m-6)x_m^{(3)} - 2x_1^{(3)} x_m^{(3)} + 2(m-3)mx_1^{(3)} \\
&\quad - 2(m-3)(m-4) - (m-3)(2m-2)x_m^{(3)} + 2(m-3)(m-3) \\
&\quad + y(4 + 2(2m-2)x_1^{(3)} - (4m-6)(m-2) - 2x_1^{(3)}(m-2) - 2x_m^{(3)} \\
&\qquad + 2(m-3)m - (m-3)(2m-2)(m-2)) \\
&\quad + y^2((2m-2) - 2(m-2)) \\
&= 4x_1^{(3)} + (2m-2)x_1^{(3)} x_1^{(3)} - (2m-6) - (4m-6)x_m^{(3)} - 2x_1^{(3)} x_m^{(3)} + 2(m-3)mx_1^{(3)} \\
&\quad - 2(m-3)(m-4) - (m-3)(2m-2)x_m^{(3)} + 2(m-3)(m-3) \\
&\quad + y(2mx_1^{(3)} - 2x_m^{(3)} + 4 - (4m-6)(m-2) + 2(m-3)m \\
&\qquad - (m-3)(2m-2)(m-2)) \\
&\quad + 2y^2,
\end{aligned}$$

and therefore

$$\Delta_3 x^{(4)} - \Delta_3 x^{(3)} = y(-(2m-4)x_1^{(3)} + (2m-4)(2m-1)x_m^{(3)} + 4 - (4m-6)(m-2)$$
$$+ 2(m-3)m - (m-3)(2m-2)(m-2)$$
$$+ 2(2m-6)(m-1) + 2(m-3)(m-4)(m-1))$$
$$+ y^2(2 + (m-1)^2(2m-6))$$
$$= y(2m-4)(-x_1^{(3)} + (2m-1)x_m^{(3)} - (m-2))$$
$$+ y^2(2 + (m-1)^2(2m-6))$$
$$\geq 0.$$

The last step follows from $x_m^{(3)} \geq x_1^{(3)}$, $m \geq 3$ and $y \geq 0$.

Since $x^{(3)}$ and $x^{(4)}$ are symmetric, we also have $\Delta_2(x^{(4)}) - \Delta_2(x^{(3)}) \geq 0$, and with $x^{(3)}$ fulfilling (5), the node distribution $x^{(4)}$ fulfills (5), too. Because of $x^{(4)}$ being a node distribution with only two variables ($x_1^{(4)} = x_d^{(4)}$ and $x_m^{(4)} = x_{m+1}^{(4)}$), we can simplify $\Delta x^{(4)}$ and $\Delta d(x^{(4)})$ to

$$\Delta x^{(4)} := \Delta_2 x^{(4)} = \Delta_3 x^{(4)}$$
$$= 4x_1^{(4)} + (2m-2)x_1^{(4)}x_1^{(4)} - (2m-6) - (4m-6)x_m^{(4)} - 2x_1^{(4)}x_m^{(4)}$$
$$+ 2(m-3)mx_1^{(4)} - 2(m-3)(m-4) - (m-3)(2m-2)x_m^{(4)}$$
$$+ 2(m-3)(m-3)$$
$$= 2(m-1)x_1^{(4)}x_1^{(4)} + 2(m-1)(m-2)x_1^{(4)} - 2x_1^{(4)}x_m^{(4)} - 2m(m-2)x_m^{(4)}$$

and

$$\Delta d(x^{(4)}) = 2(m-1)x_1^{(4)} + (m-2)(m-1) + 2.$$

We now prove the lower bound.

We see that Δd is now only dependent on $x_1^{(4)}$ which means we have to minimize $x_1^{(4)}$. Since $x^{(4)}$ sums up to n, we have $n = 2x_1^{(4)} + 2x_m^{(4)} + 2m - 4$, and therefore $x_m^{(4)} = \frac{n}{2} - x_1^{(4)} - m + 2$. Substituting this into $\Delta x^{(4)}$ yields

$$\Delta x^{(4)} = 2(m-1)x_1^{(4)}x_1^{(4)} + 2(m-1)(m-2)x_1^{(4)} - 2x_1^{(4)}\left(\frac{n}{2} - x_1^{(4)} - m + 2\right)$$
$$- 2m(m-2)\left(\frac{n}{2} - x_1^{(4)} - m + 2\right)$$
$$= 2mx_1^{(4)}x_1^{(4)} + (4m(m-2) - n)x_1^{(4)} - m(m-2)(n - 2m + 4).$$

We know that $x^{(4)}$ fulfills (5) if and only if $\Delta x^{(4)} \geq 0$. Solving the quadratic inequality leaves us with

$$x_1^{(4)} \geq \frac{n}{4m} - (m-2) + \sqrt{\left(\frac{n}{4m} - (m-2)\right)^2 + (m-2)\left(\frac{n}{2} - (m-2)\right)}.$$

Because of (6), we see that the term under the root is larger than

$$0 + (m-2)\left(\frac{3(m+3)}{2} - m + 2\right)$$
$$\geq \frac{1}{2}(m-2)(m-2).$$

We therefore obtain

$$x_1^{(4)} \geq \frac{n}{4m} - (m-2) + \frac{\sqrt{2}}{2}(m-2)$$
$$\geq \frac{n}{4m} - \frac{m-2}{2},$$

and finally

$$\Delta d(x^{(4)}) \geq 2(m-1)\left(\frac{n}{4m} - \frac{m-2}{2}\right) + (m-2)(m-1) + 2 \geq \frac{m-1}{m} \cdot \frac{n}{2} \geq \frac{n}{3},$$

which concludes the proof. ◀

Next, we show that we can find a SMRCST in polynomial time via Algorithm 1 and even guarantee some bounds on the social welfare of the resulting network. Our algorithm employs

■ **Algorithm 1** Computes a SMRCST for a given connected host network.

Input: connected host network H
Output: SMRCST T
1 $\mathsf{P} \leftarrow \texttt{greedyLongPath}\,(\mathsf{H})$;
2 $\mathsf{T} \leftarrow$ *extend* P *to form a spanning tree of* H;
3 while $\exists e \in E_{\mathsf{T}}, e' \in E_{\mathsf{H}} \setminus E_{\mathsf{T}} : d_{\mathsf{T}-e+e'}(V_{\mathsf{H}}, V_{\mathsf{H}}) > d_{\mathsf{T}}(V_{\mathsf{H}}, V_{\mathsf{H}})$ **do**
4 $\quad$ $\mathsf{T} \leftarrow \mathsf{T} - e + e'$
5 end

---

a greedy algorithm developed by Karger et al. [35] which can find a path of length at least $\frac{|E_H|}{|V_H|}$ in $\mathcal{O}(|E_H|)$ as a subroutine for initialization. We call this subroutine *greedyLongPath*. This will help us to derive bounds on the total distances of the computed SMRCST later. For extending the path to a spanning tree in line 2 of Algorithm 1, we can simply iterate over all edges and add them to the network if they do not close a cycle.

▶ **Theorem 11.** *Let H be a connected network containing n nodes and m edges. Then Algorithm 1 finds a Swap-Maximal Routing-Cost Spanning Tree of H in runtime $\mathcal{O}(n^5 m)$.*

Proof. It is easy to see that, by construction, T is always a spanning tree of H. The condition in the while-loop ensures that all possible swaps are tried. This means that the while-loop ends if and only if T is a SMRCST. Therefore if the while-loop stops, the result is correct.

In every iteration in which the while-loop does not stop, the total distances of T increase by at least 1. Since the tree maximizing the total distances is the path, we get its total distances $\frac{1}{3}(n-1)n(n+1) \in \mathcal{O}(n^3)$ as an upper bound for the number of iterations.

The runtime of Algorithm 1 is clearly dominated by the while-loop. Since T has $n-1$ edges which can be removed and $E_H \setminus E_T$ has $\mathcal{O}(m)$ possible edges to add, the number of possible swaps is in $\mathcal{O}(nm)$. For each swap, the total distances can be computed in $\mathcal{O}(n)$ [43]. Therefore computing the condition can be done in $\mathcal{O}(n^2 m)$. Altering the current solution in the body of the while-loop only takes $\mathcal{O}(n)$ when using adjacency lists. Since there are at most $\mathcal{O}(n^3)$ iterations of the while-loop, the overall runtime is in $\mathcal{O}(n^5 m)$. ◀

For the K-SDNCG, the social optima were also stable. For general host networks, this is not necessarily the case. However, we can show that for $\alpha \leq \frac{n}{3}$ there are stable states which approximate the social welfare better than with the trivial factor of $\mathcal{O}(n)$.

▶ **Theorem 12** (OPT-Approximation via the MRCST)**.** *Let H be a connected host network containing n nodes and m edges and T be the MRCST of H.*

(1) *We have $\frac{\text{SW(OPT}_H)}{\text{SW}(T)} \in \mathcal{O}\left(\frac{m}{n}\right)$.*

(2) *For $\alpha \in \mathcal{O}\left(\frac{n^2}{m}\right)$, we have $\frac{\text{SW(OPT}_H)}{\text{SW}(T)} \in \mathcal{O}(1)$.*

(3) *For $\alpha \in \omega\left(\frac{n^2}{m}\right)$, we have $\frac{\text{SW(OPT}_H)}{\text{SW}(T)} \in \mathcal{O}\left(\min\left\{\frac{m}{n}, \alpha\frac{n}{m}\right\} + 1\right)$.*

Proof of (1). Let $V := V_H$ and $O := \text{OPT}_H$. Since T is a MRCST of H, we have $d_T(V,V) \geq d_O(V,V)$. This yields

$$\frac{\text{SW}(O)}{\text{SW}(T)} = \frac{2\alpha|E_O| + d_O(V,V)}{2\alpha|E_T| + d_T(V,V)} \leq \frac{2\alpha m + d_T(V,V)}{2\alpha(n-1) + d_T(V,V)} \leq \frac{m}{n-1} + 1 \in \mathcal{O}\left(\frac{m}{n}\right). \qquad \blacktriangleleft$$

Proof of (2). Let $V := V_H$ and $O := \text{OPT}_H$. We have $2\alpha|E_T| \leq 2\alpha|E_O| \leq 2\alpha m \in \mathcal{O}(n^2)$ and $d_O(V,V) \in \Omega(n^2)$. We also know that $d_O(V,V) \leq d_T(V,V)$ because T is a MRCST. This yields

$$\frac{\text{SW}(O)}{\text{SW}(T)} = \frac{2\alpha|E_O| + d_O(V,V)}{2\alpha|E_T| + d_T(V,V)} \in \mathcal{O}\left(\frac{d_O(V,V)}{d_T(V,V)}\right) = \mathcal{O}(1). \qquad \blacktriangleleft$$

Proof of (3). Let $V := V_H$ and $O := \text{OPT}_H$. Algorithm 1 uses the greedyLongPath algorithm to construct an initial tree that contains a path of at least length $l \geq \frac{m}{n}$ [35]. Every node in that tree has a distance of at least $\frac{l}{3}$ to at least $\frac{l}{3}$ of the nodes on the long path. This means that the total distances are at least $n\left(\frac{l}{3}\right)^2 \in \Omega\left(\frac{m^2}{n}\right)$. Since Algorithm 1 only increases the total distances, this is a lower bound for every solution found by the algorithm, and therefore also for the MRCST. With this, we get

$$\frac{\text{SW}(O)}{\text{SW}(T)} = \frac{2\alpha|E_O| + d_O(V,V)}{2\alpha|E_T| + d_T(V,V)} \leq \frac{2\alpha m}{2\alpha(n-1) + d_T(V,V)} + \frac{d_T(V,V)}{2\alpha(n-1) + d_T(V,V)}$$

$$\in \mathcal{O}\left(\frac{\alpha m}{\alpha n + \frac{m^2}{n}} + 1\right) = \mathcal{O}\left(\min\left\{\frac{m}{n}, \alpha\frac{n}{m}\right\} + 1\right). \qquad \blacktriangleleft$$

Since finding the MRCST is NP-hard [16], these are only existence results. However, the next theorem yields a bound for dense networks and the computed SMRCST from Algorithm 1.

▶ **Theorem 13.** *Let H be a connected host network containing n nodes and m edges and T be the SMRCST obtained by Algorithm 1. Then for $\alpha \in \mathcal{O}(n)$, we have $\frac{\text{SW(OPT}_H)}{\text{SW}(T)} \in \mathcal{O}\left(\frac{n^4}{m^2}\right)$.*

Proof. Let $V := V_H$ and $O := \text{OPT}_H$. We can trivially upper bound the distances and number of edges with $d_O(V,V) \in \mathcal{O}(n^3)$ and $|E_O| \leq m$. Since T is a result of Algorithm 1, we have $d_T(V,V) \in \Omega\left(\frac{m^2}{n}\right)$, as seen in the previous proof. This yields

$$\frac{\text{SW}(O)}{\text{SW}(T)} = \frac{2\alpha|E_O| + d_O(V,V)}{2\alpha|E_T| + d_T(V,V)} \leq \frac{2\alpha m}{2\alpha(n-1) + \frac{m^2}{n}} + \frac{n^3}{2\alpha(n-1) + \frac{m^2}{n}}$$

$$\in \mathcal{O}\left(\frac{n^2}{m} + \frac{n^4}{m^2}\right) = \mathcal{O}\left(\frac{n^4}{m^2}\right). \qquad \blacktriangleleft$$

This means, for $\alpha \leq \frac{n}{3}$ and a dense host network, we can compute a state which is pairwise stable and also has a favorable social welfare.
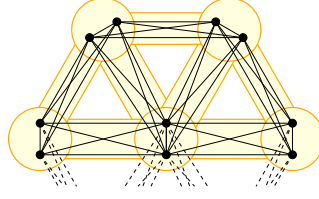
Figure 6 This figure shows a clique network (black) for a wheel network (yellow).

3.2 Price of Anarchy and Price of Stability

We derive several bounds on the PoA and the PoS for the SDNCG. For the K-SDNCG, the PoA is already quite high for small α. The next Theorem shows that this gets even worse for general host networks since the PoA is linear up to $\alpha \le n$ and super-constant for $\alpha \in o(n^2)$.

▶ **Theorem 14** (Price of Anarchy).
(1) *The Price of Anarchy is in $\mathcal{O}(n)$.*
(2) *For $\alpha < 1$, the Price of Anarchy is in $\Theta(n)$.*
(3) *For $1 \le \alpha \le n$, the Price of Anarchy is in $\Theta(n)$.*
(4) *For $n < \alpha \le n^2$, the Price of Anarchy is in $\Omega\left(\frac{n^2}{\alpha}\right)$.*
(5) *For $\frac{1}{4}(n-1)^2 < \alpha \le \frac{1}{24}(n-2)n(n+2)$, the Price of Anarchy is in $\Theta(1)$.*
(6) *For $\alpha > \frac{1}{24}(n-2)n(n+2)$, the Price of Anarchy is 1.*

Proof of (1) and (2). This follows in the same way as (1) and (2) of Theorem 5.   ◀

Proof of (3) and (4). Let $W = (V_W, E_W)$ be a wheel network on $n' \coloneqq \lfloor \frac{n}{2} \rfloor$ nodes, i.e.,

$$V_W \coloneqq \{v_1, \ldots, v_{n'}\} \quad \text{and}$$
$$E_W \coloneqq \{\{v_1, v_i\} \mid 2 \le i \le n'\} \cup \{\{v_i, v_{i+1}\} \mid 2 \le i \le n'\} \cup \{\{v_2, v_{n'}\}\}.$$

We then define the host network H as the clique network obtained by replacing every node of W by a clique of size 2. (See Figure 6 for an illustration.) For odd n, we instead replace the central node by a clique of size 3. We see that H contains n nodes, $\Theta(n)$ edges, and most importantly a Hamilton path. We also know that H is stable because of Theorem 3 and since no edge can be added. This yields the following lower bound for the Price of Anarchy

$$\text{PoA} \ge \frac{SW(P)}{SW(H)} = \frac{\alpha(n-1) + \Theta(n^3)}{\alpha\Theta(n) + \Theta(n^2)} \in \Omega\left(\min\left\{n, \frac{n^2}{\alpha}\right\}\right),$$

which proves the claim.   ◀

Proof of (5). Let H be a (connected) host network and $O \coloneqq \text{OPT}_H$ and $V \coloneqq V_H$. From Item 2 we know that H itself is the only pairwise stable network. Since we can bound $|E_O| \le |E_H|$ and $d_O(V, V) \le n^3$, we obtain

$$\text{PoA}_n \le \frac{SW(O)}{SW(H)} = \frac{2\alpha|E_O| + d_O(V,V)}{2\alpha|E_H| + d_H(V,V)} \le \frac{2\alpha|E_H| + n^3}{2\alpha|E_H|} \in \Theta\left(1 + \frac{n^3}{\alpha n}\right) = \Theta(1).$$

◀

Proof of (6). This follows directly from the host network being socially optimal and the only stable network (see Theorem 7 and Theorem 9).   ◀

▶ **Theorem 15** (Price of Stability).
(1) *The Price of Stability is in $\mathcal{O}(n)$.*
(2) *For $\alpha \leq 1$, the Price of Stability is 1.*
(3) *For $1 < \alpha \leq \frac{n}{3}$, the Price of Stability is in $\mathcal{O}(\sqrt{n})$.*
(4) *For $\frac{1}{4}(n-1)^2 < \alpha \leq \frac{1}{24}(n-2)n(n+2)$, the Price of Stability is in $\Theta(1)$.*
(5) *For $\alpha > \frac{1}{24}(n-2)n(n+2)$, the Price of Stability is 1.*

Proof of (1), (4), and (5). This follows from Theorem 14 and the Price of Anarchy being an upper bound for the Price of Stability. ◄

Proof of (2). This follows directly from the MRCST being socially optimal and stable (see Theorem 7 and Theorem 9). ◄

Proof of (3). Let H be a (connected) host network and T be the MRCST of H. Let furthermore $O \coloneqq \mathrm{OPT}_H$ and $V \coloneqq V_H$. From Theorem 10 we know that T is stable. From Theorem 7 and Theorem 12, we know that

$$\frac{\mathrm{SW}(O)}{\mathrm{SW}(T)} \in \mathcal{O}\left(\min\left\{\frac{m}{n}, \alpha \frac{n}{m}\right\}\right) \leq \mathcal{O}\left(\min\left\{\frac{m}{n}, \frac{n^2}{m}\right\}\right) \leq \mathcal{O}\left(\sqrt{n}\right). \qquad ◄$$

4 Conclusion

We introduced and analyzed a natural game-theoretic model for network formation governed by social distancing. Besides modeling this timely issue, our model resembles the inverse compared to the well-known (bilateral) Network Creation Game [23, 18]. Thus, via our analysis we could explore the impact of inverting the utility function in a non-trivial strategic game. We find that this inverts some of the properties, like the rough structure of optimum states, while it also yields non-obvious insights. First of all, for the variant with non-complete host networks we could show a strong equilibrium existence result, whereas no such result is known for the inverse model. Moreover, we established that the PoA is significantly higher in the $(K\text{-})$SDCNG compared to the (bilateral) NCG. This demonstrates that the impact of the agents' selfishness is higher under social distancing, which calls for external coordination.

The most obvious open question for future work is to settle the equilibrium existence. Do pairwise stable states exist for all connected host networks H and α? Another research direction would be to consider the unilateral variant of the SDNCG. While this no longer realistically models the formation of social networks, it might still yield interesting insights and it allows for studying stronger solution concepts like the Nash equilibrium or strong Nash equilibria, similar to [33, 6]. Also, altering the utility function, e.g., to using the maximum distance instead of the summed distances, or the probability of infection, similar to [13], seems promising. Finally, also considering weighted host networks, as in [11], where the edge weight models the benefit of the social interaction, would be an interesting generalization.

─── **References** ───

1 Daron Acemoglu, Asuman Ozdaglar, and Alireza Tahbaz-Salehi. Systemic risk and stability in financial networks. *American Economic Review*, 105(2):564–608, 2015. `doi:10.1257/aer.20130456`.
2 Susanne Albers, Stefan Eilts, Eyal Even-Dar, Yishay Mansour, and Liam Roditty. On nash equilibria for a network creation game. *ACM TEAC*, 2(1):1–27, 2014. `doi:10.1145/2560767`.
3 Franklin Allen and Douglas Gale. Financial contagion. *Journal of Political Economy*, 108(1):1–33, 2000. `doi:10.1086/262109`.

4   Carme Àlvarez and Arnau Messegué. Network creation games: Structure vs anarchy. *CoRR*, abs/1706.09132, 2017. `doi:10.48550/arXiv.1706.09132`.

5   Carme Àlvarez and Arnau Messegué. On the price of anarchy for high-price links. In *WINE 2019*, pages 316–329, 2019. `doi:10.1007/978-3-030-35389-6\_23`.

6   Nir Andelman, Michal Feldman, and Yishay Mansour. Strong price of anarchy. *Games and Economic Behavior*, 65(2):289–317, 2009. `doi:10.1016/j.geb.2008.03.005`.

7   Elliot Anshelevich, Anirban Dasgupta, Jon M. Kleinberg, Éva Tardos, Tom Wexler, and Tim Roughgarden. The price of stability for network design with fair cost allocation. *SIAM Journal on Computing*, 38(4):1602–1623, 2008. `doi:10.1137/070680096`.

8   Elliot Anshelevich, Anirban Dasgupta, Éva Tardos, and Tom Wexler. Near-optimal network design with selfish agents. *Theory of Computing*, 4(1):77–109, 2008. `doi:10.4086/toc.2008.v004a004`.

9   Venkatesh Bala and Sanjeev Goyal. A noncooperative model of network formation. *Econometrica*, 68(5):1181–1229, 2000. `doi:10.1111/1468-0262.00155`.

10   Davide Bilò, Tobias Friedrich, Pascal Lenzner, Stefanie Lowski, and Anna Melnichenko. Selfish creation of social networks. In *AAAI 2021*, pages 5185–5193, 2021. `doi:10.1145/3323165.3323199`.

11   Davide Bilò, Tobias Friedrich, Pascal Lenzner, and Anna Melnichenko. Geometric network creation games. In *SPAA 2019*, pages 323–332, 2019. `doi:10.1145/3323165.3323199`.

12   Davide Bilò and Pascal Lenzner. On the tree conjecture for the network creation game. *Theory of Computing Systems*, 64(3):422–443, 2020. `doi:10.1007/s00224-019-09945-9`.

13   Larry Blume, David A. Easley, Jon M. Kleinberg, Robert D. Kleinberg, and Éva Tardos. Network formation in the presence of contagious risk. In *EC 2011*, pages 1–10, 2011. `doi:10.1145/1993574.1993576`.

14   Prosenjit Bose and Michiel Smid. On plane geometric spanners: A survey and open problems. *Computational Geometry*, 46(7):818–830, 2013. `doi:10.1016/j.comgeo.2013.04.002`.

15   Ricardo Caballero and Alp Simsek. Fire sales in a model of complexity. *The Journal of Finance*, 68, 2009. `doi:10.2139/ssrn.1496592`.

16   P.M. Camerini, G. Galbiati, and F. Maffioli. On the complexity of finding multi-constrained spanning trees. *Discrete Applied Mathematics*, 5(1):39–50, 1983. `doi:10.1016/0166-218X(83)90014-8`.

17   Yu Chen, Shahin Jabbari, Michael J. Kearns, Sanjeev Khanna, and Jamie Morgenstern. Network formation under random attack and probabilistic spread. In *IJCAI 2019*, pages 180–186, 2019. `doi:10.24963/ijcai.2019/26`.

18   Jacomo Corbo and David C. Parkes. The price of selfish behavior in bilateral network formation. In *PODC 2005*, pages 99–107, 2005. `doi:10.1145/1073814.1073833`.

19   Erik D. Demaine, Mohammad Taghi Hajiaghayi, Hamid Mahini, and Morteza Zadimoghaddam. The price of anarchy in cooperative network creation games. *SIGecom Exchanges*, 8(2):2:1–2:20, 2009. `doi:10.1145/1980522.1980524`.

20   Erik D. Demaine, Mohammad Taghi Hajiaghayi, Hamid Mahini, and Morteza Zadimoghaddam. The price of anarchy in network creation games. *ACM Transactions on Algorithms*, 8(2):13, 2012. `doi:10.1145/2151171.2151176`.

21   Jack Dippel and Adrian Vetta. An improved bound for the tree conjecture in network creation games. *CoRR*, abs/2106.05175, 2021. `doi:10.48550/arXiv.2106.05175`.

22   Andrey A. Dobrynin, Roger C. Entringer, and Ivan Gutman. Wiener index of trees: Theory and applications. *Acta Applicandae Mathematica*, 66:211–249, 2001. `doi:10.1023/A:1010767517079`.

23   Alex Fabrikant, Ankur Luthra, Elitza N. Maneva, Christos H. Papadimitriou, and Scott Shenker. On a network creation game. In *PODC 2003*, pages 347–351, 2003. `doi:10.1145/872035.872088`.

24   Wilhelm Friedemann, Tobias Friedrich, Hans Gawendowicz, Pascal Lenzner, Anna Melnichenko, Jannik Peters, Daniel Stephan, and Michael Vaichenker. Efficiency and stability in euclidean network design. In *SPAA 2021*, pages 232–242, 2021. `doi:10.1145/3409964.3461807`.

25   Tobias Friedrich, Sven Ihde, Christoph Keßler, Pascal Lenzner, Stefan Neubert, and David Schumann. Efficient best response computation for strategic network formation under attack. In *SAGT 2017*, pages 199–211, 2017. `doi:10.1007/978-3-319-66700-3\_16`.

26   Giulia Galbiati, Angelo Morzenti, and Francesco Maffioli. On the approximability of some maximum spanning tree problems. *Theoretical Computer Science*, 181(1):107–118, 1997.

27   Sanjeev Goyal, Shahin Jabbari, Michael J. Kearns, Sanjeev Khanna, and Jamie Morgenstern. Strategic network formation with attack and immunization. In *WINE 2016*, pages 429–443, 2016. `doi:10.1007/978-3-662-54110-4\_30`.

28   Ronald L. Graham and Pavol Hell. On the history of the minimum spanning tree problem. *IEEE Annals of the History of Computing*, 7(1):43–57, 1985. `doi:10.1109/MAHC.1985.10011`.

29   Andrew G Haldane and Robert M May. Systemic risk in banking ecosystems. *Nature*, 469(7330):351–355, 2011. `doi:10.1038/nature09659`.

30   T. C. Hu. Optimum communication spanning trees. *SIAM Journal on Computing*, 3(3):188–195, 1974. `doi:10.1137/0203015`.

31   Matthew O Jackson. *Social and economic networks*. Princeton university Press, 2008. `doi:10.2307/j.ctvcm4gh1`.

32   Matthew O. Jackson and Asher Wolinsky. A strategic model of social and economic networks. *Journal of Economic Theory*, 71(1):44–74, 1996. `doi:10.1006/jeth.1996.0108`.

33   Tomasz Janus and Bart de Keijzer. On strong equilibria and improvement dynamics in network creation games. In *WINE 2017*, pages 161–176, 2017. `doi:10.1007/978-3-319-71924-5\_12`.

34   David S. Johnson, Jan Karel Lenstra, and A. H. G. Rinnooy Kan. The complexity of the network design problem. *Networks*, 8(4):279–285, 1978. `doi:10.1002/net.3230080402`.

35   David R. Karger, Rajeev Motwani, and G. D. S. Ramkumar. On approximating the longest path in a graph. *Algorithmica*, 18(1):82–98, 1997. `doi:10.1007/BF02523689`.

36   Bernd Kawald and Pascal Lenzner. On dynamics in selfish network creation. In *SPAA 2013*, pages 83–92, 2013. `doi:10.1145/2486159.2486185`.

37   Elias Koutsoupias and Christos H. Papadimitriou. Worst-case equilibria. In *STACS 1999*, pages 404–413, 1999. `doi:10.1007/3-540-49116-3\_38`.

38   Pascal Lenzner. On dynamics in basic network creation games. In *SAGT 2011*, pages 254–265, 2011. `doi:10.1007/978-3-642-24829-0\_23`.

39   Pascal Lenzner. Greedy selfish network creation. In *WINE 2012*, pages 142–155, 2012. `doi:10.1007/978-3-642-35311-6\_11`.

40   Thomas L Magnanti and Richard T Wong. Network design and transportation planning: Models and algorithms. *Transportation science*, 18(1):1–55, 1984. `doi:10.1287/trsc.18.1.1`.

41   Akaki Mamageishvili, Matús Mihalák, and Dominik Müller. Tree nash equilibria in the network creation game. *Internet Mathematics*, 11(4-5):472–486, 2015. `doi:10.1080/15427951.2015.1016248`.

42   Matúš Mihalák and Jan Christoph Schlegel. The price of anarchy in network creation games is (mostly) constant. *TCS*, 53(1):53–72, 2013. `doi:10.1007/s00224-013-9459-y`.

43   Bojan Mohar and Tomaž Pisanski. How to compute the wiener index of a graph. *Journal of Mathematical Chemistry*, 2(3):267–277, 1988. `doi:10.1007/BF01167206`.

44   Dov Monderer and Lloyd S. Shapley. Potential games. *Games and Economic Behavior*, 14(1):124–143, 1996. `doi:10.1006/game.1996.0044`.

45   Giri Narasimhan and Michiel Smid. *Geometric spanner networks*. Cambridge University Press, 2007. `doi:10.1017/CBO9780511546884`.

46   Tim Roughgarden and Éva Tardos. How bad is selfish routing? *Journal of the ACM (JACM)*, 49(2):236–259, 2002. `doi:10.1145/506147.506153`.

47   Hans Wiener. Structural determination of paraffin boiling points. *Journal of the American Chemical Society*, 69 1:17–20, 1947. `doi:10.1021/ja01193a005`.

48 Kexiang Xu, Muhuo Liu, Kinkar Das, Ivan Gutman, and Boris Furtula. A survey on graphs extremal with respect to distance–based topological indices. *Match (Mulheim an der Ruhr, Germany)*, 71:461–508, 2014.

49 Ľubomír Šoltés. Transmission in graphs: A bound and vertex removing. *Mathematica Slovaca*, 41(1):11–16, 1991. URL: `http://hdl.handle.net/10338.dmlcz/132387`.