Not All Strangers Are the Same: The Impact of Tolerance in Schelling Games

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

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

Alexandros A. Voudouris
School of Computer Science and Electronic Engineering, University of Essex, UK

Abstract

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.

2012 ACM Subject Classification Theory of computation → Algorithmic game theory and mechanism design

Keywords and phrases Schelling games, Equilibria, Price of anarchy, Price of stability

Digital Object Identifier 10.4230/LIPIcs.MFCS.2022.60

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 [5] 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 [23]). The most prominent of those models is that of Schelling [21, 22], which studies how motives at an individual level can lead to macroscopic behavior and, ultimately, to segregation. In particular, the individuals are modelled 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 et al. [16] 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. [15] 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. [16] 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, 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 [19] (Theorems 8 and 9) and price of stability [2] (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 [13], Economics [20, 25], Physics [24], and Computer Science [4, 6, 8, 9, 17].

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. [12] 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. [15]. In this paper we extend the utility model of Elkind et al. [16], who initially refined the cost model of Chauhan et al. [12]. They introduced a simpler utility function (fraction of same-type agents in the 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. They also proposed many interesting variants, such as *enemy aversion* (agents might prefer being alone to being in a group full of agent 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. [18] and Chan et al. [11], respectively. Agarwal et al. [1] studied similar existence, complexity and qualitative questions for swap games. Bilò et al. [7] 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. Finally, Bullinger et al. [10] and Deligkas et al. [14] 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 Schelling games studied in the related literature (e.g., see [16]), 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. Balanced games are the most fundamental ones that admit non-trivial and interesting results[1]. 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 $\mathcal{I} = (N, G)$.

[1] Note that equilibria are not guaranteed to exist even for balanced games (see Theorem 2), while it is not hard to observe that the price of anarchy can be unbounded when the sizes of the types are arbitrary [16].

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:

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

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 v of the topology. That is,

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

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

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

Therefore, agent i has no incentive to jump to the empty node v 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 $\mathbf{v}'$:

- In $\mathbf{v}$, an 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 v 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 v as both nodes give her utility 1/2. On the other hand, in $\mathcal{I}'$, agent i has utility $(1+t_2)/2 = 1/2$ and has incentive to jump to v to increase her utility to $(1+t_1)/2 = 3/4$. 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 v 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 v 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 $(1 + t_1)/3 = 1/2$ by occupying node v_i, which is exactly the utility she would also obtain by jumping to v. 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 [16], 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 et al. [16] 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.*

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.

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.*

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 $\lambda \geq 3$ 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.*

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 $m \leq M$. Let $x = n/\lambda \geq 2$ 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 1, which in turn relies on the Tile procedure described in Algorithm 2. In particular, Algorithm 2 considers the yet unassigned agents in increasing type according to the ordering $\succ$, and assigns them in an $r \times M$ subgrid having row s as the top row, so that the k leftmost nodes of the top row are left empty, while all other nodes host an agent (assuming the number of unassigned agents is large enough). Tile visits these rows in a column-major order, skipping the empty nodes. Informally, Algorithm 1 repeatedly calls Algorithm 2 to compute an assignment for consecutive sub-grids, along the largest dimension of the topology. The exact size of the sub-grid considered at each time is determined by the number of remaining rows of the topology.

Algorithm 1 terminates immediately (at any step) when all agents have been assigned. First, observe that if it terminates in Lines 3 or 7, each agent of type ℓ has neighbors of types in $\{\ell-1, \ell, \ell+1\}$, and, hence, $\mathbf{v}$ is an equilibrium. Note that the algorithm cannot terminate at Line 11 since $e < M$, so let us assume that the algorithm terminates in Line 14. 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.

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

Algorithm 1 Equilibrium construction for a 2-binary λ-TS game on an $m \times M$ grid.

/* x: number of agents per type */
/* e: number of empty nodes */
/* The algorithm terminates immediately when all agents have been assigned. */
1 Initialize $k = 0$
2 **while** $x \leq m - k$ and $e \geq M$ **do**
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**
7   Tile$(k+1, m-k, 0)$
8 **else** /* In this case it holds that $e < M$ and $x \leq m - k$ */
9   Define non-negative integers $\alpha \in \mathbb{N}_{>0}$ and $\beta \leq x - 1$ such that $m - k = \alpha x + \beta$
10   **for** $i = 1, \ldots, \alpha - 1$ **do**
11   Tile$(k+1, x, 0)$
12   update $k := k + x$
13   **if** $\beta = 0$ **then**
14   Tile$(k+1, x, e)$
15   **else if** $\beta = 1$ **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$(k+x+1, \beta, e)$

Algorithm 2 Tile(s, r, k).

/* s, r: starting row and number of rows defining an $r \times M$ grid */
/* k: number of nodes to be left empty */
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 $\succ$ at node (i, j)

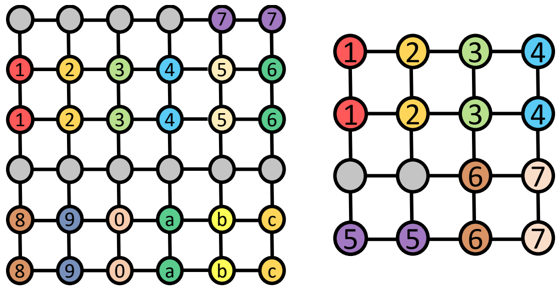

Figure 1 On the left, an example of how Algorithm 1 operates when it terminates in Line 19. On the right, an example when the algorithm terminates in Line 14. Agents of the same number and color are of the same type, while $\succ$ is $\{1, 2, \ldots, 9, 0, a, b, c\}$.

If the algorithm terminates in Line 19 (see also the leftmost part of Figure 1), 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, 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 1 may fail to return an equilibrium for lexicographically larger tolerance vectors. Indeed, consider a 4×4 grid and 7 types of two agents each. Algorithm 1 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; see also the rightmost example in Figure 1. 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 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.

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).

/* $tree_1, \dots, tree_k$ 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 $a \leq \lceil \frac{\lambda}{2} \rceil$ 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 $b \geq \lceil \frac{\lambda+1}{2} \rceil$ 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.

▶ **Theorem 6.** *For $\lambda \geq 3$, 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.*

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

▶ **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 $\alpha \geq \lfloor \frac{\lambda}{2} \rfloor$ for $\lambda \geq 4$, 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 $x \geq 3$ 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 $k \geq 2$ 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 all the agents of $\mathcal{T}_1$ are covered by either agents of the same type or agents of type $\mathcal{T}_2$, and the assignment for the remaining agents 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.

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 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 $\lambda \geq 4$.

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.

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

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). ◀

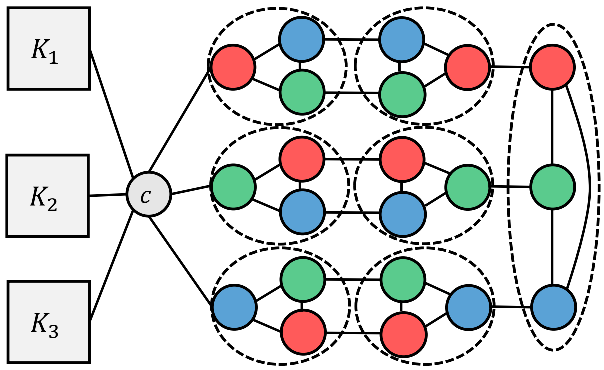

Figure 2 An instance used for the proof of Theorem 9 for the case of 3 types and 21 agents, so that each type has 7 agents. The big squares K_1, K_2, K_3 correspond to cliques of size 7 (the number of agents per type), while the ovals represent cliques of size 3 (the number of types). In an optimal assignment, 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 $(\tau_\ell - 1)/2$.

For each $\ell \in \{1, \dots, \lambda\}$, 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; see Figure 2 for a sketch of the proof for $\lambda = 3$.

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

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

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 $\Theta(\lambda/\tau)$.*

Theorem 9 allows us to provide concrete bounds for subclasses of λ-TS games. In particular, for λ-ZTS games, since $\tau_\ell = 1$ for every $\ell \in [\lambda]$, we have $\sum_{\ell\in[\lambda]}\tau_\ell = \lambda$, and thus the left-hand-side of the inequality in Theorem 9 allows us to improve upon the weaker lower of [16] and obtain the following tight bound, for any values of n and λ.

▶ **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, the following lower bound improves upon the bound of 34/33 of Elkind et al. [16], 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 $\epsilon > 0$.*

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. [16], and is inspired by fractional hedonic games [3].

References

1 Aishwarya Agarwal, Edith Elkind, Jiarui Gan, and Alexandros A. Voudouris. Swap stability in Schelling games on graphs. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

2 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.

3 Haris Aziz, Florian Brandl, Felix Brandt, Paul Harrenstein, Martin Olsen, and Dominik Peters. Fractional hedonic games. *ACM Transactions on Economics and Computation*, 7(2):6:1–6:29, 2019.

4 George Barmpalias, Richard Elwes, and Andrew Lewis-Pye. Digital morphogenesis via Schelling segregation. In *Proceedings of the 55th IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, pages 156–165, 2014.

5 Patrick Bayer, Robert McMillan, and Kim Rueben. The causes and consequences of residential segregation: An equilibrium analysis of neighborhood sorting, 2001.

6 Prateek Bhakta, Sarah Miracle, and 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)*, pages 327–340, 2014.

7 Davide Bilò, Vittorio Bilò, Pascal Lenzner, and Louise Molitor. Topological influence and locality in swap Schelling games. In *Proceedings of the 45th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, pages 15:1–15:15, 2020.

8 Thomas Bläsius, Tobias Friedrich, Martin S. Krejca, and 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)*, pages 29:1–29:17, 2021.

9 Christina Brandt, Nicole Immorlica, Gautam Kamath, and Robert Kleinberg. An analysis of one-dimensional Schelling segregation. In *Proceedings of the 44th Symposium on Theory of Computing (STOC)*, pages 789–804, 2012.

10 Martin Bullinger, Warut Suksompong, and Alexandros A. Voudouris. Welfare guarantees in Schelling segregation. *Journal of Artificial Intelligence Research*, 71:143–174, 2021.

11 Hau Chan, Mohammad T. Irfan, and 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)*, pages 240–248, 2020.

12 Ankit Chauhan, Pascal Lenzner, and Louise Molitor. Schelling segregation with strategic agents. In *Proceedings of the 11th International Symposium on Algorithmic Game Theory (SAGT)*, pages 137–149, 2018.

13 William Clark and Mark Fossett. Understanding the social context of the Schelling segregation model. *Proceedings of the National Academy of Sciences*, 105(11):4109–4114, 2008.

14 Argyrios Deligkas, Eduard Eiben, and Tiger-Lily Goldsmith. The parameterized complexity of welfare guarantees in schelling segregation. *CoRR*, abs/2201.06904, 2022. arXiv:2201.06904.

15 Hagen Echzell, Tobias Friedrich, Pascal Lenzner, Louise Molitor, Marcus Pappik, Friedrich Schöne, Fabian Sommer, and David Stangl. Convergence and hardness of strategic Schelling segregation. In *Proceedings of the 15th International Conference on Web and Internet Economics (WINE)*, pages 156–170, 2019.

16 Edith Elkind, Jiarui Gan, Ayumi Igarashi, Warut Suksompong, and Alexandros A. Voudouris. Schelling games on graphs. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 266–272, 2019.

17 Nicole Immorlica, Robert Kleinberg, Brendan Lucier, and 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)*, pages 984–993, 2017.

18 Panagiotis Kanellopoulos, Maria Kyropoulou, and Alexandros A. Voudouris. Modified Schelling games. *Theoretical Computer Science*, 880:1–19, 2021.

19 Elias Koutsoupias and Christos H. Papadimitriou. Worst-case equilibria. In *Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 404–413, 1999.

20 Romans Pancs and Nicolaas J. Vriend. Schelling's spatial proximity model of segregation revisited. *Journal of Public Economics*, 91(1–2):1–24, 2007.

21 Thomas C. Schelling. Models of segregation. *American Economic Review*, 59(2):488–493, 1969.

22 Thomas C. Schelling. Dynamic models of segregation. *Journal of Mathematical Sociology*, 1(2):143–186, 1971.

23 Charles M. Tiebout. A pure theory of local expenditures. *Journal of Political Economy*, 64(5):416–424, 1956.

24 Dejan Vinković and Alan Kirman. A physical analogue of the schelling model. *Proceedings of the National Academy of Sciences*, 103(51):19261–19265, 2006.

25 Junfu Zhang. Residential segregation in an all-integrationist world. *Journal of Economic Behavior and Organization*, 54(4):533–550, 2004.