Parameterized Complexity of Hotelling-Downs with Party Nominees

Argyrios Deligkas, **Eduard Eiben**, **Tiger-Lily Goldsmith**
Royal Holloway, University of London
{Argyrios.Deligkas, Eduard.Eiben}@rhul.ac.uk, tigerlilygoldsmith@gmail.com

Abstract

We study a generalization of the Hotelling-Downs model through the lens of parameterized complexity. In this model, there is a set of voters on a line and a set of parties that compete over them. Each party has to choose a nominee from a set of candidates with predetermined positions on the line, where each candidate comes at a different cost. The goal of every party is to choose the most profitable nominee, given the nominees chosen by the rest of the parties; the profit of a party is the number of voters closer to their nominee minus its cost. We examine the complexity of deciding whether a pure Nash equilibrium exists for this model under several natural parameters: the number of different positions of the candidates, the discrepancy and the span of the nominees, and the overlap of the parties. We provide FPT and XP algorithms and we complement them with a W[1]-hardness result.

1 Introduction

The Hotelling-Downs model [Downs, 1957; Hotelling, 1929] is probably the most well-established framework in studying strategic positioning on spatial competition. In the original model, there are two ice cream sellers on a beach and they want to choose a position for their shop in order to maximize their customers, under the assumption that a customer will choose the seller that is closer to their location. The simplicity and the elegance of this framework served as a basis for numerous other models and applications ranging from spatial design, to strategic candidacy [Eiselt *et al.*, 1993; Eiselt, 2011].

In fact, in [Downs, 1957] it was argued that the equilibrium of the game above, could be potentially used to predict the positions of the parties within the political spectrum. However, the original model has two strong assumptions: there are *only* two agents and every agent can choose *every* possible location in the spectrum; the unique equilibrium of the game is when both vendors choose the median of the beach. In reality though, several political parties are involved. In addition, the available positions of a party in the political spectrum are constrained from two crucial factors: the *ideology* of the party and the *available candidates or policies*, it can choose from.

In order to bypass the aforementioned limitations, a new model that is closer to reality was recently proposed [Harrenstein *et al.*, 2021]. In this model, there are many parties, each of whom is associated with a *set of candidates* on a line. In addition, there is a set of voters distributed in an arbitrary way over the line. To play the game, every party should choose one *nominee* from their set of candidates, while again they are trying to maximize the number of voters that are closer to their nominee compared to the nominees of the remaining parties. The authors showed that these games do not always possess a pure Nash equilibrium and in fact it is NP-complete to decide if an equilibrium exists.

Indeed the extended model is closer to reality, albeit in the current political spectrum there are further extra constraints within the parties. In many cases, the ideology of a party dictates a *high level position* of the party on the spectrum, which is usually a priori fixed, while the different candidates and policy adoptions correspond to the *fine grained position* of the party. For example, it is not uncommon that the spread of the candidates of a party within the political spectrum is not too wide. Furthermore, not many parties share the same ideology; usually the parties target a specific demographic of voters within the political spectrum and the overlap of this range with the remaining parties is limited. Inspired by these constraints of real-life politics we study the following question for the generalized Hotelling-Downs model.

> *What is the complexity of equilibrium computation under natural constraints on the candidates?*

Our contribution. Our contribution is twofold: (a) we further extend the Hotelling-Downs framework of Harrenstein *et al.* [2021] by associating each candidate with a cost and (b) we study the equilibrium computation problem for the extended model through the lens of parameterized complexity. In real-life politics, many times, every position at the political spectrum comes at a different cost. In fact, it is not uncommon that even candidates on *the same position* come at a different cost for different parties. For example, this can be due to preexisting positions, investments, and advertisements of the party towards specific campaigns. For this extended model, we study the parameterized complexity of the equilibrium-computation problem under several parameters that capture natural constraints over the candidates of the parties. We denote the corresponding computational problem HOTELLINGDOWNS-NE (HD-NE). More specifically, we focus on the *overlap* the parties have due to their candidates, and the *span*, the *discrepancy*, and the number of *different positions* of the candidates. The overlap parameter intuitively captures the maximum number of parties that have overlapping ideologies, if we assume that the ideology of a party is expressed via the spread of its candidates. The span is the difference between the leftmost candidate of any party and the rightmost candidate of any party; observe that the span parameter naturally bounds the number of candidate positions. Finally, the discrepancy of the candidates of a party is the difference between the leftmost and the rightmost candidate the party has; the discrepancy parameter is the maximum discrepancy over the parties.

Firstly, we focus on the overlap parameter and we prove that in this case HD-NE admits an XP algorithm (Theorem 1). In fact, the same algorithm yields fixed-parameter tractability by overlap and discrepancy (Corollary 2). In addition, by observing that overlap is trivially bounded by the number of parties, we can get as an immediate corollary that HD-NE is fixed-parameter tractable by discrepancy and the number of parties. We show that the XP algorithm is in fact the best we can hope for since the problem is W[1]-hard when parameterized only by overlap, even when every candidate has zero cost (Theorem 5). In fact, our result shows that our XP algorithm is essentially optimal, assuming the exponential time hypothesis (ETH). This is our main technical result; our hardness reduction is rather intricate and requires a careful design of an instance that utilizes Sidon sequences [Erdös and Turán, 1941]. Next, we turn our attention to discrepancy and span parameters. In Theorem 6 we derive a fixed-parameter tractable algorithm parameterized by the number of candidate positions for all parties when every party has uniform cost over their candidates. Using this result and the fact that the span bounds the number of different positions of the candidates, we get as corollary that HD-NE is fixed-parameter tractable by span, when every party suffers uniform cost over the candidates (Corollary 7). In our last result, we show that the problem is XP by discrepancy (Theorem 8).

Further Related Work. Since the original Hotelling-Downs model was introduced, several extensions have been proposed and studied [Brusco *et al.*, 2012; Shen and Wang, 2016; Feldman *et al.*, 2016b]. Two influential studies include [Stokes, 1963] and [Eiselt *et al.*, 1993], which study Hotelling-Downs framework in spatial competition. [Sengupta and Sengupta, 2008] and [Eaton and Lipsey, 1975] study modifications of the model by considering multiple players and different voting rules, while [Ben-Porat and Tennenholtz, 2019] studies the case where each party can choose multiple positions on the line. In algorithmic game theory, our research relates to Voronoi games where every player's utility depends on the points closest to them [Ahn *et al.*, 2004]. This is a highly studied problem on graphs and [Dürr and Thang, 2007] proved that it is NP-complete to decide whether a Nash equilibrium exists on arbitrary graphs. The Hotelling-Downs model is closely related to other problems from computational social choice [Feldman *et al.*, 2016a; Faliszewski *et al.*, 2016; Ahn *et al.*, 2004; Brusco *et al.*, 2012] and strategic voting [Meir, 2018].

2 Preliminaries

The space of the game we consider is $[m] = \{1, 2, \ldots, m\}$, which we view as a path with m positions. There is a set of *voters* located on $[m]$; on position i there are $v_i \in \mathbb{N}$ voters.

Parties. There are n *parties* $\Pi_1, \ldots, \Pi_n$ that correspond to the players of the game. Every party Π_i has a set of *candidates* $C_i \subseteq [m]$, where $C_i := \{c_{i1}, \ldots, c_{ik_i}\}$ and $c_{i1} < c_{i2} < \ldots < c_{ik_i}$. In addition, every candidate is associated with a *cost* given by $p_i : C_i \to \mathbb{N}$. If $p_i(x) = a_i$ for every $x \in C_i$, then we say that the party suffers *uniform cost*. To play the game, every party Π_i has to *nominate* a *single* candidate $s_i \in C_i$; this is the *strategy* of Π_i. We denote $\mathrm{s} = (s_i, \mathrm{s}_{-i})$ as the *strategy profile* for the parties, where s_{-i} is the $(n-1)$-dimensional vector produced by s, after deleting its i-th entry.

Utilities. The *utility* of party Π_i under strategy profile (s_i, s_{-i}), is denoted $u_i(s_i, \mathrm{s}_{-i})$. At a high level, $u_i(s_i, \mathrm{s}_{-i})$ is the number of voters that are located closest to the candidate s_i chosen by party Π_i compared to the candidates s_{-i} chosen by the rest of the parties, minus the cost $p_i(s_i)$ associated for the chosen candidate. To define formally $u_i(s_i, \mathrm{s}_{-i})$ we will need the auxiliary sets $R(s_i, \mathrm{s}_{-i})$ and $T(s_i, \mathrm{s}_{-i})$. $R(s_i, \mathrm{s}_{-i})$ contains all points of $[m]$ that are closer to s_i compared to any other s_j *different than* s_i of s_{-i}; $T(s_i, \mathrm{s}_{-i})$ contains all points of $[m]$ (at most two of them) that are in equal distance to s_i and some other s_j *different than* s_i of s_{-i}. In addition, given a strategy profile s and a $j \in [m]$, define n_j to be the set of parties $i \in [n]$ such that $s_i = j$. We are now ready to formally define the utility of any party. So, the utility $u_i(s_i, \mathrm{s}_{-i})$ of party i under strategy profile (s_i, s_{-i}), where $s_i = j$, is

$$\frac{1}{|n_j|} \cdot \left(\sum_{\ell \in R(s_i, \mathrm{s}_{-i})} v_\ell + \sum_{\ell \in T(s_i, \mathrm{s}_{-i})} \frac{v_\ell}{2} \right) - p_i(s_i).$$

Strategy $s_i \in C_i$ is a *best response* against s_{-i}, if $u_i(s_i, \mathrm{s}_{-i}) \geq u_i(s'_i, \mathrm{s}_{-i})$, for every $s'_i \in C_i$. A strategy profile is a *Nash equilibrium* if every party plays a best response; in other words, in a Nash equilibrium no party can increase its utility by unilaterally changing its strategy.

Definiton 1 (HOTELLINGDOWNS-NE (HD-NE)). Input: $\mathcal{I} = \langle m, n, \mathcal{C}, \mathcal{P}, \mathcal{V} \rangle$, where $[m]$ is the space of the game, n is the number of parties, $\mathcal{C} = \{C_1, \ldots, C_n\}$ is the set of candidate sets, $\mathcal{P} = \{p_1, \ldots, p_n\}$ are the cost functions, and $\mathcal{V} = \{v_1, \ldots, v_m\}$ are the voters on every position. Question: Is there a Nash equilibrium for $\mathcal{I}$?

Parameters. We will study the complexity of HD-NE under the following parameters.

- **Parties.** This is the number of parties n.
- **Candidate-positions.** This is the number of different positions of the candidates for all the candidates of all the parties, formally $|C_1 \cup C_2 \cup \ldots \cup C_n|$.
- **Discrepancy.** This parameter captures the distance between the candidates within every party and is defined as $\max_{i \in [n]}\{c_{ik_i} - c_{i1}\}$. We denote the discrepancy of an instance $\mathcal{I}$ as $\mathsf{d}(\mathcal{I})$, or d if the instance is clear from the context.

- **Span.** This parameter captures the distance between the leftmost and the rightmost candidate, formally the span is $\max_{i\in[n]} c_{ik_i} - \min_{i\in[n]} c_{i1}$.
- **Overlap.** This parameter captures the maximum number of parties who intersect on the line. For every $j \in [m]$, define $o_j := \{i \in [n] : j \in [c_{i1}, c_{ik_i}]\}$. The overlap parameter is then defined as $\max_{j\in[m]} |o_j|$. We denote the overlap of an instance $\mathcal{I}$ as $\mathsf{ov}(\mathcal{I})$ or ov if the instance is clear from the context.

Parameterized Complexity. Parameterized complexity studies the running-time of an algorithm with respect to a parameter $k \in \mathbb{N}_0$ and input size n [Cygan *et al.*, 2015]. The most favorable complexity class is FPT, which contains all problems that can be decided by an algorithm running in time $f(k) \cdot n^{\mathcal{O}(1)}$, where f is a computable function. Algorithms with this running-time are called *fixed-parameter (FPT) algorithms*. A less favorable, but still positive, outcome is an XP *algorithm*, which has running-time $\mathcal{O}(n^{f(k)})$; problems admitting such algorithms belong to the class XP. Showing that a problem is $\mathsf{W}[1]$-hard rules out the existence of a fixed-parameter algorithm under the well-established assumption that $\mathsf{W}[1] \neq \mathsf{FPT}$.

3 Overlap

In this section we study HD-NE under the overlap parameter. We first design an XP algorithm for the problem. Our algorithm in fact allows us to derive fixed-parameter tractable algorithms as corollaries that are parameterized by overlap and discrepancy, or by discrepancy and number of parties.

Theorem 1 ($\star$). *HD-NE admits an* XP *algorithm parameterized by overlap and it runs in time* $\mathcal{O}(\mathsf{d}^{\mathcal{O}(\mathsf{ov})} \cdot \mathsf{ov}^{\mathcal{O}(\mathsf{ov})} \cdot \mathrm{poly}(|\mathcal{I}|))$*, where* $|\mathcal{I}|$ *is the size of the input.*

The algorithm is a dynamic programming along the space of the game $[m]$. For each $j \in [0, m+1]$ we will compute a table Γ_j. In order to define Γ_0 and Γ_{m+1}, we define $o_0 = o_{m+1} = \emptyset$. The table Γ_j has entries for some tuples of the form $(\mathrm{Nom}, \#\mathrm{Parties}, \mathrm{Left}, \mathrm{Right}, \mathrm{aft}, \mathrm{bef})$, where:

- Nom represents overlap nominees, that is for each party $i \in o_j$ it contains a single candidate $s_i \in C_i$ representing the strategy of the party Π_i;
- $\#\mathrm{Parties}$ contains for each $i \in o_j$ the number $n_i \in [\mathsf{ov}]$ that represents the number of parties with the strategy s_i;
- Left contains for each $i \in o_j$ the position $\ell_i \in [m] \cup \{-\infty\}$, representing the position of a closest nominee selected as a strategy to the left of s_i;
- Right contains for each $i \in o_j$ the position $r_i \in [m] \cup \{\infty\}$, representing the position of a closest nominee selected as a strategy to the right of s_i;
- $\mathrm{aft} \in [j, m] \cup \{\infty\}$ represents the position of the closest nominee selected as a strategy to the right of $j-1$ (after $j-1$);
- $\mathrm{bef} \in \{1, \ldots j\} \cup \{-\infty\}$ represents the position of the closest nominee selected as a strategy to the left of $j+1$ (before $j+1$).

We call each such tuple a *configuration* for j. We do not care about all configurations for j; we care only about those for which there exists a strategy profile s, called a *satisfiability witness*, that conforms to the intended meaning of the configuration $\sigma = (\mathrm{Nom}, \#\mathrm{Parties}, \mathrm{Left}, \mathrm{Right}, \mathrm{aft}, \mathrm{bef})$. A configuration is *satisfiable*, if it admits a satisfiability witness.

Claim 1 ($\star$). *Given a configuration* σ*, we can in polynomial time verify if there exists a satifiability witness of* σ.

Claim 2. *There are at most* $\mathcal{O}(\mathsf{d}^{3|o_j|+2} \cdot \mathsf{ov}^{|o_j|})$ *many satisfiable configurations for* j *and we can enumerate them in* $\mathcal{O}(\mathsf{d}^{3|o_j|+2} \cdot \mathsf{ov}^{|o_j|} \cdot \mathrm{poly}(|\mathcal{I}|))$ *time.*

Proof of Claim. Since for each $i \in o_j$ the party Π_i has a candidate at the position s_i, it follows that there are at most $|C_i| \le p+1$ possibilities for s_i for each $i \in s_i$. To verify that, given s_i we have at most $2p+2$ many possibilities of ℓ_i, we observe that it is either $-\infty$ or it is within p positions to the left of s_i, or there is a party $\Pi_{i'}$ such that $c_{i'k_{i'}} < s_i$ and for every party $\Pi_{i''}$ either $c_{i''k_{i''}} \le c_{i'k_{i'}}$ or $c_{i''k_{i''}} \ge s_i$. Since the configuration is satisfiable, we get that $\ell_i \ge c_{i'1}$. Moreover, by the choice of i', no party has a candidate between $c_{i'k_{i'}}$ and $s_i - p - 1$. Finally, since if $\ell_i \neq -\infty$, then at least one party has to have a candidate at the position ℓ_i, it follows that $\ell \in \{-\infty\} \cup [c_{i'1}, c_{i'k_{i'}}] \cup [s_i - p, s_i - 1]$. By an analogous argument, we can also show that there are only $2p+2$ positions to consider for r_i ($i \in o_j$), aft, and bef. $\square$

Γ_j contains an entry for each satisfiable configuration and $\Gamma_j[(\mathrm{Nom}, \#\mathrm{Parties}, \mathrm{Left}, \mathrm{Right}, \mathrm{aft})]$ is either NONE, or it is a single partial strategy profile s that contains a strategy for every party Π_i with $c_{i1} \le j$, such that s satisfies the following conditions:

1. for every $i \in o_j$, the strategy of Π_i is s_i;
2. for every $i \in o_j$, if $s_i \le j$, then the number of parties $\Pi_{i'}$ such that the strategy of $\Pi_{i'}$ is s_i is exactly n_i;
3. for every $i \in o_j$, no party has its strategy between ℓ_i and s_i, moreover if $\ell_i \le j$, then there exists a party with strategy ℓ_i in s;
4. for every $i \in o_j$, no party has its strategy between s_i and r_i, moreover if $r_i \le j$, then there exists a party with strategy r_i in s.
5. for every $s_i \in \mathsf{s}$, if $s_i \le j$, then $s_i \le \mathrm{bef}$, moreover there exists a party $\Pi_{i'}$ with strategy bef in s;
6. for every party Π_i and every $s'_i \in C_i \cap [j]$ it holds $u_i(s_i, \mathsf{s}_{-i}) \ge u_i(s'_i, \mathsf{s}_{-i})$, where the value of u_i is computed under the assumption that besides the parties with a strategy in s there are additional parties with strategy aft and the strategies ℓ_i and r_i for $i \in o_j$.

It is easy to see that Γ_{m+1} contains at most $m+1$ entries, one for each satisfiable configuration $\sigma_{\mathrm{bef}} = (\emptyset, \emptyset, \emptyset, \emptyset, \infty, \mathrm{bef})$, $\mathrm{bef} \in [m] \cup \{-\infty\}$, and any Nash equilibrium s for the original instance satisfies all the conditions for $\Gamma_{m+1}[\sigma_{\mathrm{bef}}]$, for some satisfiable configuration σ_{bef}. Finally, if $\Gamma_{m+1}[\sigma_{\mathrm{bef}}]$ is not NONE, then it contains a strategy profile s that has a strategy for every party in $[n]$, which is a satisfiability witness for σ_{bef}. Moreover, from the fact that

$o_{m+1} = \emptyset$ and Condition 6, it follows that s is a Nash equilibrium. It remains to show that we can correctly compute values for all configurations with a satisfiability witness and all tables $\Gamma_0, \ldots, \Gamma_{m+1}$.

Claim 3 ($\star$). *We can compute* Γ_0 *in time* $\mathcal{O}(\mathsf{d})$.

Claim 4 ($\star$). *Given* Γ_j, *we can compute* Γ_{j+1} *in time* $\mathcal{O}(\mathsf{d}^{\mathcal{O}(\mathsf{ov})} \cdot \mathsf{ov}^{\mathcal{O}(\mathsf{ov})} \cdot \mathrm{poly}(|\mathcal{I}|))$.

Proof Sketch of Claim. It follows from Claim 2 that it suffices to compute each entry in Γ_{j+1} in time polynomial in $|\Gamma_j|$ and $|\mathcal{I}|$. Let $\sigma = (\mathrm{Nom}, \#\mathrm{Parties}, \mathrm{Left}, \mathrm{Right}, \mathrm{aft}, \mathrm{bef})$ be a satisfiable configuration for Γ_{j+1}. First note that, for each $i \in o_{j+1}$ we can determine whether the utility of $j+1$ is at most the utility of s_i under any satisfiability witness. Hence, we can already now check that $u_i(j+1, \mathbf{s}_{-i}) \leq u_i(s_i, \mathbf{s}_{-i})$ for every $i \in o_{j+1}$. From now on, we assume that it holds that $u_i(j+1, \mathbf{s}_{-i}) \leq u_i(s_i, \mathbf{s}_{-i})$ for every $i \in o_{j+1}$, where utilities are computed under the assumption that aft, bef, and $n_{i'}$, $\ell_{i'}$, $r_{i'}$, for all $i' \in o_{j+1}$, have their intended meaning in s.

We say that $\sigma' = (\mathrm{Nom}', \#\mathrm{Parties}', \mathrm{Left}', \mathrm{Right}', \mathrm{aft}', \mathrm{bef}')$ for Γ_j is a configuration that is *compatible* with σ, if the following is satisfied: (a) for every $i \in o_j \cap o_{j+1}$, $s_i = s'_i$, $n_i = n'_i$ $\ell_i = \ell'_i$, and $r_i = r'_i$; (b) either $\mathrm{bef} = \mathrm{bef}'$ or $\mathrm{bef} = \mathrm{aft} = j+1$; (c) either $\mathrm{aft} = \mathrm{aft}'$ or $\mathrm{bef}' = \mathrm{aft}' = j$; (d) if $s_i = j+1$ for $i \in o_{j+1}$, then $\ell_i = \mathrm{bef}'$; and (e) if $s'_i = j$ for $i \in o_j$, then $r'_i = \mathrm{aft}$.

To finish the proof of the claim one only needs to show $\Gamma_{j+1}[\sigma]$ is not NONE if and only if there exists a compatible configuration σ' such that $\Gamma_{j+1}[\sigma]$ is not NONE. □

The theorem is proven by applying Claim 3, then Claim 4 $m+1$ times, and finally checking all entries in Γ_{m+1}.

The running time of the XP algorithm from Theorem 1 immediately yields the following corollary.

Corollary 2. HD-NE *is fixed-parameter tractable parameterized by overlap and discrepancy.*

In addition, since overlap is trivially bounded by the number of parties, Corollary 2 gives the following.

Corollary 3. HD-NE *is fixed-parameter tractable parameterized by discrepancy and number of parties.*

3.1 W[1]-hardness

Next, we derive W[1]-hardness for HD-NE, when parameterized by overlap. To prove our result, we reduce from PARTITIONED SUBGRAPH ISOMORPHISM (PSI), which proves that the XP algorithm from Theorem 1 is essentially optimal, unless the exponential time hypothesis (ETH) fails [1].

In PSI we are given as input two undirected graphs G and H with $|V(H)| \leq |V(G)|$ (H is *smaller*) and a mapping $\psi \colon V(G) \to V(H)$ and the task is to determine whether H is isomorphic to a subgraph of G, i.e., decide if there is an injective mapping $\phi \colon V(H) \to V(G)$ such that $\{\phi(u), \phi(v)\} \in E(G)$ for each $\{u, v\} \in E(H)$ and $\psi \circ \phi$ is the identity.

Theorem 4 (see [Marx, 2010] and [Eiben *et al.*, 2019]). *If there is an algorithm* $\mathbb{A}$ *and an arbitrary function* f *such that* $\mathbb{A}$ *correctly decides every instance* (G, H, ψ) *of PSI with the smaller graph* H *being 3-regular in time* $f(|V(H)|)|V(G)|^{o(|V(H)|/\log|V(H)|)}$, *then ETH fails.*[2]

Theorem 5 ($\star$). HD-NE *is* W*[1]-hard parameterized by overlap of the input instance, even when the cost of the each position for each party is zero. Moreover, assuming ETH, there is no* $(n+m)^{o(\mathsf{ov})/\log(\mathsf{ov})}$ *time algorithm for* HD-NE.

Proof Sketch. We give a reduction from an instance (G, H, ψ) of PSI with H being 3-regular to HD-NE, such that $n + m$ is polynomial in the number of vertices of the graph G and overlap is linear in the number of vertices of H.

Let $q = |V(H)|$ and let $V_1, V_2, \ldots, V_q$ be the partition of $V(G)$ given by ψ^{-1}. Hence, two vertices $u, v \in V(G)$ are in the same part if and only if $\psi(u) = \psi(v)$. Let us denote by $\psi(V_i)$, $i \in [q]$, the vertex $\psi(u)$ for $u \in V_i$ (it is the same for every $u \in V_i$). Moreover, if for $i, j \in [q]$ there is an edge between $\psi(V_i)$ and $\psi(V_j)$ in $E(H)$, then let $E_{i,j}$ denote all the edges with one endpoint in V_i and the other in V_j. Note that the task in PSI is to select exactly one vertex in each of the sets V_i, $i \in [q]$, such that if $\psi(V_i)\psi(V_j)$ is an edge in H and we selected v_i in V_i and v_j in V_j, then $v_i v_j$ is an edge in $E_{i,j}$.

We now construct an instance of HD-NE that has a Nash equilibrium if and only if (G, H, ψ) is Yes-instance of PSI.

The instance will have the following parties:

- for each $i \in [q]$, there is a party R_i and a party G_i;
- for each $i, j \in [q]$ such that there is the set $E_{i,j}$ (i.e., there is the edge $\psi(V_i)\psi(V_j)$ in H) we have a party B^i_j (note that parties B^i_j and B^j_i are different);
- for each $i, j \in [q]$ such that there is the set $E_{i,j}$ and each $\ell \in [|E_{i,j}| - 1]$ there is party $P^\ell_{i,j}$;
- there are many so-called "dummy" parties, each dummy party will have exactly one candidate and the main purpose of dummy parties is to separate the gadgets we create. We denote the set of dummy parties D.

We will now create $|V(H)| + |E(H)|$ many "gadgets", where each gadget is a path and represents a subinterval of the space of the game. Each gadget will start and end with a candidate from some dummy parties that they have to be selected always as a strategy of these dummy parties. Hence, every non-dummy party that selects a candidate in some gadget, will only get voters inside that gadget. The game space is obtained by putting these gadgets in series in arbitrary order.

To create our gadgets we need to assign to every vertex of G a unique number such that given the sum of values assigned to two different vertices, we can uniquely determine the vertices. That is if we afterwards assign to each edge the sum of its endpoints, then each edge will be assigned a different value. Fortunately, such an assignment always exists. Indeed, a *Sidon sequence* is a sequence of natural numbers such that the sum of every two distinct numbers in the sequence is unique. Moreover, a result by [Erdös and Turán,

[1] ETH states that 3SAT, the canonical NP-complete problem, cannot be solved in subexponential time [Impagliazzo and Paturi, 2001].

[2] We would like to point out that, as far as we know, it is open whether PSI admits even an $f(|V(H)|) \cdot |V(G)|^{o(|V(H)|)}$ algorithm.

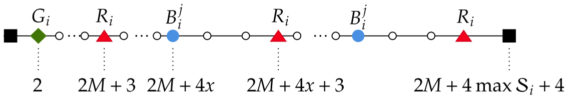

Figure 1: Illustration of the vertex set gadget for V_i. Above each candidate is the party it belongs to, below is its position.

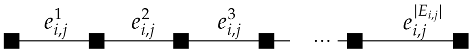

Figure 2: Illustration of the edge set gadget for $E_{i,j}$.

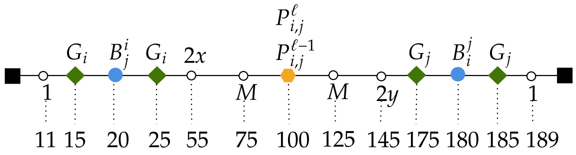

Figure 3: Illustration of the edge gadget for $e^\ell_{i,j}$. Above each candidate are the parties it belongs to, below each position with at least one voter is the number of the voters at the position. At the bottom are the indexes of the positions within the edge gadget.

1941] implies that a Sidon sequence containing $|V(G)|$ elements, whose largest element is at most $8|V(G)|^2$, can be found in polynomial time. In the following we will assume that we are given such a Sidon sequence $\mathcal{S}$ and that for every $x, y \in \mathcal{S}$ we have $|x - y| \geq 10$ and $\min(x, y) > 10$. Finally, before we construct the gadgets, we also need to fix some large number M that is much larger than $\max(\mathcal{S})$. Let us fix, for example, $M = (10\max(\mathcal{S}) + 100)^2$. Note that $M = \mathcal{O}(|V(G)|^4)$.

Let us assign to every vertex in G a unique number in $\mathcal{S}$. For the sake of exposition, for any $i \in [q]$ and $x \in \mathcal{S}$, we will denote by v^i_x the vertex of V_i that has been assigned the number x from the Sidon sequence $\mathcal{S}$. For each $i \in [q]$, let $\mathcal{S}_i$ denote the elements of $\mathcal{S}$ assigned to vertices in V_i.

We will have two types of gadgets:

Vertex Set Gadget. For every $i \in [q]$, we create the following gadget, called the vertex set gadget for V_i, (see Figure 1 for a depiction of the gadget). The gadget has $2M + 4 \cdot \max(\mathcal{S}_i) + 4$ positions. We describe the gadget as it was the interval $[2M + 4 \cdot \max(\mathcal{S}_i) + 3]$. We have two dummy parties, one with a unique candidate at 1 and one with a unique candidate at $2M + 4 \cdot \max(\mathcal{S}_i) + 4$. The party G_i has a candidate at position 2. The party R_i has candidates at position $2M + 4x + 3$ for each $x \in \mathcal{S}_i$ and a candidate at $2M + 3$. Each candidate, besides the candidate at the position $2M + 3$, is associated with a single vertex in V_i and we associate the candidate at the position $2M + 4x + 3$ with the vertex v^i_x (this is the vertex associated with the number $x \in \mathcal{S}$). Each of the parties B^i_j has a candidate at position $2M + 4x$ for each $x \in \mathcal{S}_i$. The party R_i will not have any other candidate, while G_i and B^i_j's will have also candidate in other gadgets. Finally, there is a single voter at every position in $[2M + 4 \cdot \max(\mathcal{S}_i) + 3]$. The intention behind this gadget is that in a Nash equilibrium, G_i selects the candidate at position 2, parties B^i_j do not choose their candidate inside the vertex set gadget for V_i, and R_i chooses some candidate at position $2M + 4x + 1$, for some $x \in \mathcal{S}_i$, representing the selection of the vertex v^i_x from V_i. Note that in this case the utility of R_i is $M + 2\max(\mathcal{S}_i)$, regardless of the selection of x and the utility of G_i is $M + 2x + 1$.

Edge Set Gadget. For every $E_{i,j}$ we create the following edge set gadget. Note that an edge $e = v^i_x v^j_y$ in $E_{i,j}$ is associated with the number $x + y$ and each edge is associated with a different number. First, we order the edges in $E_{i,j}$ in ascending order according to $x + y$. Let us denote the ℓ-th edge in this ordering $e^\ell_{i,j}$. The edge set gadget is split into $|E_{i,j}|$ many ”edge” gadgets, each starting and ending with a dummy candidate (i.e., a candidate for a dummy party) such that the dummy candidate at the end of the gadget for the edge $e^\ell_{i,j}$, $\ell \in [|E_{i,j}| - 1]$, is the same as the dummy candidate at the start of the gadget for $e^{\ell+1}_{i,j}$ (see Figure 2 for illustration).

The edge gadget of $e^\ell_{i,j}$ looks as follows (see Fig. 3 for illustration). It has 199 positions and we describe it as the interval $[199]$. There are two dummy parties with candidate at 1 and 199. If $\ell > 1$, then the party $P^{\ell-1}_{i,j}$ has a candidate at position 100. Similarly if $\ell < |E_{i,j}|$, then $P^\ell_{i,j}$ has also candidate at position 100. The party G_i has candidates at 15 and 25 and the party B^i_j has the candidate at position 20. Symmetrically, G_j has candidates at 175 and 185 and the party B^j_i has the candidate at position 180. The number of voters depends on the two vertices v^i_x and v^j_y such that $e^\ell_{i,j} = v^i_x v^j_y$. There is 1 voter at positions 11 and 189, there are $2x$ voters at position 55 and $2y$ voters at position 145, finally there at M voters at positions 75 and 125. The intention is that each party $P^\ell_{i,j}$ selects one of the two of its candidates in a way that no two of these parties select the same candidate. In this case the utility of $P^\ell_{i,j}$ is $2(M + x + y)$ for some $x, y \in \mathcal{S}$. This leaves precisely one of the edge gadget inside of the edge set gadget for $E_{i,j}$ empty, in which the parties B^i_j and B^j_i select their candidate. B^i_j gets utility $M + 2x + 1$, for some $x \in \mathcal{S}_i$ and B^j_i gets utility $M + 2y + 1$, for some $y \in \mathcal{S}_j$.

To obtain the game space we just fix an arbitrary ordering of the vertex set and edge set gadgets and identify the candidate for the dummy party at the end of one gadget with the candidate for the dummy party at the start of the next gadget in the ordering. The game space is then $[m]$, where m is the position of the last dummy candidate. Note that each vertex set gadget has length $\mathcal{O}(M + \max(\mathcal{S})) = \mathcal{O}(|V(G)|^4)$ and each edge set gadget has length $\mathcal{O}(|E_{i,j}|) = \mathcal{O}(|V(G)|^2)$. Hence $m = \mathcal{O}(|V(H)| \cdot |V(G)|^4) = \mathcal{O}(|V(G)|^5)$. This finishes the construction of the instance. It is rather straightforward to verify that each set o_j can contain an index of at most one dummy party and at most two from the $P^j_{\ell,i}$ parties. Since the number of remaining parties is $\mathcal{O}(|V(H)| + |E(H)|) = \mathcal{O}(|V(H)|)$, the overlap of the instance is indeed $\mathcal{O}(|V(H)|)$. It only remains to verify that the "intended strategy profile" is the only possible Nash equilibrium of the instance and it exists if and only if the instance (G, H, ψ) of PSI has a solution. □

4 Number of Positions, Discrepancy and Span

In this section, we study the number of different positions for the candidates, the discrepancy, and the span. First, we focus on the case where parties suffer uniform cost over the candidates. This allows us to derive a fixed-parameter tractable algorithm by the number of candidates.

Theorem 6 (⋆)**.** *When every party suffers uniform cost,* HD-NE *is in* FPT *parameterized by candidate positions.*

Proof Sketch. Let $\mathrm{cdp} = |C_1 \cup \cdots \cup C_n|$. We observe that for every party Π_i, we have $C_i \subseteq C_1 \cup \cdots \cup C_n$ and hence there are only 2^{cdp} many different possibilities for the set of the candidates of a party. For this proof, we will call the set C_i the *type* of the party Π_i. Since there are only 2^{cdp} many types and each type has at most cdp many candidates, we can for each type t and each candidate j in the type branch on whether at least one party of type t nominates j as the strategy of the party in an Nash equilibrium. Moreover, for each $j \in C_1 \cup \cdots \cup C_n$, we can branch on whether zero, one, or at least two parties select j as their strategy. This results to $3^{\mathrm{cdp}} \cdot 2^{\mathrm{cdp} \cdot 2^{\mathrm{cdp}}}$ many branches. Moreover, in each branch, the positions of all strategies in a strategy profile are determined by the branch and this is true even after removing one party strategy from the profile. Given this, and since the parties suffer uniform costs, the position of a best response for each party only depends on the number of parties at each of the positions determined by the branch. Moreover, given two candidates j_1 and j_2 one can easily derive a linear inequality (with number of parties with strategies j_1 and j_2, respectively, as variables) that is satisfied if and only if the party is happy with position j_1 and switching to j_2 would not increase its utility. Moreover, this inequality is exactly the same for every party with strategy j_1 and candidate at j_2. This allows us to create in each branch an instance of INTEGER LINEAR PROGRAMMING (ILP) that has variables x_j, $j \in C_1 \cup \cdots \cup C_n$, representing the number of parties with the strategy j and y_j^t representing the number of parties of type t with strategy j. The inequalities in the ILP then ensure that any solution to the inequality is consistent with the branch, each party is assigned some strategy and if $y_j^t \geq 1$, then j is a best response for every party of type t. Note that the number of variables in the ILP is at most $\mathrm{cdp} \cdot (1 + 2^{\mathrm{cdp}})$, and hence it can be solved in FPT-time [Lenstra, Jr., 1983]. □

Observe that the number of different positions for the candidates is trivially bounded by the span plus 1.

Corollary 7. *When every party suffers uniform cost,* HD-NE *is in* FPT *parameterized by span.*

Theorem 8 (⋆)**.** HD-NE *admits an* XP *algorithm parameterized by discrepancy.*

Proof Sketch. Similarly as in the proof of Theorem 6, we will refer to the set of candidates C_i of the party Π_i as the type of the party Π_i. The algorithm is again a dynamic programming along the space of the game $[m]$. For every $j \in m$ we have an entry for every tuple $(X, Y, \mathrm{aft}, \mathrm{bef})$, where

- $X = (x_{j-\mathsf{d}}, x_{j-\mathsf{d}+1}, \ldots, x_{j+\mathsf{d}})$ and $x_{j'} \in [|o_{j'}|]$; $x_{j'}$ represent the number of parties with strategy j';
- Y contains a number $y_{j'}^t \in [|o_{j'}|]$ for every j' in $[j - \mathsf{d}, j + \mathsf{d}]$ and every t the type of some party Π_i with $i \in o_j$; $y_{j'}^t$ represents the number of parties of type t with strategy j'.
- $\mathrm{aft} \in [j + \mathsf{d} + 1, m] \cup \{\infty\}$ represents the position of the closest nominee selected as a strategy to the right of $j + \mathsf{d}$ (after $j + \mathsf{d}$).
- $\mathrm{bef} \in \{1, \ldots, j\} \cup \{-\infty\}$ represents the position of the closest nominee selected as a strategy to the left of $j + 1$ (before $j + 1$). In particular if for some $j' \in [j - \mathsf{d}, j]$ we have $x_{j'} > 0$, then bef is the maximum $j' \in [j - \mathsf{d}, j]$ such that $x_{j'} > 0$, else $\mathrm{bef} < j - \mathsf{d}$.

Now for every $j \in [m]$ we will compute a table Γ_j that contains for every tuple $(X, Y, \mathrm{aft}, \mathrm{bef})$ either NONE or a partial strategy profile s that contains a strategy for every party Π_i with $c_{i1} \leq j$ that is *consistent* with the tuple $(X, Y, \mathrm{aft}, \mathrm{bef})$. We say that s is *consistent* with $(X, Y, \mathrm{aft}, \mathrm{bef})$ if:

1. for $j' \in [j - \mathsf{d}, j]$, the number of parties with strategy j' is precisely $x_{j'}$ and for $j' \in [j + 1, j + \mathsf{d}]$ it is at most $x_{j'}$;
2. for $j' \in [j - \mathsf{d}, j, j + \mathsf{d}]$ and type t, if Y contains $y_{j'}^t$, then the number of parties of type t with strategy j' is precisely $y_{j'}^t$;
3. for every $s_i \in \mathsf{s}$, if $s_i \leq j$, then $s_i \leq \mathrm{bef}$, moreover there exists a party $\Pi_{i'}$ with strategy bef in s;
4. every party with a strategy in s plays the best response if we assume in addition that the number of parties with strategy $j' \in [j + 1, j + \mathsf{d}]$ is precisely $x_{j'}$, there exist at least $x_{j'}$ parties with candidate j', and there is a party with candidate aft.

We finish the proof by showing that we need to consider only $(\mathsf{ov} + 1)^{(2\mathsf{d}+1)(1+2^{2\mathsf{d}+1})} \cdot (2\mathsf{d} + 1)^2$ many tuples, we can compute Γ_0, and we can compute Γ_{j+1} from Γ_j. □

5 Conclusions

Our results show that natural constraints on the positions of the candidates can, in many cases, yield fixed-parameter tractable algorithms. To this end, there are several interesting questions that stem from our work. First, can we strengthen Theorem 6 by removing the assumption of uniform cost, or does the problem become $\mathsf{W}[1]$-hard? Another question is whether HD-NE is in FPT parameterized by the discrepancy. A different direction is to combine the model from [Ben-Porat and Tennenholtz, 2019], where parties can choose more than one positions, with our model.

Another, really intriguing, question is the computation of *mixed* Nash equilibria; there each party chooses a candidate according to a probability distribution. Then, a Nash equilibrium *always* exists. In fact, in this case the problem becomes very interesting even if there are *only two* parties. If the parties have uniform costs, then the underlying game is constant sum and a Nash equilibrium can be computed via linear programming. On the other hand, if the parties suffer general costs, then it is not clear whether a polynomial-time algorithm exists, or the problem becomes PPAD-complete!

References

[Ahn *et al.*, 2004] Hee-Kap Ahn, Siu-Wing Cheng, Otfried Cheong, Mordecai Golin, and Rene Van Oostrum. Competitive facility location: the Voronoi game. *Theoretical Computer Science*, 310(1-3):457–467, 2004.

[Ben-Porat and Tennenholtz, 2019] Omer Ben-Porat and Moshe Tennenholtz. Multiunit facility location games. *Mathematics of Operations Research*, 44(3):865–889, 2019.

[Brusco *et al.*, 2012] Sandro Brusco, Marcin Dziubiński, and Jaideep Roy. The Hotelling-Downs model with runoff voting. *Games and Economic Behavior*, 74(2):447–469, 2012.

[Cygan *et al.*, 2015] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015.

[Downs, 1957] Anthony Downs. *An economic theory of democracy*. Harper & Row New York, 1957.

[Dürr and Thang, 2007] Christoph Dürr and Nguyen Kim Thang. Nash equilibria in Voronoi games on graphs. In *European Symposium on Algorithms*, pages 17–28. Springer, 2007.

[Eaton and Lipsey, 1975] B Curtis Eaton and Richard G Lipsey. The principle of minimum differentiation reconsidered: Some new developments in the theory of spatial competition. *The Review of Economic Studies*, 42(1):27–49, 1975.

[Eiben *et al.*, 2019] Eduard Eiben, Dusan Knop, Fahad Panolan, and Ondrej Suchý. Complexity of the Steiner network problem with respect to the number of terminals. In *STACS 2019*, volume 126 of *LIPIcs*, pages 25:1–25:17, 2019.

[Eiselt *et al.*, 1993] Horst A Eiselt, Gilbert Laporte, and Jacques-Francois Thisse. Competitive location models: A framework and bibliography. *Transportation science*, 27(1):44–54, 1993.

[Eiselt, 2011] Horst A Eiselt. Equilibria in competitive location models. In *Foundations of location analysis*, pages 139–162. Springer, 2011.

[Erdös and Turán, 1941] Paul Erdös and Pál Turán. On a problem of sidon in additive number theory, and on some related problems. *Journal of the London Mathematical Society*, 1(4):212–215, 1941.

[Faliszewski *et al.*, 2016] Piotr Faliszewski, Laurent Gourvès, Jérôme Lang, Julien Lesca, and Jérôme Monnot. How hard is it for a party to nominate an election winner? In *Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016*, pages 257–263, 2016.

[Feldman *et al.*, 2016a] Michal Feldman, Amos Fiat, and Iddan Golomb. On voting and facility location. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pages 269–286, 2016.

[Feldman *et al.*, 2016b] Michal Feldman, Amos Fiat, and Svetlana Obraztsova. Variations on the Hotelling-Downs model. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[Harrenstein *et al.*, 2021] Paul Harrenstein, Grzegorz Lisowski, Ramanujan Sridharan, and Paolo Turrini. A Hotelling-Downs framework for party nominees. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, pages 593–601, 2021.

[Hotelling, 1929] Harold Hotelling. Stability in competition. *The Economic Journal*, 39(153):41–57, 1929.

[Impagliazzo and Paturi, 2001] Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-sat. *Journal of Computer and System Sciences*, 62(2):367–375, 2001.

[Lenstra, Jr., 1983] H. W. Lenstra, Jr. Integer programming with a fixed number of variables. *Math. Oper. Res.*, 8(4):538–548, 1983.

[Marx, 2010] Dániel Marx. Can you beat treewidth? *Theory Comput.*, 6(1):85–112, 2010.

[Meir, 2018] Reshef Meir. Strategic voting. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 13(1):1–167, 2018.

[Sengupta and Sengupta, 2008] Abhijit Sengupta and Kunal Sengupta. A Hotelling-Downs model of electoral competition with the option to quit. *Games and Economic Behavior*, 62(2):661–674, 2008.

[Shen and Wang, 2016] Weiran Shen and Zihe Wang. Hotelling-Downs model with limited attraction. *arXiv preprint arXiv:1611.05959*, 2016.

[Stokes, 1963] Donald E Stokes. Spatial models of party competition. *American political science review*, 57(2):368–377, 1963.