Theoretical Computer Science 971 (2023) 114065

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Not all strangers are the same: The impact of tolerance in Schelling games ☆,☆☆

Panagiotis Kanellopoulos *, Maria Kyropoulou, Alexandros A. Voudouris

School of Computer Science and Electronic Engineering, University of Essex, UK

A R T I C L E I N F O

Article history:
Received 20 December 2022
Received in revised form 3 April 2023
Accepted 7 July 2023
Available online 13 July 2023
Communicated by M. Mavronicolas

Keywords:
Schelling's segregation model
Nash equilibria
Price of anarchy

A B S T R A C T

Schelling's famous model of segregation assumes agents of different types, who would like to be located in neighborhoods having at least a certain fraction of agents of the same type. We consider natural generalizations that allow for the possibility of agents being tolerant towards other agents, even if they are not of the same type. In particular, we consider an ordering of the types, and make the realistic assumption that the agents are in principle more tolerant towards agents of types that are closer to their own according to the ordering. Based on this, we study the strategic games induced when the agents aim to maximize their utility for a variety of tolerance levels. We provide a collection of results about the existence of equilibria, and their quality in terms of social welfare.

1. Introduction

Residential segregation is a broad phenomenon affecting most metropolitan areas, and is known to be caused due to racial or socio-economic differences. The severity of its implications to society [6] is the main reason for the vast research attention it has received, with many different models being proposed over the years that aim to conceptualize it (e.g., see [27]). The most prominent of those models is that of Schelling [25,26], which studies how motives at an individual level can lead to macroscopic behavior and, ultimately, to segregation. In particular, the individuals are modeled as agents of two different types (usually referred to using colors, such as red and blue), and the environment is abstracted by a topology (such as a grid graph), representing a city. The agents occupy nodes of the topology and prefer neighborhoods in which the presence of their own type exceeds a specified tolerance threshold. If an agent is unhappy with her current location, then she either jumps to a randomly selected empty node of the topology, or swaps positions with another random unhappy agent. Schelling's crucial observation was that such dynamics might lead to largely segregated placements, even when the agents are relatively tolerant of mixed neighborhoods.

A recent series of papers (discussed in Section 1.2) have generalized Schelling's model to include more than two types, and have taken a game-theoretic approach, according to which the agents behave strategically rather than randomly, aiming to maximize their individual *utility*. There are many ways to define the utility of an agent i of type T. For instance, Elkind

☆ This article belongs to Section A: Algorithms, automata, complexity and games, Edited by Paul Spirakis.
☆☆ A preliminary version of this paper has appeared in *Proceedings of the 47th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, pages 60:1–60:14, 2022.

* Corresponding author.
E-mail addresses: panagiotis.kanellopoulos@essex.ac.uk (P. Kanellopoulos), maria.kyropoulou@essex.ac.uk (M. Kyropoulou), alexandros.voudouris@essex.ac.uk (A.A. Voudouris).

https://doi.org/10.1016/j.tcs.2023.114065

et al. [18] defined it as the ratio of the number of agents of type T in i's neighborhood over the *total* number of agents therein. Echzell et al. [17] proposed a similar definition, which however does not take into account all the agents of different type in the denominator, but only those of the *majority* type. The first definition essentially assumes that the agents view all the agents of different type as *enemies*. On the other hand, the second definition assumes that the agents view only the majority type as hostile. An alternative way of thinking about these particular utility functions is as if the agents have binary *tolerance* towards other agents in the sense that agents are either friends or enemies; in the case of Elkind et al. all the neighbors of an agent are taken into account when computing her utility, whereas in the case of Echzell et al. some of her neighbors are ignored.

These functions are natural generalizations of the quantity that determines the happiness of agents in Schelling's original model for two types. However, they fail to capture realistic scenarios in which the agents do not have a single-dimensional view of the other agents, but rather have different *preferences* over the different types of agents. For example, suppose that the agents correspond to voters while the types correspond to political parties. In this case, the preferences of voters over other voters are defined based on the distances of the political views expressed by the parties they are affiliated with. Another example is when the types correspond to research areas, in which case people working on a specific research agenda will be more willing to collaborate with other people working on related problems.

1.1. Our contribution

To capture scenarios like the examples above, we propose a clean model that naturally extends the model of Elkind et al. [18] by incorporating *different levels of tolerance* among agent types, and study the induced strategic games in terms of the existence and quality of their equilibria; in Section 5, we discuss potential generalizations of our model.

To be more specific, our model consists of a set of agents who are partitioned into $\lambda \geq 2$ types of equal size,[1] a graph topology, and an ordering of the different agent types which determines the relative tolerance among agents of different types. Naturally, we assume that there is higher tolerance between agents whose types are closer according to the ordering. The exact degree of tolerance between the different types is specified by a *tolerance vector*, which consists of weights representing the tolerance between the different types depending on their distance in the given ordering. For example, agents of the same type are in distance 0 and are fully tolerant towards each other, which is captured by a weight of 1. The utility of an agent can then be computed as a weighted average of the tolerance that she has towards her neighbors, and every agent aims to occupy a node of the topology to maximize her utility; agents are allowed to unilaterally *jump* to empty nodes to increase their utility.

We study the dynamics of such *tolerance Schelling games*. We first focus on questions related to equilibrium existence. For general games, we show that equilibria are not guaranteed to exist if agents are not fully tolerant towards agents in type-distance 1 (Theorem 2). We complement this impossibility by showing many positive results for important subclasses of games, in which the topology is a structured graph (such a 4-grid or a tree) and the tolerance vector satisfies certain properties (Theorems 3, 5, 6 and 7). We then turn our attention to the quality of equilibria measured by the social welfare objective, defined as the total utility of the agents, and prove (asymptotically tight) bounds on the price of anarchy [22] (Theorems 8 and 9) and price of stability [3] (Theorem 14), which depend on the number of types, the number of agents and/or the tolerance parameters.

1.2. Related work

Residential segregation, and Schelling's original randomized model in particular, has been the basis of a continuous stream of multidisciplinary research in Sociology [15], Economics [24,29], Physics [28], and Computer Science [5,7,10,11,20].

Most related to our work is a quite recent series of papers in the TCS and AI communities, which deviated from the premise of random behavior, and instead studied the strategic games induced when the agents act as utility-maximizers. Chauhan et al. [14] studied questions related to dynamics convergence in games with two types of agents who can either jump to empty nodes of a topology (as in our case) or *swap* locations with other agents to minimize a *cost* function; their model was generalized to multiple types of agents by Echzell et al. [17]. In this paper we extend the utility model of Elkind et al. [18], who refined the cost model of Chauhan et al. [14]. They introduced a simpler utility function (fraction of same-type agents in one's neighborhood) which the agents aim to maximize, and studied the existence, complexity and quality of equilibria in jump games with multiple types of agents and general topologies. Among other results, we recover the upper bound of $\frac{\lambda n}{n-\lambda}$ on the price of anarchy of balanced jump Schelling games shown by Elkind et al. [18] and prove that it is actually tight, by improving the lower bound. We also improve their lower bound of 34/33 on the price of stability to approximately 2; see the discussion in Section 4 for more details.

The existence of equilibria in the model of Elkind et al. [18] was further studied by Kreisel et al. [23]. Elkind et al. [18] also proposed many interesting variants, such as *enemy aversion* (agents might prefer being alone to being in a group full

[1] We focus on such *balanced games* as this class of games is arguably the most fundamental one and provides us with a structure that we can exploit to show positive results in terms of the existence and the quality equilibria. In general, equilibria are not guaranteed to exist even for balanced games (see Theorem 2), and so it makes sense to pinpoint special cases for which existence can be shown, as we do in Section 3. In addition, it not hard to observe that the price of anarchy can be unbounded when the sizes of the types are arbitrary [18].

of agents of different type than their own) and *social Schelling games* (where the agents types are determined by a social network), which have been partially studied by Kanellopoulos et al. [21] and Chan et al. [13], respectively. Agarwal et al. [1] studied existence, complexity and qualitative questions for swap games; the results of Elkind et al. [18] for jump games and Agarwal et al. [1] for swap games were combined into a journal version [2]. Bilò et al. [9] also focused on swap games, and in particular, on a constrained setting, where the agents can only view a small part of the topology near their current location and can only swap with agents in this part of the topology. Recently, Bilò et al. [8] considered swap games when the agents aim to optimize general single-peaked utility functions of the fraction of same-type agents, showing that some tolerance is needed for the existence of equilibria. The jump version of such single-peaked games has been studied by Friedrich et al. [19]. Finally, Bullinger et al. [12] and Deligkas et al. [16] studied the (parameterized) complexity of computing assignments with good welfare guarantees, focusing on the social welfare, Nash welfare, and Pareto optimality, and many other welfare objectives.

2. Preliminaries

A λ-*type tolerance Schelling game* consists of:

- A set N of $n \geq 4$ *agents*, partitioned into $\lambda \geq 2$ disjoint sets $T_1, \ldots, T_\lambda$ representing *types*, such that $\bigcup_{\ell \in [\lambda]} T_\ell = N$.
- A simple connected undirected graph $G = (V, E)$ called *topology*, such that $|V| > n$.
- A *tolerance vector* $\mathbf{t}_\lambda = [t_0, \ldots, t_{\lambda-1}]$ consisting of λ parameters, such that t_d represents the tolerance that agents of type T_ℓ have towards agents of type T_k in Manhattan distance $|\ell - k| = d \in \{0, \ldots, \lambda - 1\}$ according to a given ordering $\succ$ of the types (say, $T_1 \succ \ldots \succ T_\lambda$). We assume that agents are more tolerant towards agents of types that are closer to their own according to $\succ$, and we thus have that $1 = t_0 \geq \ldots \geq t_{\lambda-1} \geq 0$. We also assume that $t_{\lambda-1} < 1$; otherwise, all agents are completely tolerant towards all others and the game is trivial. Let $\tau = \sum_{d=0}^{\lambda-1} t_d$ be the sum of all tolerance parameters.

Clearly, the class of λ-type tolerance Schelling games includes as a special case the classic *jump Schelling games* studied in the related literature (e.g., see [18]), for which $t_0 = 1$ and $t_d = 0$ for every $d \in \{1, \ldots, \lambda - 1\}$. Because of this particular tolerance vector, we will use the term λ-*type zero-tolerance games* to refer to the classic Schelling games.

In this paper we consider *balanced* games, in which the agents are partitioned in types of equal size, such that $|T_\ell| = n/\lambda \geq 2$ for every $\ell \in [\lambda]$; thus, n is a multiple of λ. We use the abbreviation λ-TS to refer to such a balanced λ-type tolerance Schelling game $\mathcal{I} = (N, G, \mathbf{t}_\lambda)$. For convenience, we will also use the abbreviation λ-ZTS to refer to a balanced λ-type zero-tolerance game $\mathcal{I} = (N, G)$.

Let $\mathbf{v} = (v_i)_{i \in N}$ be an *assignment* specifying the node v_i of G that each agent $i \in N$ *occupies*, such that $v_i \neq v_j$ for $i \neq j$. The *neighborhood* of a node v consists of the nodes at distance 1 from v in G. For every node v, we denote by $n_\ell(v|\mathbf{v})$ the number of agents of type T_ℓ that occupy nodes in the neighborhood of v according to the assignment $\mathbf{v}$, and also let $n(v|\mathbf{v}) = \sum_{\ell \in [\lambda]} n_\ell(v|\mathbf{v})$. Given an assignment $\mathbf{v}$, the utility of agent i of type T_ℓ is computed as

$$u_i(\mathbf{v}) = \frac{1}{n(v_i|\mathbf{v})} \sum_{k \in [\lambda]} t_{|\ell-k|} \cdot n_k(v_i|\mathbf{v}),$$

if $n(v_i|\mathbf{v}) \neq 0$, and 0 otherwise (in which case we say that the agent is *isolated*). The agents are *strategic* and aim to maximize their utility by *jumping* to empty nodes of the topology if they can increase their utility by doing so. We say that an assignment $\mathbf{v}$ is an *equilibrium* if no agent i of any type T_ℓ has incentive to jump to any empty node v of the topology, that is, $u_i(\mathbf{v}) \geq u_i(v, \mathbf{v}_{-i})$, where $(v, \mathbf{v}_{-i})$ is the assignment resulting from this jump. Let $\text{EQ}(\mathcal{I})$ be the set of equilibrium assignments of a given λ-TS game $\mathcal{I}$.

The *social welfare* of an assignment $\mathbf{v}$ is defined as the total utility of the agents, that is,

$$\text{SW}(\mathbf{v}) = \sum_{i \in N} u_i(\mathbf{v}).$$

Let $\text{OPT}(\mathcal{I}) = \max_{\mathbf{v}} \text{SW}(\mathbf{v})$ be the maximum social welfare among all possible assignments in the λ-TS game $\mathcal{I}$. For a given subclass $\mathcal{C}$ of λ-TS games, the *price of anarchy* is defined as the worst-case ratio, over all possible games $\mathcal{I} \in \mathcal{C}$ such that $\text{EQ}(\mathcal{I}) \neq \varnothing$, between $\text{OPT}(\mathcal{I})$ and the *minimum* social welfare among all equilibria:

$$\text{PoA}(\mathcal{C}) = \sup_{\mathcal{I} \in \mathcal{C}: \text{EQ}(\mathcal{I}) \neq \varnothing} \frac{\text{OPT}(\mathcal{I})}{\min_{\mathbf{v} \in \text{EQ}(\mathcal{I})} \text{SW}(\mathbf{v})}.$$

Similarly, the *price of stability* takes into account the ratio between $\text{OPT}(\mathcal{I})$ and the *maximum* social welfare among all equilibria:

$$\text{PoS}(\mathcal{C}) = \sup_{\mathcal{I} \in \mathcal{C}: \text{EQ}(\mathcal{I}) \neq \varnothing} \frac{\text{OPT}(\mathcal{I})}{\max_{\mathbf{v} \in \text{EQ}(\mathcal{I})} \text{SW}(\mathbf{v})}.$$

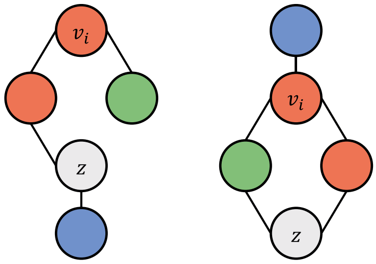

Fig. 1. The partial assignments $\mathbf{v}$ (left) and $\mathbf{v}'$ (right) used in the proof of Theorem 1. Here, agents of type T_1 are colored red, agents of type T_2 are colored blue, and agents of type T_3 are colored green. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

3. Equilibrium existence

In this section, we show several positive and negative results about the existence of equilibrium assignments for interesting subclasses of tolerance Schelling games. We start with the relation of equilibrium assignments in λ-ZTS games and general λ-TS games.

Theorem 1. *Consider a λ-ZTS game $\mathcal{I} = (N, G)$ and a λ-TS game $\mathcal{I}' = (N, G, \mathbf{t}_\lambda)$. For $\lambda = 2$, EQ($\mathcal{I}'$) $\subseteq$ EQ($\mathcal{I}$) and EQ($\mathcal{I}$) $\setminus$ EQ($\mathcal{I}'$) consists of assignments with isolated agents. For $\lambda \geq 3$, EQ($\mathcal{I}$) and EQ($\mathcal{I}'$) are incomparable.*

Proof. We start with $\lambda = 2$; for convenience, we will refer to the two types as red and blue. Let $\mathbf{v}$ be an equilibrium of $\mathcal{I}'$. Clearly, for $\mathcal{I}$ and $\mathcal{I}'$ to be different, it must be the case that $t_1 > 0$. Consequently, there are no isolated agents in $\mathbf{v}$ as they would have incentive to deviate to nodes that are adjacent to any other agent and increase their utility from 0 to (at least) t_1. We will show that $\mathbf{v}$ is an equilibrium of $\mathcal{I}$ as well. Without loss of generality, consider a red agent i who occupies a node v_i that is adjacent to $n_r(v_i)$ red and $n_b(v_i)$ blue agents. Since agent i is not isolated, it holds that $n_r(v_i) + n_b(v_i) \geq 1$. If $n_b(v_i) = 0$, then agent i has maximum utility 1 in both $\mathcal{I}$ and $\mathcal{I}'$. Hence, we can assume that $n_b(v_i) \geq 1$. Since $\mathbf{v}$ is an equilibrium of $\mathcal{I}'$, agent i has no incentive to unilaterally jump to any empty node z of the topology. That is,

$$\frac{n_r(v_i) + t_1 \cdot n_b(v_i)}{n_r(v_i) + n_b(v_i)} \geq \frac{n_r(z) + t_1 \cdot n_b(z)}{n_r(z) + n_b(z)} \Leftrightarrow (1 - t_1)\left(\frac{n_r(v_i)}{n_b(v_i)} - \frac{n_r(v)}{n_b(v)}\right) \geq 0,$$

where $n_r(z)$ and $n_b(z)$ are the number of red and blue agents that are adjacent to z *after* agent i jumps to z; observe that $n_b(z) \geq 1$, as otherwise agent i would obtain maximum utility of 1 by jumping to z, contradicting that $\mathbf{v}$ is an equilibrium of $\mathcal{I}'$. Since $t_1 < 1$, we equivalently have that

$$\frac{n_r(v_i)}{n_b(v_i)} \geq \frac{n_r(z)}{n_b(z)} \Leftrightarrow \frac{n_r(v_i)}{n_r(v_i) + n_b(v_i)} \geq \frac{n_r(z)}{n_r(z) + n_b(z)}.$$

Therefore, agent i has no incentive to jump to the empty node z in $\mathcal{I}$, and $\mathbf{v}$ is an equilibrium of $\mathcal{I}$ as well. Using similar arguments, we can show that any equilibrium of $\mathcal{I}$ such that there is no isolated agent is also an equilibrium of $\mathcal{I}'$.

For $\lambda \geq 3$, to show that EQ($\mathcal{I}$) is incomparable to EQ($\mathcal{I}'$), consider the tolerance vector $\mathbf{t_3} = (1, 1/2, 0)$ and the following two partial assignments $\mathbf{v}$ and $\mathbf{v}'$ (see also Fig. 1):

- In $\mathbf{v}$, a red agent i of type T_1 occupies a node v_i which is adjacent to two nodes, one occupied by an agent of type T_1 and one occupied by an agent of type T_3. There is also an empty node z which is adjacent to two nodes, one occupied by an agent of type T_1 and one occupied by an agent of type T_2. In $\mathcal{I}$, agent i has no incentive to jump from v_i to z as both nodes give her utility $1/2$. On the other hand, in $\mathcal{I}'$, agent i has utility $(1 + 0)/2 = 1/2$ and has incentive to jump to v to increase her utility to $(1 + 1/2)/2 = 3/4$. Hence, $\mathbf{v}$ can be an equilibrium of $\mathcal{I}$, but not of $\mathcal{I}'$.
- In $\mathbf{v}'$, an agent i of type T_1 occupies a node v_i which is adjacent to three nodes, one occupied by an agent of type T_1, one occupied by an agent of type T_2 and one occupied by an agent of type T_3. There is also an empty node z which is adjacent to two nodes, one occupied by an agent of type T_1 and one occupied by an agent of type T_3. In $\mathcal{I}$, agent i has incentive to jump from v_i to z in order to increase her utility from $1/3$ to $1/2$. However, in $\mathcal{I}'$, agent i has no incentive to jump as she has utility $(1 + 1/2 + 0)/3 = 1/2$ by occupying node v_i, which is exactly the utility $(1 + 0)/2$ she would also obtain by jumping to z. Consequently, $\mathbf{v}'$ can be an equilibrium of $\mathcal{I}'$, but not of $\mathcal{I}$.

This completes the proof. □

Since there exist simple 2-ZTS games that do not admit any equilibria [18], the first part of Theorem 1 implies that equilibria are not guaranteed to exist for general 2-TS games as well. In fact, by carefully inspecting the proof of Elkind

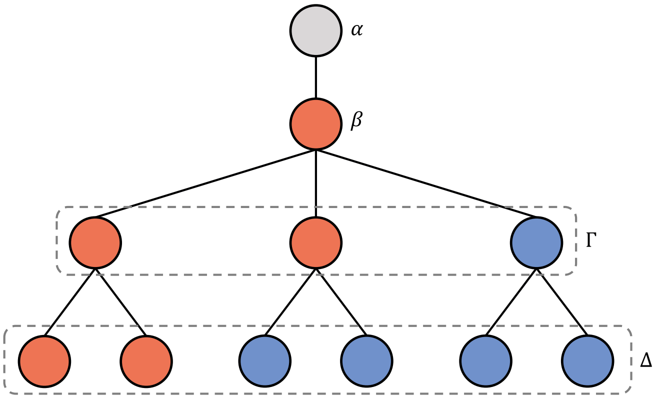

Fig. 2. The topology used in the proof of Theorem 2 for $\lambda = 2$. In the depicted assignment with red and blue agents, the empty node is $v = \alpha$ (first case); the red agent with one red and two blue neighbors has an incentive to deviate to α in order to increase her utility from $\frac{1+2t_1}{3} < 1$ to 1.

et al. [18] that λ-ZTS games played on trees do not always admit equilibria for every $\lambda \geq 2$, we can show the following stronger impossibility result.

Theorem 2. *For every $\lambda \geq 2$ and every tolerance vector $\mathbf{t}_\lambda$ such that $t_1 < 1$, there exists a λ-TS game $\mathcal{I} = (N, G, \mathbf{t}_\lambda)$ in which G is a tree and does not admit any equilibrium.*

Proof. We will show how the proof of Elkind et al. [18] that λ-ZTS games do not always admit equilibria can be directly extended to capture the class of λ-TS games with appropriate tolerance vectors. Consider a λ-TS game $\mathcal{I} = (N, G, \mathbf{t}_\lambda)$ where N is a set of $n = \lambda(2\lambda + 1)$ agents partitioned into $\lambda \geq 2$ types such that there are $(2\lambda + 1)$ agents per type, and $\mathbf{t}_\lambda$ is a tolerance vector with $t_1 < 1$. The topology G is a tree consisting of $n + 1$ nodes, distributed to four levels. There is a root node α with one child node β, which in turn has a set Γ of $2\lambda - 1$ children. Every node $\gamma \in \Gamma$ has a set Δ_γ of λ children nodes, which are leaves of the tree; let $\Delta = \bigcup_{\gamma \in \Gamma} \Delta_\gamma$. See Fig. 2 for the topology in the case where $\lambda = 2$ and for an example assignment.

Adapting the arguments of Elkind et al. [18], we will now show that this game does not admit any equilibrium. Assume otherwise, that there exists an equilibrium assignment $\mathbf{v}$. We distinguish between cases by enumerating the single node v that is left empty:

- $v = \alpha$. Suppose that the agent i who occupies node β is of type T. Since $|\Gamma| = 2\lambda - 1$, $|T \setminus \{i\}| = 2\lambda$, and $|\Delta_\gamma| = \lambda$ for each $\gamma \in \Gamma$, there must be an agent $j \neq i$ of type T with a neighbor of type $T' \neq T$. Hence, agent j has utility strictly less than 1, and has incentive to deviate to α to obtain maximum utility 1.
- $v = \beta$. Suppose that the agent i who occupies node α is of type T_ℓ. Since she is isolated, her utility is 0. Since $\mathbf{v}$ is assumed to be an equilibrium, agent i would still have utility 0 by deviating to β; note that this is impossible if all tolerance parameters are positive. As a result, all the nodes in Γ must be occupied by agents of types T_k such that $t_{|\ell-k|} = 0$. But then, this means that all agents of type T_ℓ, who must occupy nodes of Δ, also have utility 0 and incentive to deviate to β, connect to agent i, and obtain positive utility.
- $v \in \Gamma$. Since v is empty, all the agents occupying nodes of Δ_v have utility 0. Since all λ nodes of Δ_v are occupied, it must be that there exist two agents in $\Delta_v \bigcup \beta$ of the same type. At least one of these agents occupies a node in Δ_v and can increase her utility from 0 to positive by moving to v and connecting with the other agent of the same type.
- $v \in \Delta$. Suppose that the agent i who occupies the parent of v is of type T_ℓ. For such an assignment to be an equilibrium, it must be the case that all other agents of type T_ℓ have utility 1, as otherwise they would have incentive to deviate to v. Since $t_1 < 1$, this would mean that all other agents of type T_ℓ have only neighbors of type T_ℓ; it is easy to see that this is impossible.

This completes the proof. □

Since Theorem 2 implies that it is impossible to hope for general positive existence results, in the remainder of this section we focus on games with structured topologies and tolerance vectors. In particular, we consider the class of α-binary λ-TS games with $\alpha \in \{1, \ldots, \lambda\}$ in which the tolerance vector $\mathbf{t}_\lambda$ is such that

$$t_d = \begin{cases} 1, & \text{if } d < \alpha \\ 0, & \text{otherwise.} \end{cases}$$

Clearly, the class of 1-binary λ-TS games coincides with that of λ-ZTS.

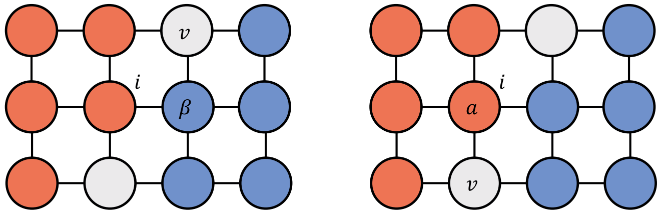

Fig. 3. The figure used in the proof of Theorem 3. The left subfigure corresponds to Case I in the proof, while the right subfigure corresponds to Case II.

We next show that when the topology is a grid[2] or a tree, there exist values of $\alpha \in \{1, \dots, \lambda\}$ for which α-binary λ-TS games played on such a topology always admit at least one equilibrium. Our first result for grids is the following.

Theorem 3. *Every 2-ZTS game $\mathcal{I} = (N, G)$ in which G is a grid admits at least one equilibrium.*

Proof. Consider an arbitrary 2-ZTS game $\mathcal{I} = (N, G)$ in which the topology G is an $m \times M$ grid (m rows and M columns), such that $m \le M$. Also, let $x \ge 2$ be the number of agents per type, such that $n = 2x$. Clearly, since $n \ge 4$ it holds $M \ge 3$. To simplify our discussion, we will refer to the two types as red and blue.

We construct an equilibrium by assigning the agents to the nodes of the grid column-wise from top to bottom and left to right as follows: We first assign all the red agents. We then leave a number $e = m \cdot M - n \ge 1$ of nodes empty, and finish by assigning all the blue agents. Let $\mathbf{v}$ be the assignment computed.

Clearly, if $e \ge m$, $\mathbf{v}$ is guaranteed to be an equilibrium since all agents have maximum utility 1. Otherwise, when $e < m$ and since $M \ge 3$, it cannot be that there are $x \le m$ agents per type. So, in the following we focus on the case where $e < m$ and $x > m$.

Let v be an empty node, and consider any red agent i that occupies node v_i according to $\mathbf{v}$; see Fig. 3. We will argue that i has no incentive to jump to v; the argument for blue agents is symmetric. Observe that agent i has utility at least $1/2$, since she is connected to at least one red agent and at most one blue agent by construction. We will focus on the case where agent i has exactly one blue neighbor, as otherwise she clearly has no incentive to jump to v. By the construction of $\mathbf{v}$ and the fact that $e < m$, we have that (i) the left neighbor of v is occupied by a red agent, (ii) the right neighbor of v is occupied by a blue agent, (iii) the top neighbor of v, call it α, is either empty or occupied by a red agent (if v is in the top row, we refer to α as empty), and (iv) the bottom neighbor of v, call it β, is either empty or occupied by a blue agent (if v is in the bottom row, we refer to β as empty). We distinguish between the following cases:

- Case I: α, β are both empty, or α is empty and β is occupied by a blue agent, or α is occupied by a red agent and β is occupied by a blue agent; the second case is represented in the left subfigure of Fig. 3. Agent i has no incentive to deviate to v since by doing so she would only be able to get utility at most $1/2$.
- Case II: α is occupied by a red agent and β is empty; see the right subfigure of Fig. 3. If agent i occupies α she has no incentive to deviate to v, as she would be able to get utility exactly $1/2$, while right now she has utility at least $1/2$. If agent i is adjacent to an empty node but does not occupy α, then i has utility 1 and is not interested in deviating. As a result, we have that agent i is not adjacent to any empty node, and by deviating to v agent i would obtain utility $2/3$. The only case in which agent i would have incentive to deviate to v is if her utility is exactly $1/2$, which would mean that she has exactly one red neighbor and one blue neighbor. This is possible only if v_i is the last node of the first column of the grid and $M = 3$. However, this contradicts the fact that $m \le M$, since there is a column with two empty nodes (v and β), one node (α) occupied by a red agent, and one node (the right neighbor of v_i) occupied by a blue agent.

This completes the proof. $\square$

The proof of Theorem 3 is constructive and such that in the computed equilibrium no agent is isolated. Consequently, in combination with Theorem 1, it further implies the following:

Corollary 4. *Every 2-TS game $\mathcal{I} = (N, G, \mathbf{t_2})$ in which G is a grid admits at least one equilibrium.*

Unfortunately, showing a result similar to Theorem 3 for every $\lambda \ge 3$ is a very challenging task. Instead, we show the following result for 2-binary games.

Theorem 5. *Every 2-binary λ-TS game $\mathcal{I} = (N, G, \mathbf{t}_\lambda)$ in which G is a grid admits at least one equilibrium.*

[2] We focus on 4-grids where internal nodes have 4 neighbors.

Algorithm 1: TILE(s, r, k).

```
/* s,k: starting row and number of nodes to be left empty in that row     */
/* r ≤ x: number of rows defining an r × M grid                          */
for i = 1 to k do
    mark node (s, i) as empty
for j = 1 to M do
    for i = s to s + r − 1 do
        if node (i, j) is unmarked then
            place the next agent (if one exists) according to the ordering ≻ at node (i, j)
```

Proof. Consider a 2-binary λ-TS game with n agents played on an $m \times M$ grid (m rows and M columns) such that $m \le M$. Let $x = n/\lambda \ge 2$ be the number of agents per type and $e = mM - n$ be the number of empty nodes.

We compute an equilibrium assignment $\mathbf{v}$ using Algorithm 2, which in turn relies on the TILE procedure described in Algorithm 1. In particular, Algorithm 1 takes as input three arguments: the starting row s, the number of nodes k to be left empty at row s, and the number of rows $r \le x$ defining an $r \times M$ subgrid with s as its topmost row. It considers the yet unassigned agents in increasing type, according to the ordering $\succ$, and assigns them in the $r \times M$ subgrid, so that the k leftmost nodes of row s are left empty, while all other nodes host an agent (assuming the number of unassigned agents is large enough). Algorithm 1 visits these rows in a column-major order, skipping the empty nodes. The key property of Algorithm 1 is that, for each agent of type ℓ processed, all its neighboring agents inside the $r \times M$ subgrid have types in $\{\ell-1, \ell, \ell+1\}$; this holds as we assume that $r \le x$.

Algorithm 2 repeatedly calls Algorithm 1 to compute an assignment for non-overlapping subgrids, where the number of rows in each subgrid varies but the number of columns is always M. We remark that Algorithm 2 terminates immediately (at any step) when all agents have been assigned. The key objective is to guarantee that almost all agents have utility 1, while those with smaller utility have no improving deviation. To do so, we exploit that each agent of type ℓ, assigned during the execution of Algorithm 1, has all its neighboring agents inside the $r \times M$ subgrid with types in $\{\ell-1, \ell, \ell+1\}$ and we handle carefully its neighboring agents outside that subgrid.

While the number of rows not already handled by Algorithm 1 is at least x and the number of empty nodes is at least M, Algorithm 2 calls Algorithm 1 to handle a subgrid and leaves the next row empty (see Lines 4-5); by the discussion above, this guarantees that all agents placed until that point have utility 1. Whenever $e \ge M$ and the number of remaining rows is less than x, running Algorithm 1 again guarantees that all agents have utility 1 (see Lines 6-7) as, once more, each agent of type ℓ has only neighbors with types in $\{\ell-1, \ell, \ell+1\}$. So far, we have argued that if Algorithm 2 terminates in Lines 3 or 7, $\mathbf{v}$ is an equilibrium.

The challenging case, therefore, arises when the number of remaining empty nodes becomes smaller than M, as it is no longer feasible to guarantee that all agents have utility 1 by separating subgrids with empty rows. The specific way in which we handle this last part depends on how the number of remaining rows relates to x, the number of agents per type; this is the focus of lines 8-25, where we allocate the remaining agents so that those with utility less than 1 have no convenient deviation.

Note that the algorithm cannot terminate at Line 11 as $e < M$ and, therefore, the number of empty nodes is not large enough to cover the remaining subgrid. So, the first non-trivial case is when the algorithm terminates in Line 14; see also the rightmost example in Fig. 4. Again, all agents placed in Line 3 have utility 1. Each agent i of type ℓ placed during Line 11 has utility at least 2/3; indeed, i has at least one neighbor of type ℓ, at least one neighbor of a type in $\{\ell-1, \ell+1\}$, and at most one neighbor of type at distance at least 2. If $\alpha = 1$, all agents placed in Line 14 have utility 1. Otherwise, agents placed in Line 14 at the last $x-1$ rows have utility 1, while any agent on the row with the empty nodes has utility at least 1/2 when $e = M - 1$, and at least 2/3 otherwise.

If the algorithm terminates in Line 19 (see also the leftmost part of Fig. 4), agents placed in Lines 3, 11, or 19 have utility at least 2/3, while agents placed in Line 18 have utility at least 1/2. If the algorithm terminates in Line 22, agents placed in Line 3 have utility 1, agents placed in Lines 11 and 22 have utility at least 2/3, while agents placed in Line 21 have utility at least 2/3 except (perhaps) the first and the last agent on the row that have utility at least 1/3. If the algorithm terminates in Line 25, again all agents placed in Line 3 have utility 1, while agents placed in Lines 11 and 24 have utility at least 2/3. Finally, the agents placed in Line 25 have utility at least 1/2 if $e = M - 1$ and at least 2/3 otherwise.

We now argue that no agent has an incentive to jump. Note that an empty node may have another empty node as a top or bottom neighbor if the algorithm terminates in Line 3, or in Line 7, or in Line 14 in case $\alpha = 1$. In all these cases, by the discussion above, all agents have utility 1 and the assignment is an equilibrium. Also, note that an empty node has always a bottom neighbor, while the only case the empty node has no top neighbor is if the algorithm terminates in Line 19. In that case, any agent with utility less than 1 can obtain utility at most 1/2 by jumping; again, $\mathbf{v}$ is an equilibrium.

So, in the following we assume that any empty node has a top and bottom neighboring agent. Observe that, in that case, an agent gets utility at most 2/3 by jumping to an empty node, since either the top or the bottom neighbor will have a large type distance and there is no left neighbor. As in almost all cases, agents in $\mathbf{v}$ have utility at least 2/3, it remains to argue about the nodes that have utility less than that. The agent in Line 14 with utility 1/2 (when $\alpha > 1$) obtains utility at most 1/2 by jumping, the agents in Line 21 with utility at least 1/3 obtain utility at most 1/3 by jumping, and, finally,

Algorithm 2: Equilibrium construction for a 2-binary λ-TS game on an $m \times M$ grid.

```
   /* Algorithm 2 terminates immediately if all agents have been assigned   */
   /* x, e: number of agents per type and number of empty nodes              */
1  Initialize k = 0
2  while x ≤ m − k and e ≥ M do /* Each agent placed in this loop has utility 1 */
3      TILE(k + 1, x, 0)
4      leave the next row empty
5      update k := k + x + 1, e := e − M
6  if x > m − k then /* Each agent placed in this step has utility 1 */
7      TILE(k + 1, m − k, 0)
8  else /* We now have e < M and x ≤ m − k */
9      Define non-negative integers α ∈ ℕ_{>0} and β ≤ x − 1 such that m − k = αx + β
10     for i = 1, ..., α − 1 do /* Algorithm 2 cannot terminate in this loop */
11         TILE(k + 1, x, 0)
12         update k := k + x
       /* Below we handle the last x + β rows */
13     if β = 0 then
14         TILE(k + 1, x, e)
15     else if β = 1 then
16         if Line 3 was executed then
17             Shift all agents down by one row
18             TILE(1, 1, e)
19             TILE(k + 2, x, 0)
20         else
21             TILE(k + 1, 1, e)
22             TILE(k + 2, x, 0)
23     else
24         TILE(k + 1, x, 0)
25         TILE(k + x + 1, β, e)
```

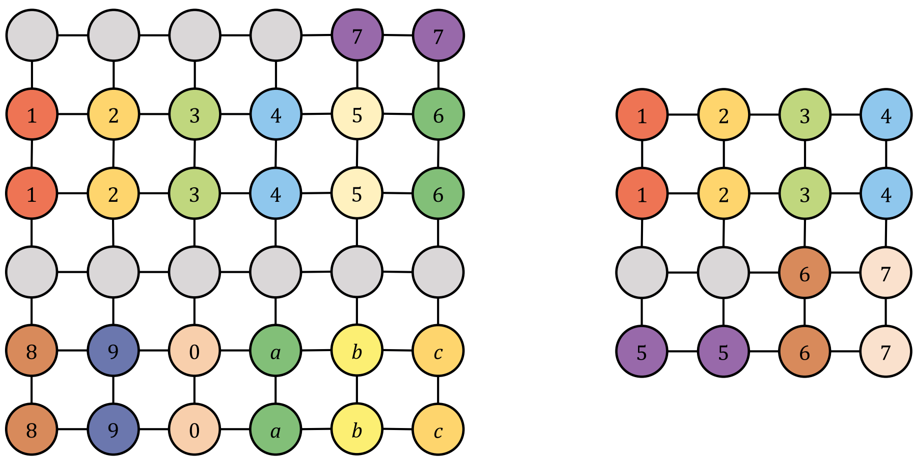

Fig. 4. On the left, an example of how Algorithm 2 operates when it terminates in Line 19. On the right, an example when the algorithm terminates in Line 14. Agents of the same number and color are of the same type, while $\succ$ is $\{1, 2, \ldots, 9, 0, a, b, c\}$.

the agent in Line 25 with utility $1/2$ obtains utility at most $1/2$ by jumping. We conclude that $\mathbf{v}$ is an equilibrium and the theorem follows. □

Note that Algorithm 2 may fail to return an equilibrium for lexicographically larger tolerance vectors. Indeed, consider again the rightmost example in Fig. 4 depicting a 4×4 grid and 7 types of two agents each. Algorithm 2 puts agents of types 1 to 4 in each of the first two rows, skips 2 nodes, puts agents of types 6 and 7 in the third row, and places agents of types 5, 5, 6 and 7 in the last row. Under tolerance vector $\mathbf{t_7} = \{1, 1, 0, 0, 0, 0, 0\}$ the assignment is an equilibrium (by Theorem 5), while under tolerance vector $\mathbf{t}'_7 = \{1, 1, t_2 > \frac{1}{2}, 0, 0, 0, 0\}$, the agent of type 4 in the second row has utility $2/3$, but can obtain utility $\frac{1+2t_2}{3} > 2/3$ by jumping to the rightmost empty node.

So, a different algorithm is needed for computing equilibria in α-binary games with $\alpha \geq 3$. While we have not been able to show this result for every α, we do show it for $\alpha \geq \sqrt{\lambda}$. In particular, the equilibrium constructed in the proof of

Algorithm 3: Equilibrium construction for a $\lfloor \frac{\lambda}{2} \rfloor$-binary λ-TS game on a tree (or games with lexicographically larger tolerance vectors).

```
/* tree_1,...,tree_k denote the subtrees of the tree topology in non-increasing order by size, when the
   topology is rooted at a centroid node.                                                            */
```

1 Run Bottom-Up($tree_1, T_1, T_2, \dots, T_{\lceil \frac{\lambda}{2} \rceil}$). If at least one agent of type T_1 remains unassigned, repeat with the next subtree. Let $a \le \lceil \frac{\lambda}{2} \rceil$ be the smallest type index among unassigned agents, and let $tree_{k_1}$ be the last subtree considered in this step.

2 Run Bottom-Up($tree_{k_1+1}, \mathcal{T}_\lambda, \mathcal{T}_{\lambda-1}, \dots, \mathcal{T}_{\lceil \frac{\lambda+1}{2} \rceil}$), where $\mathcal{T}_i$ are the unassigned agents of types T_i, $i = \lambda, \dots, \lceil \frac{\lambda+1}{2} \rceil$. If at least one agent of type T_λ remains unassigned, repeat with the next subtree. Let $b \ge \lceil \frac{\lambda+1}{2} \rceil$ be the largest type index among unassigned agents, and let $tree_{k_2}$ be the last subtree considered in this step.

3 Run Bottom-Up($tree_{k_2+1}, \mathcal{T}_a, \mathcal{T}_{a+1}, \dots, \mathcal{T}_b$), where $\mathcal{T}_i$ are the unassigned agents of types T_i, $i = a, \dots, b$. Repeat with the next subtree and the unassigned agents of these types, until all agents have been assigned.

4 If the last subtree among the ones considered in the previous steps contains at least two isolated agents, then rearrange them within this subtree so that each of them has at least one neighbor. If the last subtree contains a single isolated agent, then move this agent to the root of the tree.

the next theorem guarantees a utility of 1 to all agents, and thus it is also an equilibrium for games with lexicographically larger tolerance vectors, not necessarily binary ones.

Theorem 6. *For $\lambda \ge 3$, every $\sqrt{\lambda}$-binary λ-TS game $\mathcal{I} = (N, G, \mathbf{t}_\lambda)$ in which G is a grid admits at least one equilibrium.*

Proof. Consider a $\sqrt{\lambda}$-binary λ-TS game $\mathcal{I} = (N, G, \mathbf{t}_\lambda)$ in which the topology G is an $m \times M$ grid (m rows and M columns) such that $m \le M$. Observe that if an agent i is assigned to a node v_i such that *all* her neighbors are of types in distance strictly less than $\sqrt{\lambda}$ according to the ordering $\succ$, then agent i has maximum utility 1, and no incentive to deviate from v_i. This is the property we will exploit to construct an equilibrium assignment. We distinguish between the following two cases:

- $m \le \sqrt{n}$. We construct an equilibrium assignment $\mathbf{v}$ by considering the agents in increasing type according to the ordering $\succ$, and assign them one after the other along the columns, from top to bottom and from left to right. Since there are at most $\sqrt{n}$ rows and $\sqrt{n} = \frac{n}{\lambda} \cdot \frac{\lambda}{\sqrt{n}}$, each column of the grid can fit at most all the agents of $\frac{\lambda}{\sqrt{n}}$ different types. This means that, by the construction of $\mathbf{v}$, every agent i of type ℓ has neighbors of types T_k such that $|\ell - k| \le \frac{\lambda}{\sqrt{n}} < \sqrt{\lambda}$, where the last inequality follows by the fact that $n \ge 2\lambda$. Hence, agent i has utility 1 and no incentive to deviate.
- $m > \sqrt{n}$. Consider the sub-grid containing only the nodes in the first $\sqrt{n}$ rows. Since $M \ge m > \sqrt{n}$, all n agents can be assigned to these nodes, and we thus can repeat the process of the previous case by limiting our attention to the sub-grid. This again guarantees that every agent has utility 1 and the resulting assignment is an equilibrium.

This completes the proof. □

Next we turn our attention to games in which the topology is a tree. We show the following result for α-binary games when $\lambda \ge 3$.

Theorem 7. *Every 2-binary 3-TS game $\mathcal{I} = (N, G, \mathbf{t_3})$ and every α-binary λ-TS game $\mathcal{I} = (N, G, \mathbf{t}_\lambda)$ where $\alpha \ge \lfloor \frac{\lambda}{2} \rfloor$ for $\lambda \ge 4$, in which G is a tree, admit at least one equilibrium.*

Proof. To construct an equilibrium, we exploit the following known property of trees: Every tree with $x \ge 3$ nodes contains a *centroid* node, whose removal splits the tree into at least two subtrees with at most $x/2$ nodes each. We root the tree from such a centroid node, and leave the root empty. This leads to a partition of the topology in $k \ge 2$ subtrees, which we order in non-increasing size and denote by $tree_1, \dots, tree_k$.

To assign the agents we use Algorithm 3, which in turn uses the Bottom-Up allocation procedure (described in Algorithm 4). The procedure Bottom-Up($tree, \mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_s$) assigns the unassigned agents of types $\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_s$ to the nodes of the subtree *tree* from bottom to top (higher to lower depth), so that the parent node of each agent of $\mathcal{T}_1$ is occupied by either an agent of the same type or an agent of type $\mathcal{T}_2$, and the assignment in the corresponding subtree is connected. Informally, Algorithm 3 roots the topology at a centroid node and considers subtrees in non-increasing size. As long as agents of type T_1 are remaining, Algorithm 3 applies the Bottom-Up procedure to the next subtree with agents in increasing type index. Then, as long as agents of type T_λ are remaining, Algorithm 3 applies the Bottom-Up procedure to the next subtree with agents in decreasing type index. The remaining (smaller) subtrees are filled with the remaining agents, again using the Bottom-Up procedure. See Fig. 5 for the assignment computed by Algorithm 3 for the case of $\lambda = 4$ types of three agents each.

We first claim that at the end of Step 3 of Algorithm 3, every agent either gets utility 1 or gets utility 0 if she is isolated. Indeed, it holds that agents of type T_1 can only be adjacent to agents of type T_1 and T_2. Similarly, the agents of type T_λ can only be adjacent to agents of type T_λ and $T_{\lambda-1}$. In addition, by design, the maximum type distance among all the

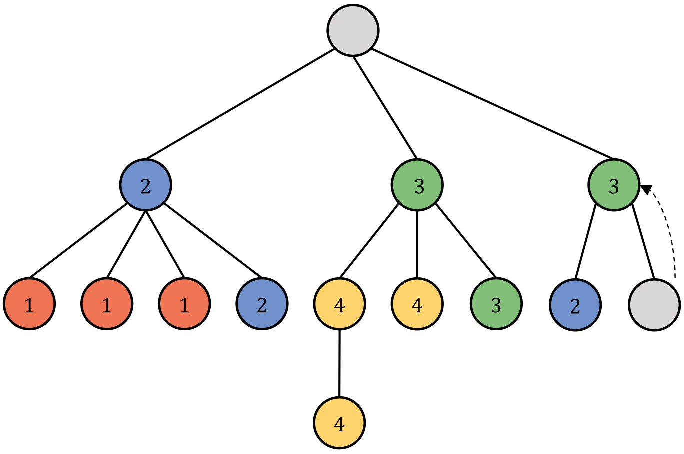

Fig. 5. The assignment computed by Algorithm 3 in the proof of Theorem 7 for the case of $\lambda = 4$ types of three agents each. Agents of type T_1 are colored red, agents of type T_2 are colored blue, agents of type T_3 are colored green, and agents of type T_4 are colored yellow. The arrow represents the change performed in Step 4 of the algorithm.

Algorithm 4: BOTTOM-UP($tree, \mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_s$).

/* For $i = 1, \ldots, s$, $\mathcal{T}_i$ is the set of unassigned agents of a given type */
/* The algorithm terminates immediately when all agents have been assigned or all nodes of *tree* have been occupied. */

1 Start at the lowest level of *tree* and place agents of type $\mathcal{T}_1$ so that an agent of type $\mathcal{T}_1$ is placed at level h only if all nodes at levels at least $h+1$ have been filled. Furthermore, and assuming the previous condition holds, after filling a node at level h we give priority to its sibling nodes. Continue until all agents of type $\mathcal{T}_1$ have been assigned.

2 Consider the agents of type $\mathcal{T}_2$. Begin by placing an agent of type $\mathcal{T}_2$ to any empty node having a child occupied by an agent of type $\mathcal{T}_1$ and repeat until the parent nodes of all agents of type $\mathcal{T}_1$ are occupied. This is feasible as long as there are at least as many agents of type $\mathcal{T}_2$ as there are agents of type $\mathcal{T}_1$. Continue by placing agents of type $\mathcal{T}_2$ arbitrarily in *tree* by maintaining a connected assignment for all agents in *tree*.

3 Arbitrarily assign the remaining agents in order of input so that the assignment in *tree* remains connected after assigning each agent.

other agents assigned in Steps 1 and 2 is $\lceil \frac{\lambda}{2} \rceil - 2$. By this discussion, all agents have utility 1 when $\lambda = 3$ and the game is 2-binary. Below, we assume that $\lambda \geq 4$.

To see the claim is true for agents assigned in Step 3, observe that if Step 2 is applied on a subtree of at least $n/3$ nodes, then since we visit subtrees in non-increasing order of their size, Step 1 is also applied on a subtree of at least $n/3$ nodes. Hence, at most $n/3$ agents remain to be allocated. Otherwise, if no subtree on which Step 2 is applied has at least $n/3$ nodes, then, again due to the order we visit subtrees, any subtree to which we perform Step 3 has less than $n/3$ nodes. In any case, at most $n/3$ agents will be allocated at Step 3 at any given subtree. These agents belong to at most $\lceil \lambda/3 \rceil + 1$ different types and, due to Steps 2 and 3 in Algorithm 4, we are guaranteed that no agent allocated in Step 3 will have a neighbor of type-distance $\lceil \lambda/3 \rceil$. Since $\lceil \lambda/3 \rceil - 1 \leq \lfloor \lambda/2 \rfloor - 1$, such agents either get utility 1, or 0 if they are isolated, as required.

It remains to argue that after a possible execution of Step 4, no agent has a profitable deviation. We distinguish between the following two cases when Step 4 is performed:

- Case I: There are at least two isolated agents in the last subtree among those considered in the first three steps. First observe that, since the subtrees are considered in non-increasing order by size and the last subtree contains at least two agents, there is no subtree with a single isolated agent. Now, by the definition of the bottom-up-like allocation algorithm, all these agents must be of the last type T_b, since if agents of two or more types are assigned in the same subtree, the resulting assignment therein is by construction connected. Therefore, by rearranging the agents of type T_b in the last subtree so that all of them have at least one neighbor, each of them gets utility 1 and the assignment is an equilibrium.
- Case II: There is a single isolated agent i in the last subtree of the last type T_b considered, who is moved to the root of the tree. Since Step 4 is performed, all the subtrees that have been considered in the first three steps are full, with the exception of the last subtree which has been left empty after moving agent i. Thus, the empty nodes of the topology are only adjacent to other empty nodes or the root. As a result, an agent of some type $\ell \in [\lambda]$ would be able to get utility $t_{|\ell-b|}$ by jumping to an empty node that is adjacent to the root, and utility 0 by jumping to any other empty node. However, every agent $j \neq i$ already has utility at least $t_{|\ell-b|}$. In particular, agent j has utility 1 if she is not adjacent to the root, utility at least $\frac{1+t_{|\ell-b|}}{2} \geq t_{|\ell-b|}$ if she is adjacent to the root but not isolated before moving i to the root, and utility exactly $t_{|\ell-b|}$ if she is adjacent to the root and was isolated before moving i to the root.

This completes the proof. □

For $\lambda = 3$, Theorem 7 is tight in the sense that equilibria are not guaranteed to exist when $t_1 < 1$ (Theorem 2). For $\lambda \geq 4$, it is not hard to observe that the assignment computed is also an equilibrium in games with lexicographically larger vectors (not necessarily binary ones) than the one stated.

4. Quality of equilibria

In this section, we consider the quality of equilibria measured in terms of social welfare, and bound the price of anarchy and price of stability. Recall that these notions compare the social welfare achieved in the *worst* and *best* equilibrium to the maximum possible social welfare achieved in any assignment. We start with a general upper bound on the price of anarchy, whose proof follows by bounding the social welfare at equilibrium by the total utility the agents would be able to obtain by jumping to an arbitrary empty node. Recall that $\tau = \sum_{d=0}^{\lambda-1} t_d$.

Theorem 8. *The price of anarchy of λ-TS games with tolerance vector $\mathbf{t}_\lambda$ is at most $\frac{\lambda n}{\tau n - \lambda}$.*

Proof. Consider a λ-TS game $\mathcal{I} = (N, G, \mathbf{t}_\lambda)$ with $\text{EQ}(\mathcal{I}) \neq \varnothing$. Let $\mathbf{v}$ be an equilibrium, and denote by v an empty node. The utility that an agent of type T_ℓ, $\ell \in [\lambda]$ would obtain by unilaterally jumping to v is

- $\frac{1}{n(v)} \sum_{k\in[\lambda]} t_{|\ell-k|} \cdot n_k(v)$ if she is not adjacent to v;
- $\frac{1}{n(v)-1} \left(\sum_{k\in[\lambda]} t_{|\ell-k|} \cdot n_k(v) - 1\right)$ otherwise.

Also observe that for every type T_ℓ, $\ell \in [\lambda]$ there are exactly $\frac{n}{\lambda} - n_\ell(v)$ agents that are not adjacent to v, and $n_\ell(v)$ agents that are adjacent to v. Since $\mathbf{v}$ is an equilibrium, every agent of type T_ℓ is guaranteed to have at least as much utility as if she were to deviate to v, and therefore the social welfare is

$$\begin{aligned}
\text{SW}(\mathbf{v}) &\geq \frac{1}{n(v)} \sum_{\ell\in[\lambda]} \left(\frac{n}{\lambda} - n_\ell(v)\right) \sum_{k\in[\lambda]} t_{|\ell-k|} \cdot n_k(v) + \frac{1}{n(v)-1} \sum_{\ell\in[\lambda]} n_\ell(v) \cdot \left(\sum_{k\in[\lambda]} t_{|\ell-k|} \cdot n_k(v) - 1\right) \\
&\geq \frac{1}{n(v)} \sum_{\ell\in[\lambda]} \left(\frac{n}{\lambda} \sum_{k\in[\lambda]} t_{|\ell-k|} \cdot n_k(v) - n_\ell(v)\right) \\
&= \frac{1}{n(v)} \sum_{\ell\in[\lambda]} n_\ell(v) \cdot \left(\frac{n}{\lambda} \sum_{k\in[\lambda]} t_{|\ell-k|} - 1\right) \\
&= \frac{1}{\lambda \cdot n(v)} \sum_{\ell\in[\lambda]} n_\ell(v) \left(n \sum_{k\in[\lambda]} t_{|k-\ell|} - \lambda\right).
\end{aligned}$$

The second inequality is due to increasing the denominator of the second fraction. The first equality follows by aggregating the factors of $n_\ell(v)$ for every $\ell \in [\lambda]$. Finally, the second equality follows by factorizing λ. Now observe that because the tolerance vector $\mathbf{t}_\lambda$ is non-increasing, we have that $\sum_{k\in[\lambda]} t_{|\ell-k|} \geq \sum_{d=0}^{\lambda-1} t_d = \tau$. Combining this together with the fact that $n(v) = \sum_{\ell\in[\lambda]} n_\ell(v)$, we obtain

$$\text{SW}(\mathbf{v}) \geq \frac{\tau n - \lambda}{\lambda}.$$

The bound on the price of anarchy follows by the fact that the optimal welfare is at most n (the maximum utility of any agent is 1). □

For each $\ell \in \{1, \ldots, \lambda\}$, let $\tau_\ell = \sum_{k\in[\lambda]} t_{|\ell-k|}$ be the total tolerance of agents of type ℓ towards any subset containing one agent of every type. We can show the following general lower bound on the price of anarchy, as a function of these parameters.

Theorem 9. *The price of anarchy of λ-TS games with tolerance vector $\mathbf{t}_\lambda$ is at least*

$$\frac{\lambda n}{\frac{\sum_{\ell\in[\lambda]} \tau_\ell}{\lambda} n - \frac{\lambda^2 - \sum_{\ell\in[\lambda]} \tau_\ell}{\lambda-1}} \geq \frac{\lambda n}{\frac{2(\lambda-1)\tau}{\lambda} n - \frac{\lambda^2}{\lambda-1} + 2\tau}.$$

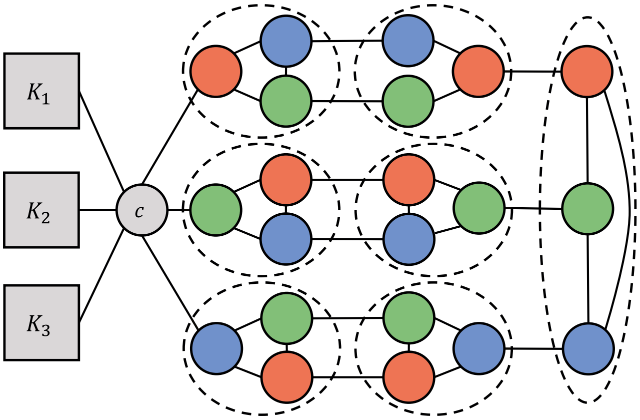

Fig. 6. An instance used for the proof of Theorem 9 for the case of 3 types and 21 agents, so that each type has 7 agents. The big squares K_1, K_2, K_3 correspond to cliques of size 7 (the number of agents per type), while the ovals represent cliques of size 3 (the number of types). In an optimal assignment, each large clique contains agents of the same type and each agent gets utility 1. In a bad equilibrium, each small clique contains a single agent of each type and all gray nodes are left empty. For each type $\ell \in [3]$, all but one agents of type ℓ get utility $\tau_\ell/3$, while the last agent gets utility $(\tau_\ell - 1)/2$.

Proof. Consider a λ-TS game $\mathcal{I} = (N, G, \mathbf{t}_\lambda)$, where $n = |N| = \lambda(2\lambda\mu + 1)$ for some integer $\mu \geq 1$. The topology G contains $2n + 1$ nodes as follows. There exists a central node c and λ large cliques K_i, with $i \in [\lambda]$, where each K_i contains $2\lambda\mu + 1$ nodes. There also exist $2\lambda\mu$ small cliques $K_{i,j}$, with $i \in [\lambda]$ and $j \in \{1, \dots, 2\mu\}$, each of size λ, as well as another small clique K of size λ. Apart from the edges within the cliques, we add the following edges so that any node of a small clique is connected to a single node outside its clique. First, we connect the central node c to a single node of cliques K_i and $K_{i,1}$, for $i \in [\lambda]$. Then, for each $K_{i,1}$, with $i \in [\lambda]$, we connect each node that is not a neighbor of c to a single distinct node in $K_{i,2}$. Similarly, for each $K_{i,j}$, with $i \in \{2, \dots, \lambda\}$ and $j \in \{1, \dots, 2\mu - 1\}$, we connect any node without a neighbor in $K_{i,j-1}$ to a single distinct node in $K_{i,j+1}$. Finally, for each $K_{i,2\mu}$, with $i \in \{1, \dots, 2\mu\}$, we connect the only node with no neighbors in $K_{i,2\mu-1}$ to a single distinct node in K; this completes the description of G.

In the optimal assignment, for $\ell \in [\lambda]$, all agents of type ℓ are at the large clique K_ℓ and $\mathrm{OPT}(\mathcal{I}) = n$. There exists an equilibrium assignment $\mathbf{v}$ where all agents are placed in the small cliques as follows: each small clique contains a single agent of each type, the agents neighboring the (empty) central node are all of distinct types, and, finally, for each pair of neighboring agents across different small cliques, both agents are of the same type. It is always possible to obtain such an assignment; see Fig. 6 for an example with $\lambda = 3$ and $\mu = 1$.

Observe that, for $\ell \in [\lambda]$, each agent i of type ℓ that is not a neighbor of the central node has utility $u_i(\mathbf{v}) = \frac{\tau_\ell}{\lambda}$ and will obtain the same utility if she jumps to the central node. Similarly, each agent i of type ℓ who is a neighbor of c has utility $\frac{\tau_\ell - 1}{\lambda - 1}$ and will get the same utility by jumping to the central node. Clearly, any agent will get utility 0 by jumping to another empty node. Hence, $\mathbf{v}$ is an equilibrium with

$$
\begin{aligned}
\mathrm{SW}(\mathbf{v}) &= \sum_{\ell \in [\lambda]} \left(2\lambda\mu \frac{\tau_\ell}{\lambda} + \frac{\tau_\ell - 1}{\lambda - 1} \right) \\
&= \sum_{\ell \in [\lambda]} \left(\left(\frac{n}{\lambda} - 1 \right) \frac{\tau_\ell}{\lambda} + \frac{\tau_\ell - 1}{\lambda - 1} \right) \\
&= \frac{\sum_{\ell \in [\lambda]} \tau_\ell}{\lambda^2} n + \frac{\sum_{\ell \in [\lambda]} \tau_\ell - \lambda^2}{\lambda(\lambda - 1)}.
\end{aligned}
$$

Since $\mathrm{OPT}(\mathcal{I}) = n$, we obtain the following lower bound on the price of anarchy:

$$
\frac{\lambda n}{\frac{\sum_{\ell \in [\lambda]} \tau_\ell}{\lambda} n - \frac{\lambda^2 - \sum_{\ell \in [\lambda]} \tau_\ell}{\lambda - 1}}.
$$

Furthermore, since $\tau_\ell < 2\tau$ for every $\ell \in \{2, \dots, \lambda - 1\}$ and $\tau = \tau_1 = \tau_\lambda$, it holds that $\sum_{\ell \in [\lambda]} \tau_\ell \leq 2(\lambda - 1)\tau$. Hence, we have a lower bound of

$$
\frac{\lambda n}{\frac{\sum_{\ell \in [\lambda]} \tau_\ell}{\lambda} n - \frac{\lambda^2 - \sum_{\ell \in [\lambda]} \tau_\ell}{\lambda - 1}} \geq \frac{\lambda n}{\frac{2(\lambda - 1)\tau}{\lambda} n - \frac{\lambda^2}{\lambda - 1} + 2\tau}. \quad \square
$$

From Theorems 8 and 9 we obtain an asymptotically tight bound for general λ-TS games.

Corollary 10. *The price of anarchy in λ-TS games is $\Theta(\lambda/\tau)$.*

Theorems 8 and 9 allow us to provide concrete bounds for subclasses of λ-TS games. In particular, for λ-ZTS games (the class of balanced jump Schelling games first studied by Elkind et al. [18]), we have that $\tau_\ell = 1$ for every $\ell \in [\lambda]$, and thus $\sum_{\ell\in[\lambda]} \tau_\ell = \lambda$. Therefore, Theorem 8 gives us an upper bound of $\frac{\lambda n}{n-\lambda}$ (which was also previously implicitly shown by Elkind et al. [18]) and Theorem 9 gives us a corresponding matching lower bound; the latter improves upon the weaker lower bound of $\frac{n}{\frac{n}{\lambda}-1+x}$ with $x > 0$ shown in [18,2].[3]

Corollary 11. *The price of anarchy of λ-ZTS games is $\frac{\lambda n}{n-\lambda}$.*

We now define the following two natural classes of λ-TS games in which the tolerance parameters are specific functions of the distance between the types. In the first one, the difference of the tolerance level is proportional to the type distance, while in the other, the difference of the tolerance is decreasing in the type distance in an inversely proportional way.

- *Proportional* λ-TS games: $t_d = 1 - \frac{d}{\lambda-1}$ for each $d \in \{0, \dots, \lambda-1\}$, while $\tau = \sum_{\ell\in[\lambda]} \frac{\ell-1}{\lambda-1} = \frac{\lambda}{2}$.
- *Inversely proportional* λ-TS games: $t_d = \frac{1}{d+1}$ for every $d \in \{0, \dots, \lambda-1\}$. We have $\tau = \sum_{\ell\in[\lambda]} \frac{1}{\ell} = H_\lambda$, where H_λ is the λ-th harmonic number.

By Theorems 8, 9 and the above definitions, we obtain the following corollaries.

Corollary 12. *For every $\lambda \geq 2$, the price of anarchy of proportional λ-TS games is at most $\frac{2n}{n-2}$ and at least $\frac{\lambda n}{(\lambda-1)n - \frac{\lambda}{\lambda-1}}$.*

Corollary 13. *For every $\lambda \geq 2$, the price of anarchy of inversely proportional λ-TS games is at most $\frac{\lambda n}{H_\lambda n - \lambda}$ and at least $\frac{\lambda n}{\frac{2(\lambda-1)}{\lambda} H_\lambda n - \frac{\lambda^2}{\lambda-1} + 2H_\lambda}$.*

We conclude our technical contribution with a lower bound on the price of stability for the case of two types of agents. For 2-ZTS games (balanced jump Schelling games), the following lower bound improves upon the bound of 34/33 of Elkind et al. [18], and is also tight when the number of agents tends to infinity because of the upper bound implied by Theorem 8; recall that $\tau = 1$ for λ-ZTS games.

Theorem 14. *The price of stability of 2-TS games is at least $2/\tau - \epsilon$, for any $\epsilon > 0$.*

Proof. We consider a particular 2-TS game $(N, G, \mathbf{t_2} = (1, 0))$ and we will show that it admits a unique (up to symmetry) equilibrium $\mathbf{v}$. Furthermore, we will show that $\mathbf{v}$ has no isolated agents. By Theorem 1, the 2-TS game $(N, G, \mathbf{t_2}' = (1, t_1 < 1))$ also admits $\mathbf{v}$ as its unique equilibrium. It will then suffice to argue about the social welfare in $\mathbf{v}$. Clearly, if $t_1 = 1$ then $\tau = 2$ and the theorem holds trivially.

Let b be an arbitrarily large even positive integer that is not a multiple of 3 and set $z = 2b + 1$ and $c = bz = 2b^2 + b$. In our proof we exploit that $\gcd(cz/2 + c, b + 1) = \gcd(2b^3 + 4b^2 + 3b/2, b + 1) = 1$.[4]

Let $n = |N| = 2c(z + 1)$ and consider the following topology G with $n + 1$ nodes; see also Fig. 7. The set of nodes comprises six sets I, J, K, X, Y, and S. Set I contains $cz/2$ nodes, set J contains c nodes, while K is a clique of $cz/2$ nodes. Set X is a collection of z subsets X_1, $X_2 \dots$, X_z, where each subset has b nodes, while set Y is a collection of z subsets Y_1, $Y_2, \dots$, Y_z where each subset contains c nodes. Finally, set S contains a single node s.

Apart from the edges in the clique K, the following edges also exist. To begin with, each node in I is connected to a unique distinct node in K. Each node in K is also connected to all nodes in X, while each node in X_i is also connected to all nodes in Y_i, for $i \in \{1, \dots, z\}$. Finally, node s is connected to all nodes in $Y \cup J$.

Consider the following assignment $\mathbf{v}$. All nodes in $I \cup K \cup X$ host red agents, all nodes in $S \cup J$ host blue agents, while all remaining $cz - 1$ blue agents are in Y. We claim that $\mathbf{v}$ is an equilibrium. To see why, observe that all agents in $I \cup K \cup S \cup J$ have utility 1 and, hence, no incentive to jump to the empty node. Each (blue) agent in Y has utility $\frac{1}{b+1}$ and would obtain the same utility by jumping. Finally, each (red) agent in X has utility at least $\frac{z}{z+2}$ and would obtain utility at most $\frac{b}{b+1}$ by jumping; since $z > 2b$, $\mathbf{v}$ is indeed an equilibrium.

[3] In the conference version of their paper, Elkind et al. [18] showed a lower bound of $\frac{n}{\frac{n}{\lambda}-1+\lambda}$. In the journal version [2], they used a construction with just two agents per type, which can be extended in several ways to n/λ agents per type, all of which lead to a lower bound of $\frac{n}{\frac{n}{\lambda}-1+x}$ for some $x > 0$. Here, we were able to obtain an instance with $x = 0$, thus obtaining a tight bound.

[4] Recall that b is even and note that $2(2b^3 + 4b^2 + 3b/2) - (4b^2 + 4b - 1)(b + 1) = 1$. The claim follows since the greatest common divisor of two integers equals their smallest positive linear combination.

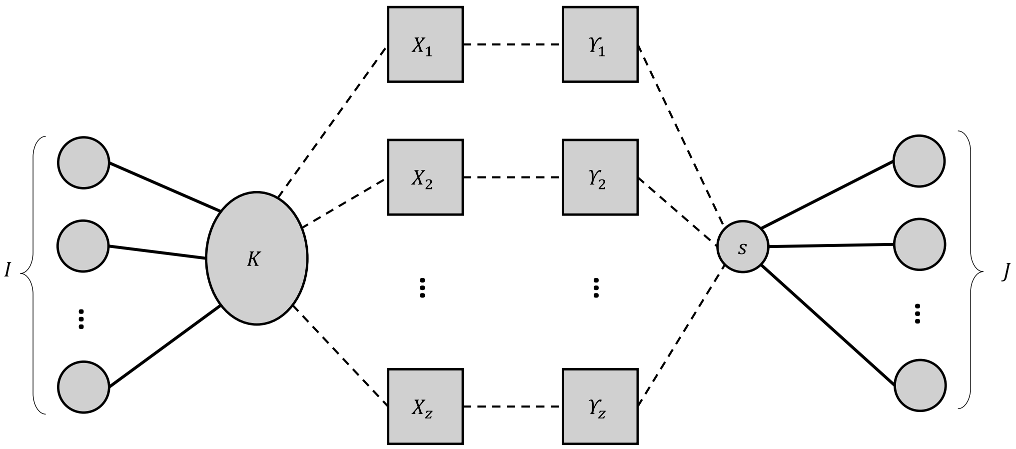

Fig. 7. The topology G used in the proof of Theorem 14. Small circles correspond to nodes, the oval is the clique K while rectangles correspond to sets of nodes. Dashed edges connecting two shapes imply that each node of one shape is connected to all nodes of the other shape.

Next, we show that $\mathbf{v}$ is the unique equilibrium assignment. This will be accomplished through a series of claims regarding the structure of any equilibrium assignment and the location of the unique empty node. In the following, let r_K and b_K denote the number of red and blue agents in K, r_{x_j} and b_{x_j} the number of red and blue agents in X_j for $j \in \{1, \dots, z\}$, and, similarly, r_{y_j} and b_{y_j} the number of red and blue agents in Y_j where $j \in \{1, \dots, z\}$. Finally, let r_X and b_X be the total number of red and blue agents in X.

Claim 15. *In any equilibrium, there are at least two red and at least two blue agents with utility less than* 1.

Proof. Observe that if $K \cup X$ contains at least two red and two blue agents, then the claim trivially holds. So, without loss of generality, let $K \cup X$ contain at most one red agent. Then, Y must contain at least two red agents as the $cz/2 + c + 1$ nodes in $I \cup S \cup J$ cannot host all $cz + c - 1$ remaining red agents. The claim follows by considering these red agents in Y and their blue neighbors in X. □

Claim 16. *In any equilibrium, the empty node cannot be in $I \cup J$.*

Proof. Assume otherwise. Then, any agent having the same type as the unique neighbor of the empty node would obtain utility 1 by jumping. Claim 15 guarantees that at least one such agent, other than the unique neighbor, exists. □

Claim 17. *In any equilibrium, each agent in J has utility* 1.

Proof. Assume otherwise and let $i \in J$ be an agent with utility 0; without loss of generality let us assume that i is red. Since i has no incentive to jump, it must be that all agents neighboring the empty node, except possibly for i, are blue. If i is not a neighbor of the empty node, by Claim 15, there exists at least one blue agent that has incentive to jump to the empty node and obtain utility 1. So, we assume that s is the empty node. In that case, all agents in J have utility 0 and it is not hard to see that such an assignment cannot be an equilibrium, since J contains either at least two red agents or at least two blue agents. □

This implies the following.

Claim 18. *In any equilibrium, s cannot be empty and all agents in $S \cup J$ are of the same type.*

Claim 19. *In any equilibrium, each agent in I has utility* 1.

Proof. Assume otherwise and let $i \in I$ be an agent with utility 0; without loss of generality let us assume that i is red. As in the proof of Claim 17, since i has no incentive to jump, it must be that all agents neighboring the empty node, except possibly for i, are blue. If i is not a neighbor of the empty node, by Claim 15, there exists at least one blue agent that has incentive to jump to the empty node and obtain utility 1. Otherwise, if the empty node is in K, then all agents in $K \cup X$ are blue. Furthermore, any other agent in I except for i is blue, as otherwise the agent would jump to the empty node and improve her utility from 0 to strictly positive. We conclude that $I \cup K \cup X$ contains $c(z+1) - 2$ blue agents, while the last two blue agents must be in Y, by Claim 18. The claim follows since each of these two blue agents has utility $\frac{b}{b+1}$ and would obtain utility $\frac{cz/2-1+c}{cz/2+c}$ by jumping; note that $cz/2-1+c>b$. □

By Claim 18, we assume, without loss of generality, that all agents in $S \cup J$ are blue. This, together with Claim 19, leads to:

$$2r_K + r_X + r_Y = c(z+1), \tag{1}$$

and

$$2b_K + b_X + b_Y = cz-1. \tag{2}$$

The following claim relies on more complex arguments about almost the entire topology.

Claim 20. *In any equilibrium, the empty node cannot be in X.*

Proof. Assume otherwise that the empty node is in X_1, without loss of generality. We begin by considering why agents in $Y \setminus Y_1$ have no incentive to jump. Let i be an agent in Y_j, where $j \in \{2,\dots,z\}$ and observe that $r_K + b_K = cz/2$, $r_{x_j} + b_{x_j} = b$, and $r_{y_1} + b_{y_1} = c$.

If agent i is red, her utility is $\frac{r_{x_j}}{b+1}$, as s hosts a blue agent, while by jumping agent i would get utility $\frac{r_K+r_{y_1}}{cz/2+c}$. Hence, since i has no incentive to jump, we obtain $\frac{r_{x_j}}{b+1} \geq \frac{r_K+r_{y_1}}{cz/2+c}$ which gives $\frac{b_{x_j}+1}{b+1} \leq \frac{b_K+b_{y_1}}{cz/2+c}$. Similarly, if agent i is blue, we obtain $\frac{b_{x_j}+1}{b+1} \geq \frac{b_K+b_{y_1}}{cz/2+c}$ and, therefore, $\frac{r_{x_j}}{b+1} \leq \frac{r_K+r_{y_1}}{cz/2+c}$.

We conclude that if Y_j contains both red and blue agents, it must hold that $\frac{r_{x_j}}{b+1} = \frac{r_K+r_{y_1}}{cz/2+c}$, and $\frac{b_{x_j}+1}{b+1} = \frac{b_K+b_{y_1}}{cz/2+c}$, which, as $\gcd(cz/2+c, b+1)=1$, is only possible when $b_{x_j}=b$, $b_K = cz/2$ and $b_{y_1}=c$. This, however, cannot hold as we already have that all $c+1$ agents in $S \cup J$ are blue and the remaining $cz-1$ blue agents cannot fill x_j, y_1, K and, due to Claim 17, I.

So, any Y_j contains either only red or only blue agents. In the following, let us assume that $Y \setminus Y_1$ contains ψ sets consisting only of blue agents and $z-1-\psi$ sets consisting only of red agents. Hence, Equation (1) becomes

$$2r_K + r_X + r_{y_1} = c(2+\psi), \tag{3}$$

while Equation (2) becomes

$$2b_K + b_X + b_{y_1} = c(z-\psi)-1. \tag{4}$$

We now argue that K must contain agents of both types. Clearly, since $S \cup J$ hosts $c+1$ blue agents, K (and, by Lemma 17 also I) cannot contain only blue agents. If K and I contain only red agents, then there is at least one red agent, let it be agent i, in Y. The utility of agent i is at most $\frac{b}{b+1}$, as s hosts a blue agent, while by jumping agent i obtains utility at least $\frac{cz/2}{cz/2+c} > \frac{b}{b+1}$, since $z > 2b$; hence, K cannot contain only red agents.

Consider a red agent i in K; her utility is $\frac{r_K+r_X}{cz/2+c-1}$, while by jumping i would obtain utility $\frac{r_K+r_{y_1}-1}{cz/2+c-1}$. Therefore, we obtain that $r_X \geq r_{y_1}-1$. Similarly, since no blue agent in K wishes to jump, we obtain that $b_X \geq b_{y_1}-1$. Since $r_X + b_X = c-1$ and $r_{y_1}+b_{y_1}=c$, we also have that $r_X \leq r_{y_1}$ and $b_X \leq b_{y_1}$, and, in particular, either $r_X = r_{y_1}$ and $b_X = b_{y_1}-1$ or $r_X = r_{y_1}-1$ and $b_X = b_{y_1}$.

We argue that it must be the case $r_X = r_{y_1}$ and $b_X = b_{y_1}-1$. Indeed, since in $S \cup J \cup \{Y \setminus Y_1\}$ we have already allocated an odd number, i.e., $c\psi + c + 1$, of blue agents, and $I \cup K$ must contain an even number of blue agents, an odd number remains to be allocated to $X \cup Y_1$. Equation (3) becomes

$$r_K + r_{y_1} = \frac{c(2+\psi)}{2}, \tag{5}$$

while Equation (4) becomes

$$b_K + b_{y_1} = \frac{c(z-\psi)}{2}. \tag{6}$$

Since $b_X = b_{y_1}-1$, we have that $b_{y_1} > 0$, while we argue that $r_{y_1} > 0$, as otherwise, since $r_X = r_{y_1}$, any red agent in Y has utility 0 and would prefer to jump; such a red agent exists in Y as, since $S \cup J$ hosts $c+1$ blue agents, it cannot be that all $c(z+1)-1$ agents in $X \cup Y$ are also blue.

Since both $r_{y_1} > 0$ and $b_{y_1} > 0$, we consider why agents in Y_1 do not wish to jump. Since no red agent in Y_1 benefits by jumping, we obtain

$$\frac{r_{x_1}}{b} \geq \frac{r_K + r_{y_1} - 1}{cz/2 + c - 1} \tag{7}$$

and, since also no blue agent in Y_1 wishes to jump,

$$\frac{b_{x_1} + 1}{b} \geq \frac{b_K + b_{y_1} - 1}{cz/2 + c - 1}. \tag{8}$$

By combining Equations (5)-(8) and, by substituting c and z, we obtain

$$\begin{aligned} r_{x_1} &\geq \frac{2(\psi+2)b^3 + (\psi+2)b^2 - 2b}{4b^3 + 8b^2 + 3b - 2} \\ &= 1 + \psi/2 - \frac{(6+3\psi)b^2 + (5+3\psi/2)b - \psi - 2}{4b^3 + 8b^2 + 3b - 2}, \end{aligned} \tag{9}$$

and

$$\begin{aligned} b_{x_1} + 1 &\geq \frac{4b^4 + (4-2\psi)b^3 + (1-\psi)b^2 - 2b}{4b^3 + 8b^2 + 3b - 2} \\ &= b - 1 - \psi/2 + \frac{(6+3\psi)b^2 + (3+3\psi/2)b - \psi - 2}{4b^3 + 8b^2 + 3b - 2}. \end{aligned} \tag{10}$$

Recall that, since the empty node is in X_1, it must be $r_{x_1} + b_{x_1} = b - 1$. Furthermore, observe that the right-hand-side terms in the two inequalities above are not integers, for any value of $\psi \in \{0, \dots, 2b\}$. The claim follows by observing that, since r_{x_1} and b_{x_1} are integers and b is not a multiple of 3, it holds that

$$\lceil \psi/2 - \frac{(6+3\psi)b^2 + (5+3\psi/2)b - \psi - 2}{4b^3 + 8b^2 + 3b - 2} \rceil + \lceil -\psi/2 + \frac{(6+3\psi)b^2 + (3+3\psi/2)b - \psi - 2}{4b^3 + 8b^2 + 3b - 2} \rceil = 1$$

for any $\psi \in \{0, \dots, 2b\}$.[5] Hence, by Equations (9) and (10), we obtain $r_{x_1} + b_{x_1} \geq b$; a contradiction. □

By the claims above, we conclude that the empty node is in Y; without loss of generality, let the empty node be in Y_1. We now show that any equilibrium assignment $\mathbf{v}'$ is identical (up to symmetry) to $\mathbf{v}$. Recall that $r_K + b_K = cz/2$, $r_X + b_X = c$, and $r_{x_1} + b_{x_1} = b$.

We first show that all agents in K are red, by exploiting that agents in K have no incentive to jump to the empty node. Let $i \in K$ be a red agent (if one exists in K); her utility is $u_i(\mathbf{v}') = \frac{r_K + r_X}{cz/2+c}$. Since i has no incentive to jump, we obtain $\frac{r_K + r_X}{cz/2+c} \geq \frac{r_{x_1}}{b+1}$, and, therefore, $\frac{b_K + b_X}{cz/2+c} \leq \frac{b_{x_1}+1}{b+1}$. Similarly, for any blue agent $j \in K$ (again, assuming one exists) we obtain that $\frac{b_K + b_X}{cz/2+c} \geq \frac{b_{x_1}+1}{b+1}$, and, therefore, $\frac{r_K + r_X}{cz/2+c} \leq \frac{r_{x_1}}{b+1}$. By these four inequalities above, we conclude that $\frac{r_K + r_X}{cz/2+c} = \frac{r_{x_1}}{b+1}$ and $\frac{b_K + b_X}{cz/2+c} = \frac{b_{x_1}+1}{b+1}$. Since $\gcd(cz/2 + c, b + 1) = 1$, this implies that all agents in K are of the same type. Since each type has $c(z + 1)$ agents and all $c + 1$ agents in $S \cup J$ are blue, and as by Claim 17 all agents in I are of the same type as those in K, we obtain that all $cz/2$ agents in K must be red.

We now show that all agents in X are also red. Assume otherwise and consider a blue agent $j \in X$. Her utility is at most $\frac{2}{z+2}$, while by jumping j would obtain utility at least $\frac{1}{b+1}$ as s is blue. Since $z > 2b$, j has an incentive to jump and, therefore, no equilibrium has blue agents in X. We have established that all agents in $I \cup K \cup X$ are red and, since there are $c(z + 1)$ agents of each type, all remaining agents are blue.

So far, we have completed the argument that $\mathbf{v}$ is the unique Nash equilibrium, up to symmetry. Under the tolerance vector $\mathbf{t_2}' = (1, t_1 < 1)$, its social welfare is

$$\begin{aligned} \text{SW}(\mathbf{v}) &= cz + b(z-1)\frac{cz/2 + t_1 c}{cz/2 + c} + b\frac{cz/2 + t_1(c-1)}{cz/2 + c - 1} + (cz-1)\frac{1 + t_1 b}{b+1} + c + 1 \\ &< cz\frac{b(1+t_1) + 2}{b+1} + 2c + 1. \end{aligned}$$

Consider the following assignment $\mathbf{v}^*$. All nodes in $I \cup K \cup J$ host red agents, all nodes in $Y \cup S$ host blue agents, while the remaining $c - 1$ blue agents are in nodes in X. Its social welfare is

$$\text{SW}(\mathbf{v}^*) = cz/2(1 + \frac{cz/2 + t_1(c-1)}{cz/2 + c - 1}) + (c-1)\frac{c + t_1 cz/2}{cz/2 + c} + cz + \frac{cz}{cz + c}$$

[5] Note that this does not hold when b is a multiple of 3 and ψ is either $2b/3 - 1$ or $4b/3$.

$$> cz\frac{z+1+t_1}{z+2} + cz$$
$$= cz\frac{2z+3+t_1}{z+2},$$

where, in the first equality, the first term is the aggregate utility of agents in $I \cup K$, the second term is the aggregate utility of agents in X, the third term is due to agents in Y, and the last term is due to agent s. Clearly, for the optimal assignment it holds that $\text{OPT} \geq \text{SW}(\mathbf{v}^*)$.

Since we have selected b to be arbitrarily large and $z = 2b+1$, it holds that

$$\frac{\text{OPT}}{\text{SW}(\mathbf{v})} \geq \frac{cz\frac{2z+3+t_1}{z+2}}{cz\frac{b(1+t_1)+2}{b+1} + 2c + 1}$$
$$\geq \frac{2}{1+t_1} - \epsilon,$$

for $\epsilon > 0$, and the theorem follows. □

5. Open problems

The most important question that our work leaves open is the characterization of games for which equilibria always exist. As this is a quite general and challenging direction, one could start with games that exhibit some structure in terms of the topology or the tolerance vector. For instance, do equilibria exist when the topology is a grid (4-grid or 8-grid) or a regular graph, for *every* tolerance vector?

The tolerance model we defined in this paper depends on a given ordering of the types and the tolerance parameters are symmetric. While this model captures certain interesting settings, there are multiple ways in which it can be generalized. For example, the tolerance parameters do not need to be symmetric and a different tolerance vector could be defined per type. Taking this further, the tolerance between types does not need to depend on an ordering of the types. Instead, one could define a weighted, directed *tolerance graph* that is defined over the different types such that the edge weights indicate the tolerance of a type towards another type; our ordered model can be thought of as the special case with an undirected tolerance line graph. In fact, one could further generalize this idea by considering scenarios in which there are no types of agents at all, but rather the agents are connected to each other via a complete weighted social network, with the different weights indicating tolerance levels. This is essentially a generalization of the class of social Schelling games proposed by Elkind et al. [18], and is inspired by fractional hedonic games [4]. All these questions would also be very interesting in the case where the agents can pairwise swap positions rather than jump to empty nodes.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] Aishwarya Agarwal, Edith Elkind, Jiarui Gan, Alexandros A. Voudouris, Swap stability in Schelling games on graphs, in: Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI), 2020.
[2] Aishwarya Agarwal, Edith Elkind, Jiarui Gan, Ayumi Igarashi, Warut Suksompong, Alexandros A. Voudouris, Schelling games on graphs, Artif. Intell. 301 (2021) 103576.
[3] Elliot Anshelevich, Anirban Dasgupta, Jon M. Kleinberg, Éva Tardos, Tom Wexler, Tim Roughgarden, The price of stability for network design with fair cost allocation, SIAM J. Comput. 38 (4) (2008) 1602–1623.
[4] Haris Aziz, Florian Brandl, Felix Brandt, Paul Harrenstein, Martin Olsen, Dominik Peters, Fractional hedonic games, ACM Trans. Econ. Comput. 7 (2) (2019) 6:1–6:29.
[5] George Barmpalias, Richard Elwes, Andrew Lewis-Pye, Digital morphogenesis via Schelling segregation, in: Proceedings of the 55th IEEE Annual Symposium on Foundations of Computer Science (FOCS), 2014, pp. 156–165.
[6] Patrick Bayer, Robert McMillan, Kim Rueben, The causes and consequences of residential segregation: An equilibrium analysis of neighborhood sorting, 2001.
[7] Prateek Bhakta, Sarah Miracle, Dana Randall, Clustering and mixing times for segregation models on $\mathbb{Z}^2$, in: Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 2014, pp. 327–340.
[8] Davide Bilò, Vittorio Bilò, Pascal Lenzner, Louise Molitor, Tolerance is necessary for stability: single-peaked swap Schelling games, in: Proceedings of the 31st International Joint Conference on Artificial Intelligence (IJCAI), 2022, pp. 81–87.
[9] Davide Bilò, Vittorio Bilò, Pascal Lenzner, Louise Molitor, Topological influence and locality in swap Schelling games, Auton. Agents Multi-Agent Syst. 36 (2) (2022) 47.

[10] Thomas Bläsius, Tobias Friedrich, Martin S. Krejca, Louise Molitor, The impact of geometry on monochrome regions in the flip Schelling process, in: Proceedings of the 32nd International Symposium on Algorithms and Computation (ISAAC), 2021, pp. 29:1–29:17.
[11] Christina Brandt, Nicole Immorlica, Gautam Kamath, Robert Kleinberg, An analysis of one-dimensional Schelling segregation, in: Proceedings of the 44th Symposium on Theory of Computing (STOC), 2012, pp. 789–804.
[12] Martin Bullinger, Warut Suksompong, Alexandros A. Voudouris, Welfare guarantees in Schelling segregation, J. Artif. Intell. Res. 71 (2021) 143–174.
[13] Hau Chan, Mohammad T. Irfan, Cuong Viet Than, Schelling models with localized social influence: a game-theoretic framework, in: Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS), 2020, pp. 240–248.
[14] Ankit Chauhan, Pascal Lenzner, Louise Molitor, Schelling segregation with strategic agents, in: Proceedings of the 11th International Symposium on Algorithmic Game Theory (SAGT), 2018, pp. 137–149.
[15] William Clark, Mark Fossett, Understanding the social context of the Schelling segregation model, Proc. Natl. Acad. Sci. 105 (11) (2008) 4109–4114.
[16] Argyrios Deligkas, Eduard Eiben, Tiger-Lily Goldsmith, The parameterized complexity of welfare guarantees in Schelling segregation, CoRR, arXiv:2201.06904, 2022.
[17] Hagen Echzell, Tobias Friedrich, Pascal Lenzner, Louise Molitor, Marcus Pappik, Friedrich Schöne, Fabian Sommer, David Stangl, Convergence and hardness of strategic Schelling segregation, in: Proceedings of the 15th International Conference on Web and Internet Economics (WINE), 2019, pp. 156–170.
[18] Edith Elkind, Jiarui Gan, Ayumi Igarashi, Warut Suksompong, Alexandros A. Voudouris, Schelling games on graphs, in: Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI), 2019, pp. 266–272.
[19] Tobias Friedrich, Pascal Lenzner, Louise Molitor, Lars Seifert, Single-peaked jump Schelling games, CoRR, arXiv:2302.12107, 2023.
[20] Nicole Immorlica, Robert Kleinberg, Brendan Lucier, Morteza Zadimoghaddam, Exponential segregation in a two-dimensional Schelling model with tolerant individuals, in: Proceedings of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 2017, pp. 984–993.
[21] Panagiotis Kanellopoulos, Maria Kyropoulou, Alexandros A. Voudouris, Modified Schelling games, Theor. Comput. Sci. 880 (2021) 1–19.
[22] Elias Koutsoupias, Christos H. Papadimitriou, Worst-case equilibria, in: Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS), 1999, pp. 404–413.
[23] Luca Kreisel, Niclas Boehmer, Vincent Froese, Rolf Niedermeier, Equilibria in Schelling games: computational hardness and robustness, in: Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS), 2022, pp. 761–769.
[24] Romans Pancs, Nicolaas J. Vriend, Schelling's spatial proximity model of segregation revisited, J. Public Econ. 91 (1–2) (2007) 1–24.
[25] Thomas C. Schelling, Models of segregation, Am. Econ. Rev. 59 (2) (1969) 488–493.
[26] Thomas C. Schelling, Dynamic models of segregation, J. Math. Sociol. 1 (2) (1971) 143–186.
[27] Charles M. Tiebout, A pure theory of local expenditures, J. Polit. Econ. 64 (5) (1956) 416–424.
[28] Dejan Vinković, Alan Kirman, A physical analogue of the Schelling model, Proc. Natl. Acad. Sci. 103 (51) (2006) 19261–19265.
[29] Junfu Zhang, Residential segregation in an all-integrationist world, J. Econ. Behav. Organ. 54 (4) (2004) 533–550.