

Putting Fair Division on the Map

Paula Böhm, Robert Brederick, Paul Gözl, Andrzej Kaczmarczyk, and Stanisław Szufa

Abstract

The fair division of indivisible goods is not only a subject of theoretical research, but also an important problem in practice, with solutions being offered on several online platforms. Little is known, however, about the characteristics of practical allocation instances and how they compare to the characteristics of synthetic allocation instances. Taking inspiration from the work of Szufa et al. (2020), we devise a map of allocation instances for indivisible goods. This map identifies two key axes along which allocation instances differ, which help distinguish synthetic distributions, predict key features of the allocation instances, and can be conceptually interpreted.

1 Introduction

Over the past 20 years, the field of fair division has made great advances in studying allocations of indivisible goods (Amanatidis et al. 2023). To illustrate this progress, consider the axiom of *envy-freeness*, which demands that no agent prefer another agent’s bundle of allocated goods to their own. By the end of the 20th century, economists already understood envy-freeness well in settings with *divisible* goods. For example — assuming that preferences are additive, as we do throughout the paper — the allocation maximizing the Nash welfare is envy-free in these settings (Varian 1974; Shafer and Sonnenschein 1982). Little was known, however, about indivisible goods, a domain whose combinatorial structure poses additional challenges to mathematical investigation. Since envy-free allocations need not exist for all indivisible allocation instances, could envy at least be limited, or were large amounts of envy unavoidable?

Since then, fair-division researchers have gained a refined understanding of the degree to which envy can (and cannot) be avoided. Notably, the field has coalesced around an attractive relaxation of envy-freeness — *envy-freeness up to one good*, or EF1 for short (Budish 2011) — and identified elegant algorithms (Lipton et al. 2004; Caragiannis et al. 2019) that construct EF1 allocations for any instance. Though intriguing open questions remain,¹ these questions sharpen and extend a solid understanding of the landscape of allocations. For alternative families of fairness axioms (such

¹For example, whether envy-freeness up to *any* good (EFX) can be guaranteed (e.g. Chaudhury, Garg, and Mehlhorn 2020; Plaut and Roughgarden 2020; Amanatidis et al. 2021).

as the maximin share, proportionality, and equitability), fair division has made similar progress in understanding which axioms can be guaranteed even on worst-case allocation instances (see Amanatidis et al. 2023).

In parallel to these theoretical advances, algorithms for allocating indivisible goods have entered practical usage, raising new questions for fair division. The Course Match system, for example, assigns course seats to MBA students at Wharton (Budish et al. 2017), and thousands of users have used the website Spliddit (Goldman and Procaccia 2014) to divide up estates or joint possessions. The deployment of such systems makes it more pressing to study not only worst-case instances but *typical* instances encountered in practice as well. For example, though envy-free allocations do not exist for all instances, should algorithms not aim for envy-free allocations for the 71% of Spliddit instances (Bai et al. 2022) where envy-freeness is possible? If so, how to choose among envy-free allocations? Or, as a second example, which algorithms can be implemented in practice? After all, the algorithms deployed on Course Match (Budish et al. 2023) and Spliddit (Caragiannis et al. 2019) run fast on practical inputs, seemingly defying (worst-case) computational hardness results. Answers to these questions cannot be found through worst-case analysis alone.

Whereas most work in fair division follows the worst-case paradigm, a noteworthy exception is some work in the paradigm of *distributional analysis*, which assumes that allocation instances are drawn from a probability distribution (Roughgarden 2020). Typically, these distributional models assume that all agent–good values are drawn independently, either from a single distribution (Amanatidis et al. 2017; Manurangsi and Suksompong 2019, 2021), a distribution that depends on the agent (Kurokawa, Procaccia, and Wang 2016; Farhadi et al. 2019; Bai and Gözl 2022), or a distribution that depends on the alternative (Dickerson et al. 2014; Farhadi et al. 2019).

On the upside, when m and n are large, these distributions generate highly structured instances, for which fair allocations are more prevalent. For example, basic algorithms yield envy-free allocations for these instances, with high probability (Dickerson et al. 2014; Manurangsi and Suksompong 2021). On the flip side, we are not aware of any empirical work that has tested if the structures of these random

instances are present in practical allocation problems.² In recent work, Bai et al. (2022) try to overcome some of these concerns through *smoothed analysis*, which means that their probability distributions are defined by adding random noise to a worst-case utility profile. While they extend the possibility results for envy-freeness to more general distributions, they leave “whether our smoothed model accurately describes the properties of real-world utility profiles that possess envy-free allocations” to future empirical analysis.

Recently, social choice theory faced, and addressed, a similar need for bridging the gap between theoretical work and practical instances. Social choice is famously riddled with worst-case impossibilities (Campbell and Kelly 2002), and modern social choice theory has been largely divorced from the analysis of election data. Part of the theory considered distributional models of elections, but these models were overly prescriptive and, anyways, not easy to relate to real-world elections. To address these concerns, Szufa et al. (2020) created a *map of elections*: A two-dimensional embedding of election instances (coming at first from distributional models, in subsequent works (Boehmer et al. 2021; Faliszewski et al. 2023) also from real-world data) with the following properties:

- This map recovers tell-tale features of the different data sources, turning elections generated by a specific source into a compact cluster, which evidently only covers a small subset of interesting elections.
- Key features of election instances vary continuously over the map, showing that an election’s position on the map is highly informative—despite summarizing the high-dimensional space of elections in just two dimensions.
- The map’s two axes can be given a conceptual interpretation, and the boundary of the map be traced by natural “extreme” elections.

One particular achievement of this line of work was to highlight distributional models which, at least on the axes of the map, capture the range of interesting elections. We will apply a similar approach to fair division.

1.1 Our Approach and Results

In fact, we create *two* maps of allocation instances for indivisible goods in this paper. In Section 3, we create a map by closely following the methodology of Szufa et al. (2020): we define a natural distance between fair division instances (the minimal ℓ^1 distance between utility matrices after row and column permutations), as well as a more computationally tractable proxy distance. We then use multi-dimensional scaling () to find a 2-dimensional *distance-embedding map* of a given set of allocation instances that approximately preserves the pairwise distances.

Using data from three real-world data sources and several synthetic distributions over approval instances, we show that the map picks up on common properties of instances from the same data source. We also show that key features of the allocation instance (e.g., the maximum achievable Nash wel-

fare or the existence of envy-free allocations) are distributed in clear patterns across the map.

In Section 4, we go beyond this heuristically generated embedding, by providing an *explicit map*, i.e., an explicit function from allocation instances into \mathbb{R}^2 , which reproduces the general layout of the distance-embedding map. Since an instance’s position on this explicit map is given by the two largest singular values of its utility matrix, our explicit map is amenable to theoretical analysis. In particular, we tightly characterize the range of the map, and identify (up to rounding terms) the most extreme instances in the map’s four corners. We conclude by showing that the explicit map can similarly segment instances sources and features.

2 Preliminaries

2.1 Model

Let $[n]$ be a set of agents and $[m]$ be a set of goods. For ease of exposition, we assume throughout the paper that $m \geq n \geq 2$, which arguably includes all interesting allocation instances. A *fair-allocation instance* (of indivisible goods) is described by a *utility matrix* $U \in \mathbb{R}_{\geq 0}^{n \times m}$ whose entries $u_{i,j}$ describe agent i ’s utility for good j . We refer to the i th row of this matrix as i ’s *utility vector* $\vec{u}_i \in \mathbb{R}_{\geq 0}^m$. We assume that preferences are additive, so that agent i ’s utility for a *bundle* $S \subseteq [m]$ of goods is given by $u_i(S) := \sum_{j \in S} u_{i,j}$. Since we only consider tasks in which agents’ utilities $u_i([m])$ for the whole bundle are normalized to 1, we consider exactly the set of *row-stochastic* matrices U .

2.2 Characteristic Instances

As useful signposts for navigating through the space of allocation instances, we define several *characteristic instances*. Each of these instances represents an intuitively extreme scenario with easily-understood, symmetric structure. Thanks to this structure, it is easy to remember the properties of these characteristic instances, making them a useful point of reference in a map of allocation instances. Due to space limitations, we introduce these instances in words here, and refer the reader to Fig. 7 in the appendix for a matrix representation. For any n and m , our three characteristic instances are the following:

Indifference (IND) models the situation where each good is equally valuable to each agent. Thus, all entries of its utility matrix are $1/m$.

Separability (SEP) captures the scenario in which each agent values only a single good, distinct from the goods that all other agents value. Thus, its utility matrix is a matrix with ones on the diagonal and zeros everywhere else. In particular, if $m > n$, then all but the first n columns have all-zero entries.

Contention (CON) describes the case in which one single good is valued by all agents, and all other goods have no value to any agent. Hence, its utility matrix has a first column of ones, and is zero everywhere else.

In Section 4.2, the explicit map will lead us to introduce three new characteristic instances: two variants of separability, and an entirely new instance called *bicontention*.

²Bai et al. (2022) voice doubts, but also do not provide data.

2.3 Real-World Instances

We will map—and construct our distance-embedding map—from a range of real-world allocation instances³. We consider three sources of allocation instances that are derived from real-world preferences over goods:

Spliddit. The heart of our real-world data is a dataset (Shah 2022) of all allocation instances submitted to Spliddit as of 2022. This dataset is particularly valuable because it represents instances that real Spliddit users were hoping to solve. Since most of the 3000 Spliddit instances are small, our evaluation will focus on two combinations of n, m that are relatively well represented: First, we will study the setting $n = 3, m = 6$, for which the number of instances is highest (1847). Since, for larger dimensions, the number of Spliddit instances drops precipitously, only 16 Spliddit instances exist for our second evaluation scenario of $n = m = 5$.

Island. To obtain this dataset (Benadè 2023), Benade et al. (2018), though motivated by a public-goods setting, elicited additive utilities for private goods, by asking 572 crowd workers to spread 100 points between 10 items in proportion to how much they would value these items (a map, pocket knife, compass, etc.) if they were stranded on a deserted island. By repeatedly sampling sets of n agents and m goods and rescaling agents’ utilities, we simulate a (hypothetical) allocation scenario in which only those m goods stand to be allocated between those n (real but fictitiously stranded) agents.

Candy. Our final dataset (Anonymized 2023) has a similar shape and consists of the additive preferences indicated by 48 teenagers attending a summer camp over 10 types of snacks. We again obtain instances by subsampling, assuming that only one snack of each type is available.

2.4 Distributions over Synthetic Instances

In addition to the above instances derived from practical data, we consider synthetic instances drawn from the following types of distributions. We introduce the resampling and Dirichlet-resampling models (inspired by the resampling model for approval elections by Szufa et al. (2022)) as we empirically checked that using them we could cover almost the whole space of instances. The Euclidean models are standard in the social choice literature, while the attributes model, also used by Boehmer, Heeger, and Szufa (2023), seem to model a simple heuristic behavior of agents.

Resampling. For each agent we generate a set of *approved* goods over which the agent splits the total utility of 1 equally. Given two parameters $p \in [0, 1]$ and $\phi \in [0, 1]$, we first choose the instance’s *central approval set* V^* by uniformly drawing $\lfloor p \cdot m \rfloor$ goods. Then, we generate each agent’s approval set V by copying V^* and then altering it as follows: for each good, with probability $1 - \phi$ we do nothing, and with probability ϕ we resample its membership in V , i.e., we put it in V with probability p .

Dirichlet-Resampling. We first sample each agent’s approval set V using the resampling model. Then, we set each agent’s utility for goods outside of V to zero, and determine their utilities for the goods in V from a symmetric Dirichlet distribution of the appropriate number of categories, with some fixed parameter α .

Attributes. Let $d \in \mathbb{N}$ be a fixed number of *attributes*. For each good, we sample a vector \vec{g} from $[0, 1]^d$ uniformly at random (higher coordinates indicate that the good is more desirable along an attribute). For each agent we sample their utility vector \vec{a} over the attributes also from $[0, 1]^d$ (higher coordinates indicate that the agent puts more weight on an attribute). The agent’s a utility for good g is then proportional to the dot product of \vec{a} and \vec{g} .

Euclidean. For a fixed dimension $d \in \mathbb{N}$, we sample for each agent i a vector \vec{a}_i and each good j a vector \vec{g}_j , independently and uniformly from $[0, 1]^d$. Then, agent i ’s utility for good j is proportional to $1 - \frac{\|\vec{a}_i - \vec{g}_j\|}{\max_{j' \in [m]} \|\vec{a}_i - \vec{g}_{j'}\|}$.

Details on the choice of samples from each distribution and the parameters for our experiments are in Appendix B.

3 Distance-Embedding Map

We create our first map of allocation instances by following the following two steps, introduced by Szufa et al. (2020): First, we define appropriate structural distances between pairs of instances. Second, we embed this high-dimensional space (the distance matrix) on a plane in order to recognize patterns, clusters, or similar-feature areas.

3.1 Distances between Allocation Instances

Conceptually, we would like to measure the distance between two instances with equal n and m through the entry-wise ℓ^1 norm. That is, if the instances’ utility matrices are U^1 and U^2 , we would calculate their distance $\|U^1 - U^2\|_{1,1}$ by summing up, over all $n \cdot m$ coordinates, the absolute difference between U^1 ’s and U^2 ’s entry in this coordinate. This distance is, however, not desirable, since the ordering of rows and columns in a utility matrix are arbitrary, but re-ordering rows (i.e., agents) or columns (i.e., goods) should not impact the distance between instances.

We define a *valuation distance* between instances, which achieves anonymity and neutrality by explicitly minimizing over all row and column permutations. Putting this in less matrix-heavy language, suppose that we have a bijection $\pi_{agents} : [1, n] \rightarrow [1, n]$ (i.e., a permutation in Π_n) between the agents of the first and second allocation instance, as well as a similar bijection $\pi_{goods} \in \Pi_m$ from the goods of the first to the goods of the second allocation instance. By summing up, for each agent i and each good j , the absolute difference $|U^1_{i,j} - U^2_{\pi_{agents}(i), \pi_{goods}(j)}|$ between i ’s utility for j and the utility of i ’s matched agent $\pi_{agents}(i)$ for j ’s matched good $\pi_{goods}(j)$, we calculate the entry-wise ℓ^1 distance; the valuation distance is defined as the minimum of this distance, taken over all matchings π_{agents} and π_{goods} . A nice property of the valuation distance is that two instances U^1 and U^2 have distance zero exactly if they are identical

³The data was obtained on request from the data owner.

up to relabeling agents and goods, i.e., they are *isomorphic* (analogous to that of Faliszewski et al. (2019) for elections).

While conceptually appealing, the valuation distance is NP-hard to compute (see Theorem 5 in Appendix C.1). Computing it for large or many (as in our experiments) instances is computationally prohibitive (we used an ILP formulation similar to the one by Faliszewski et al. (2019)).

To sidestep this computational difficulty, we introduce the *demand distance*. Here, for each good of both instances, we build a *demand vector* containing the utility values that the good receives from different agents, sorted in decreasing order. We then find a mapping of vectors from one instance to the other that minimizes the sum of ℓ_1 distances of the mapped pairs. Hence, we obtain the following formal definition (Appendix C.2 contains a generalized definition).

Definition 1. Let U^1 and U^2 be two allocations instances with n agents and m goods. The *demand vector* $\vec{\text{dem}}_U(j)$ of good $j \in [m]$ of instance U is the vector $(U_{1,j}, U_{2,j} \dots U_{n,j})$ sorted in descending order. The *demand distance* $d_v(U^1, U^2)$ of allocations U^1 and U^2 is then

$$\min_{\pi_{\text{goods}} \in \Pi_m} \sum_{j \in [m]} \|\vec{\text{dem}}_{U^1}(j) - \vec{\text{dem}}_{U^2}(\pi_{\text{goods}}(j))\|_1.$$

Due to the fact that this definition optimizes over only a single permutation, the demand distance can be computed in polynomial-time by finding a minimum matching in a weighted bipartite graph representing the contributions to the distance from matchings between each pair of agents (see Theorem 7 and its proof in Appendix C.2). But some precision is lost: the demand distance ignores information about the identity of agents, which can lead two non-isomorphic instances to be at distance 0 from each other (Fig. 8 in Appendix C.2 depicts an example). Fortunately, we gain a drastic increase in computational efficiency with only a small loss of accuracy. The Pearson correlation coefficient of the two distances for our instances⁴ was always higher than 97%. This very high correlation strongly suggests that using the demand distance as a proxy for the valuation distance is unlikely to fundamentally change our mapping. Hence, we from now on focus on the demand distance.

Encouragingly, the demand distance also confirms our intuition that the characteristic instances introduced in Section 2.2 — indifference (IND), separability (SEP), and contention (CON)—are indeed “extreme” instances in the space of allocation instances. Indeed, if the number of agents is equal to the number of goods then: (i) the characteristic instances are mutually equidistant, at the distance of $2(m-1)$, and (ii) no other pair of instances has strictly larger distance (see Proposition 1 in Appendix C.2).

3.2 Studying the Distance-Embedding Map

In order to plot a map of fair divisions, we generate instances, compute the demand distance between each pair of instances, and embed these distances in two-dimensional

⁴Due to computational constraints, this correlation is computed on a subset of instances datasets, described in Appendix B. See Appendix C.1 in the appendix for a correlation diagram.

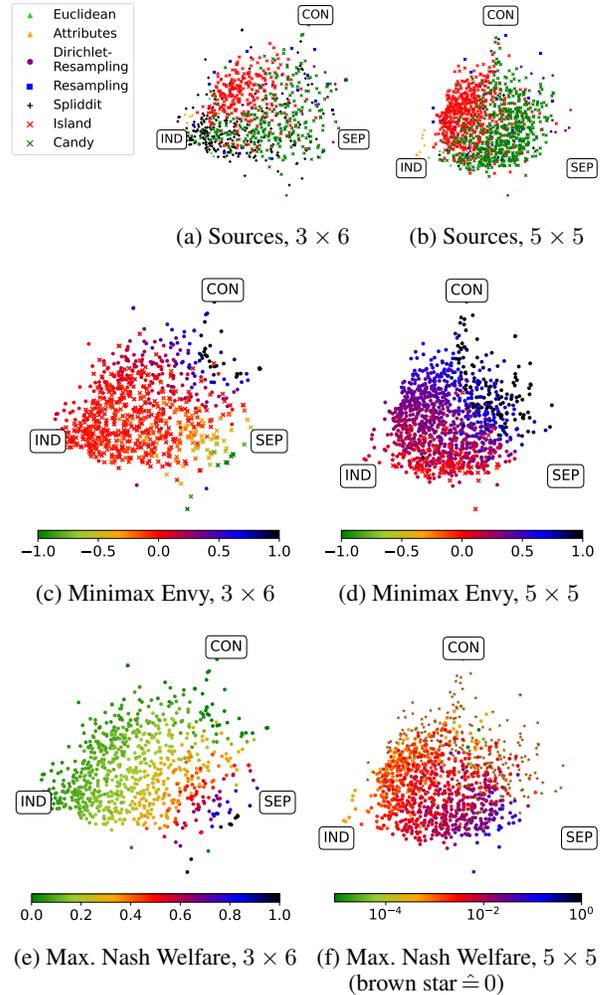


Figure 1: Distribution of instance sources and two features on our distance-embedding map for 3×6 and 5×5 instances.

Euclidean space using multi-dimensional scaling⁵ exercising mapel⁶ — a framework originally developed for understanding elections, which we extended to also handle allocation instances and their features (we implemented the approach of Brederick et al. (2021)). As mentioned in Section 2.3, we create such maps for two combinations of n, m : first, for $n = 3, m = 6$ (“ 3×6 ” from here on) and second for $n = m = 5$ (“ 5×5 ”).

We display the resulting maps in Fig. 1. The three characteristic instances lie at distinct extreme ends of the map, which is in line with our observations at the end of the previous paragraph. Most of the map is spanned in a triangle between these three instances, which indeed makes them useful points of reference of the resulting map. The map also suggests that, among 3×6 instances, instances tend to lie closer to indifference than to either contention and separa-

⁵We use `sklearn.manifold.MDS` from Python.

⁶<https://mapel.simple.ink>

bility. For the latter, we observe a slightly different pattern on 5×5 instances, where most instances seem to lie in a center, at some distance from each of the extreme points.

We can also immediately make out that the more numerous instance sources are spread unevenly over the map: instances from the candies dataset seem to be closer to separability (perhaps suggesting that the children have strong preferences for different snacks), the island dataset seems to vary more between indifference and contention, and the most of the Spliddit instances are located closer to indifference (at least for 3×6 instances, where there is enough data to make such observations). Though the synthetic distributions jointly cover a large degree of the map, there are big differences between them. For example, the instances from the attribute distribution are noticeably close to indifference, raising the question whether this distribution produces sufficiently diverse instances as a proxy for practice.

In the second row of Fig. 1, we study to which degree the instances allow for (almost) envy-free allocations. Specifically, denote an allocation of all goods over the agents by $[m] = S_1 \dot{\cup} S_2 \dot{\cup} \dots \dot{\cup} S_n$, where S_i denotes the bundle of goods given to agent i . The *minmax envy* is the minimum, over all allocations, of the largest amount by which some agent envies another, i.e., $\max_{i \neq i'} u_i(S_{i'}) - u_i(S_i)$. Note that an instance has envy-free allocations iff the minmax envy is at most 0 (we additionally highlight these instances with a cross in Figs. 1c and 1d). But the minmax envy gives a gradual measure of how far envy-freeness is from being achievable (or how much it can be overattained). As we can see, an instance’s position on the map is highly informative for the minmax envy and the existence of envy-free allocations. For 3×6 instances, envy-freeness seems to be hopeless near contention (which is also the case for contention itself) and easy near separability. For the rest of the map, minmax envy is close to zero, which means that almost envy-free allocations exist widely, and exactly envy-free allocations generally exist below the upper outline of the map. 5×5 instances are less hospitable to envy-freeness: envy-free allocations exist only near the lower border of the map, and the minmax envy becomes higher (i.e., worse), the further up on the map an instance lies.

The third row of Fig. 1 shows that the maximum Nash welfare achievable by any allocation also varies smoothly over the map. We can see that this quantity smoothly increases the closer an instance lies to separability. It is interesting to note that this map differs only slightly from the maximum utilitarian welfare that can be achieved (see Fig. 5 below). We show the distribution of various additional features, as well as disaggregated plots of the the distributions of instance sources in Appendix F.

4 Explicit Maps

Despite its many advantages, generating maps through a distance embedding entails inherent disadvantages:

Instability. The distance-embedding map may change drastically as the result of slight changes to the random seed or the set of mapped instances.

Data Dependence. Suppose that you want to place a allo-

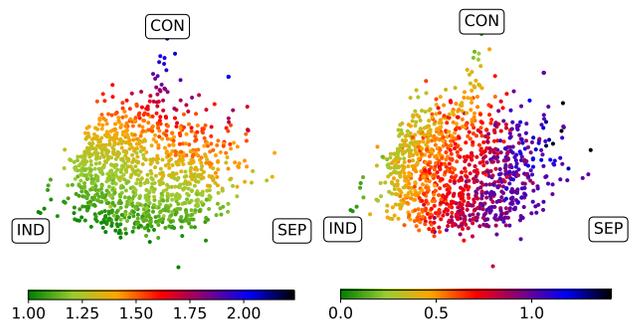


Figure 2: Distribution of σ_1 (left) and σ_2 (right) on our distance-embedding map for 5×5 .

cation instance on the map to predict its properties. This would require data for all other instances and computing pairwise distances, which would be more difficult than directly computing your instance’s properties.

Theoretical Intractability. Which instances are “most extreme”? Where are instances from a probability distribution located on the map? Such questions can be answered empirically, but not theoretically.

To overcome these challenges, we propose an *explicit map* of fair division instances: a function μ from allocation instances to \mathbb{R}^2 , which replicates the general layout of the distance-embedding map. Specifically, this function maps $n \times m$ utility matrices as follows:

$$\mu : \mathbb{R}^{n \times m} \rightarrow \mathbb{R}^2 \quad U \mapsto (\sigma_1(U), \sigma_2(U)),$$

where $\sigma_1(U)$ and $\sigma_2(U)$ are the largest and second-largest singular values of the matrix U , respectively. As Fig. 2 shows, these two values closely capture the vertical and horizontal ordering of instances in our distance-embedding map, ensuring that the two maps are closely aligned.

In this section, we show that the explicit map is similarly informative as the distance-embedding map, while being stable, data independent, and theoretically tractable by design.

4.1 Demystifying the Singular Value Map

We begin by recalling facts about singular values that make them suitable components for our explicit map function. First, the singular values are invariant under permutations of rows or columns in the utility matrix, so that relabeling agents or goods will not change the map embedding. Second, σ_1 and σ_2 are 1-Lipschitz continuous in the entries of the matrix, which together with the previous point implies that two instances with small valuation distance must be placed near each other on the explicit map. Third, adding a column of zeros, i.e., a good that no agent values, does not change the singular values, which means that instances can be naturally compared across different m . Finally, implementations of efficient algorithms for computing singular values are readily available (e.g., in numpy), which makes it easy to compute a given instance’s position on the map.

We now aim to give the reader an intuition for what information σ_1 and σ_2 express about an allocation instance and

why. We begin with σ_1 , which can be expressed as

$$\sigma_1 = \max_{\vec{v}_1 \in \mathbb{R}^m, \|\vec{v}_1\|=1} \|U \vec{v}_1\|, \quad (1)$$

where $\|\cdot\|$ is the Euclidean (ℓ^2) norm. Since we rarely think about utility matrices as linear functions over unitary vectors, it is instructive to pretend that the norms in Eq. (1) were ℓ^1 -norms. In this case (choosing \vec{v}_1 nonnegative w.l.o.g.), the $U \vec{v}_1$ being optimized over are the convex combination of U 's columns, for the coefficients given by \vec{v}_1 . If we were indeed maximizing the ℓ^1 -norm of $U \vec{v}_1$, σ_1 would be the largest column sum, or *maximum demand*. Though the ℓ^2 norm slightly complicates the picture,⁷ σ_1 and the maximum demand are very highly correlated: across our instances with $n = 3$, for example, the correlation coefficient is 97%. Thus, σ_1 can be understood to a good approximation as the maximum demand, up to shifting and rescaling.

To interpret the second-largest singular value σ_2 , we recall how the singular value decomposition of an $\mathbb{R}^{n \times m}$ matrix U can be used to find a low-dimensional embedding of the row vectors (in our case, the agents' utility vectors).⁸ For example, the line through the origin $\text{span}(\{v_1\})$, spanned by the argmax of Eq. (1), is the best 1-dimensional space to embed the rows in, in the following sense: if we sum up, for each row $\vec{u}_i \in \mathbb{R}^m$, the squared length of its projection onto this space, $\text{span}(\{v_1\})$ maximizes this sum across all 1-dimensional subspaces. In fact, this sum of squared projection lengths is σ_1^2 , which means that σ_1 measures “how much” of the row vectors can be captured by a 1-dimensional embedding. Similarly, σ_2 , which can be calculated as $\max_{\vec{v}_1 \in \mathbb{R}^m, \|\vec{v}_1\|=1, \vec{v}_2 \perp \vec{v}_1} \|U \vec{v}_2\|$, measures how much the row embedding improves when going from the optimal 1-dimensional space $\text{span}(\{v_1\})$ to the optimal 2-dimensional space $\text{span}(\{v_1, v_2\})$.

Thus, as a first approximation, σ_2 is a measure of how diverse the agents' utilities are. It is zero if all agents have the same utility vector, and it is large when there are blocks of agents that completely disagree on which goods have nonzero value. To again find a more elementary correlate, we define a fair division's “preference diversity” as the mean, over pairs of agents in the instance, of the ℓ^2 distance of their utility vectors. Again, we find a very high correlation (96% correlation coefficient for $n = 3$).

4.2 Theoretical Properties of the Map

We theoretically characterize the image of our map function μ for given dimensions n, m . Our task — characterizing

⁷It gives an advantage to combinations of columns in which several columns have positive coefficients, and it encourages making a few coordinates of $U \vec{v}_1$ large rather than all.

⁸See Blum, Hopcroft, and Kannan (2020, Ch. 3) for a detailed explanation. Though singular values are closely connected to dimensionality reduction, our use is non-standard. Applying value decomposition directly to find a 2D embedding of utility matrices, would result in embeddings highly sensitive to row and column permutations and thus not fruitful. One way to understand the discussion above is that we map each utility matrix to the square roots of the top-two eigenvalues in its principle component analysis; except that we do not shift column sums to zero, since this would, e.g., make IND and CON indistinguishable.

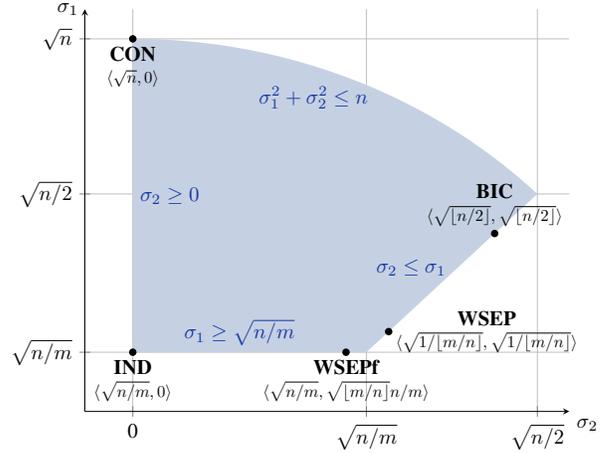


Figure 3: Bounding inequalities of the map, and locations $\langle \sigma_1, \sigma_2 \rangle$ of characteristic instances.

the combinations of singular values in stochastic rectangular matrices — is of interest independently to our fair-division setting, but has, to our knowledge not previously been undertaken. This process will give us a more precise understanding of what makes instances extreme along either dimension of our map. Figure 3 summarizes both the outlines of the map and the positions of characteristic instances, which can guide the reader through this section. We orient on the page such that σ_1 grows in the “North” and σ_2 in the “East” direction, which by Fig. 2 generally aligns with how we have presented the distance-embedding map.

Whereas CON and IND still mark the left corners of our map, the other two corners lead us to new characteristic instances. For the lower-right corner, we refine our definition of separability since SEP (with $\sigma_1 = \sigma_2 = 1$) only lies on the lower boundary if $n = m$. If m is a proper multiple of n , the lower-left corner is instead inhabited by *wide separability*, in which every agent values m/n disjoint goods, giving equal value n/m to each of them. In Appendix D.1, we extend wide separability to $n \nmid m$ in two slightly different ways: one, WSEP, always lies on the right border while the other, WSEPF always lies on the lower border.

The final characteristic instance is *bicontention*, or BIC, in which half of the agents place all utility on one common good and half of the agents on a second common good. (For odd n , one agent places all value on a third good.) Since this instance combines highly demanded goods with sharply distinct utility vectors, it always lies on the right border and, for even n , is exactly located in the upper right corner.

The main results of this section address all four sides of the map. For each side, we bound the map by an inequality, and show that the inequality is sharp using our characteristic instances. For three of the sides, we give simple, necessary-and-sufficient conditions for an instance lying on the boundary. If n is even and divides m , as in the left subplot of Fig. 4, our characteristic instances lie exactly in the four corner points of the map, and we can exactly trace three of the four sides by interpolating between corner instances.

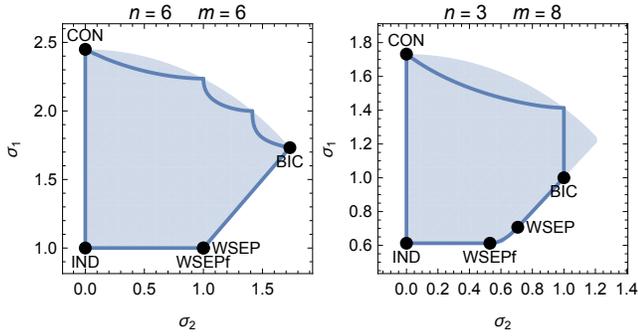


Figure 4: Explicit map for $\langle n, m \rangle = \langle 6, 6 \rangle$ and $\langle 3, 8 \rangle$. By Theorems 1 to 4, the map is contained in the shaded area. Lines trace interpolations between named instances (see Appendix D).

If these divisibility conditions do not hold, as illustrated in the Fig. 3 and the right subplot of Fig. 4, the characteristic instances lie in the corner up to rounding terms. Proofs of our characterizations tend to be short and cute, but are deferred to Appendix E.

Theorem 1 (“West”). σ_2 is at least 0. An instance lies on this boundary iff all agents have the same utility vector. In particular, IND, CON, and their convex combinations lie on this boundary.

Theorem 2 (“South”). σ_1 is at least $\sqrt{n/m}$. An instance lies on this boundary iff all columns of its utility matrix have an equal sum (namely, n/m). In particular, IND, WSEPF, and their convex combinations lie on this boundary.

Theorem 3 (“North”). σ_1 is at most $\sqrt{n - \sigma_2} \leq \sqrt{n}$. An instance lies on this boundary iff each agent values a single good, and if at most two goods are valued by any agent. In particular, IND and, if n is even, BIC lie on this boundary.

Theorem 4 (“East”). σ_2 is at most σ_1 . If U , after row and column permutation, has the block matrix structure $\begin{pmatrix} A & 0 & 0 \\ 0 & A & 0 \\ 0 & 0 & B \end{pmatrix}$ for rectangular matrices A, B and $\sigma_1(A) \geq \sigma_1(B)$, this is sufficient for lying on the boundary. (If B has height 0, we set $\sigma_1(B) = 0$.) In particular, WSEP, BIC, and a suitable interpolation lie on this boundary.

We conclude the theoretical discussion by pointing out that existing and future results in the theory of nonnegative random matrices have implications for our explicit map. For example, consider a random process in which a single utility vector is drawn from a flat Dirichlet distribution and then duplicated for all agents (thus, $\sigma_2 = 0$). For this distribution over allocation instances, Crumpton, Fyodorov, and Vivo (2022) recently derived that $\mathbb{E}[\sigma_1^2] = \frac{2n}{n+1}$ as well as formulas for σ_1^2 ’s higher moments. Brito, Dumitriu, and Harris (2022) study a random process, which, for fixed integers $d_2 \geq d_1 \geq 3$, an instance is uniformly chosen in which each agent values d_1 goods at value $1/d_1$, and each good is valued by d_2 agents. In these instances, σ_1 always equals $\sqrt{n/m}$,

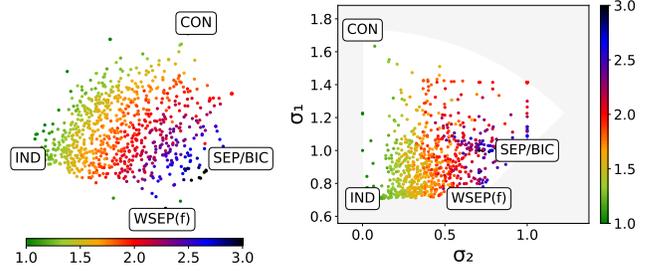


Figure 5: Distribution of maximum utilitarian welfare, on 3×6 instances, on our distance-embedding (left) and explicit (right) map.

and the authors show that, as $m, n \rightarrow \infty$, σ_2 converges to $(\sqrt{d_1 - 1} + \sqrt{d_2 - 1})/d_1$ in probability.

4.3 Empirical Comparison

Comparing the explicit map to our distance-embedding map (see Fig. 5 for an example feature), we see that the two maps have a similar layout and communicate similar information overall. In Appendix F, we provide extensive diagrams showing that this similarity extends to other features, the identifiability of instance sources, and to the 5×5 map as well. The specific feature being mapped is the maximum utilitarian welfare achievable by any allocation. Like the maximum Nash welfare before, it is higher the closer the instances are to our different variants of (wide) separability. On the one hand, a general advantage of the distance-embedding map is that it tends to fill the entire map at a relatively uniform density, whereas the explicit map clusters some instances very densely, for example close to indifference and on the $\sigma_2 = 1$ line. On the other hand, the clusters identified by the explicit map tend to indeed be very homogeneous along the features we investigate, which means that the explicit map picks up on meaningful patterns in the data.

5 Conclusion

In this paper, we introduced two, closely related maps for allocation instances. We hope that our exploration initiates discussions about which kinds of assumptions on realistic allocation instances are reasonable to make, and how fair-division theory can leverage these assumptions to provide algorithms with stronger fairness properties for the bulk of practical allocation instances.

A key limitation of our work so far is that preference data for fair-division problems is much less accessible — and the instances that are accessible more small-scale — compared to e.g. election data (Mattei and Walsh 2013). As a community, we should make efforts to collect, and to make centrally available, such datasets. More and larger-scale real-world data could show whether distance-embedding maps will continue to align with the explicit map we have constructed, or whether it will reveal additional key dimensions among which allocation instances differ in practice.

References

- Amanatidis, G.; Aziz, H.; Birmpas, G.; Filos-Ratsikas, A.; Li, B.; Moulin, H.; Voudouris, A. A.; and Wu, X. 2023. Fair Division of Indivisible Goods: Recent Progress and Open Questions. *Artificial Intelligence*, 322: 103965.
- Amanatidis, G.; Birmpas, G.; Filos-Ratsikas, A.; Hollender, A.; and Voudouris, A. A. 2021. Maximum Nash Welfare and Other Stories about EFX. *Theoretical Computer Science*, 863: 69–85.
- Amanatidis, G.; Markakis, E.; Nikzad, A.; and Saberi, A. 2017. Approximation Algorithms for Computing Maximin Share Allocations. *ACM Transactions on Algorithms (TALG)*, 13(4): 1–28.
- Anonymized. 2023. Personal Communication.
- Bai, Y.; Feige, U.; Gözl, P.; and Procaccia, A. D. 2022. Fair Allocations for Smoothed Utilities. In *Proceedings of the ACM Conference on Economics and Computation (EC)*.
- Bai, Y.; and Gözl, P. 2022. Envy-Free and Pareto-Optimal Allocations for Agents with Asymmetric Random Valuations. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 53–59.
- Benade, G.; Itzhak, N.; Shah, N.; Procaccia, A. D.; and Gal, Y. 2018. Efficiency and Usability of Participatory Budgeting Methods.
- Benadè, G. 2023. Personal Communication.
- Blum, A.; Hopcroft, J. E.; and Kannan, R. 2020. *Foundations of Data Science*. New York, NY: Cambridge University Press, first edition.
- Boehmer, N.; Bredereck, R.; Faliszewski, P.; Niedermeier, R.; and Szufa, S. 2021. Putting a Compass on the Map of Elections. In *Proceedings of IJCAI-2021*, 59–65.
- Boehmer, N.; Faliszewski, P.; Niedermeier, R.; Szufa, S.; and Was, T. 2022. Understanding Distance Measures Among Elections. In *Proceedings of IJCAI-2022*, 102–108.
- Boehmer, N.; Heeger, K.; and Szufa, S. 2023. A Map of Diverse Synthetic Stable Roommates Instances. 1003–1011.
- Bredereck, R.; Figiel, A.; Kaczmarczyk, A.; Knop, D.; and Niedermeier, R. 2021. High-Multiplicity Fair Allocation Made More Practical. 260–268.
- Brito, G.; Dumitriu, I.; and Harris, K. D. 2022. Spectral Gap in Random Bipartite Biregular Graphs and Applications. *Combinatorics, Probability and Computing*, 31(2): 229–267.
- Budish, E. 2011. The Combinatorial Assignment Problem: Approximate Competitive Equilibrium from Equal Incomes. *Journal of Political Economy*, 119(6): 1061–1103.
- Budish, E.; Cachon, G. P.; Kessler, J. B.; and Othman, A. 2017. Course Match: A Large-Scale Implementation of Approximate Competitive Equilibrium from Equal Incomes for Combinatorial Allocation. *Operations Research*, 65(2): 314–336.
- Budish, E.; Gao, R.; Othman, A.; Rubinstein, A.; and Zhang, Q. 2023. Practical Algorithms and Experimentally Validated Incentives for Equilibrium-Based Fair Division (A-CEEI). arxiv:2305.11406.
- Campbell, D. E.; and Kelly, J. S. 2002. Impossibility Theorems in the Arrowian Framework. In Arrow, K. J.; Sen, A. K.; and Suzumura, K., eds., *Handbook of Social Choice and Welfare*, volume 1, 35–94. Elsevier.
- Caragiannis, I.; Kurokawa, D.; Moulin, H.; Procaccia, A. D.; Shah, N.; and Wang, J. 2019. The Unreasonable Fairness of Maximum Nash Welfare. *ACM Transactions on Economics and Computation*, 7(3): 1–32.
- Chaudhury, B. R.; Garg, J.; and Mehlhorn, K. 2020. EFX Exists for Three Agents. In *Proceedings of the ACM Conference on Economics and Computation (EC)*, 1–19.
- Crumpton, M. J.; Fyodorov, Y. V.; and Vivo, P. 2022. Statistics of the Largest Eigenvalues and Singular Values of Low-Rank Random Matrices with Non-Negative Entries.
- Dickerson, J. P.; Goldman, J.; Karp, J.; Procaccia, A. D.; and Sandholm, T. 2014. The Computational Rise and Fall of Fairness. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.
- Faliszewski, P.; Kaczmarczyk, A.; Sornat, K.; Szufa, S.; and Wąs, T. 2023. Diversity, Agreement, and Polarization in Elections. In *Proceedings of IJCAI-2023*, 2684–2692.
- Faliszewski, P.; Skowron, P.; Slinko, A.; Szufa, S.; and Talmon, N. 2019. How Similar Are Two Elections? In *Proceedings of AAAI-2019*, 1909–1916.
- Farhadi, A.; Ghodsi, M.; Hajiaghayi, M. T.; Lahaie, S.; Pennock, D.; Seddighin, M.; Seddighin, S.; and Yami, H. 2019. Fair Allocation of Indivisible Goods to Asymmetric Agents. *Journal of Artificial Intelligence Research*, 64: 1–20.
- Goldman, J.; and Procaccia, A. D. 2014. Spliddit: Unleashing Fair Division Algorithms. *ACM SIGecom Exchanges*, 13(2): 41–46.
- Kurokawa, D.; Procaccia, A. D.; and Wang, J. 2016. When Can the Maximin Share Guarantee Be Guaranteed? In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 523–529.
- Lipton, R. J.; Markakis, E.; Mossel, E.; and Saberi, A. 2004. On Approximately Fair Allocations of Indivisible Goods. In *Proceedings of the ACM Conference on Economics and Computation (EC)*, 125–131.
- Manurangsi, P.; and Suksompong, W. 2019. When Do Envy-Free Allocations Exist? In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2109–2116.
- Manurangsi, P.; and Suksompong, W. 2021. Closing Gaps in Asymptotic Fair Division. *SIAM Journal on Discrete Mathematics*, 35(2): 668–706.
- Mattei, N.; and Walsh, T. 2013. PrefLib: A Library for Preferences. In *Proceedings of ADT-2013*, 259–270.
- Plaut, B.; and Roughgarden, T. 2020. Almost Envy-Freeness with General Valuations. *SIAM Journal on Discrete Mathematics*, 34(2): 1039–1068.
- Roughgarden, T. 2020. Distributional Analysis. In Roughgarden, T., ed., *Beyond the Worst-Case Analysis of Algorithms*, 167–188. Cambridge University Press, 1 edition.
- Shafer, W.; and Sonnenschein, H. 1982. Market Demand and Excess Demand Functions. In *Handbook of Mathematical Economics*, volume 2, 671–693. Elsevier.

Shah, N. 2022. Personal Communication.

Szufa, S.; Faliszewski, P.; Janeczko, L.; Lackner, M.; Slinko, A.; Sornat, K.; and Talmon, N. 2022. How to Sample Approval Elections? In *Proceedings of IJCAI-2022*, 496–502.

Szufa, S.; Faliszewski, P.; Skowron, P.; Slinko, A.; and Talmon, N. 2020. Drawing a Map of Elections in the Space of Statistical Cultures. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 1341–1349.

user1551 on Math Stack Exchange. ??? Characterize stochastic matrices such that max singular value is less or equal one. Mathematics Stack Exchange.

Varian, H. R. 1974. Equity, Envy, and Efficiency. *Journal of Economic Theory*, 9(1): 63–91.

A Alternative Notation

In this appendix, we partially use the following alternative notation which easily translates into that of our allocation instances.

Consider a set of resources \mathcal{R} , a set of agents \mathcal{A} . Let, for each agent $a \in \mathcal{A}$, $u_a: \mathcal{R} \rightarrow \mathbb{N}$ be a 's utility function. For *bundle* $R \subseteq \mathcal{R}$ of resources and some agent $a \in \mathcal{A}$, we denote by $u_a(R) := \sum_{r \in R} u_a(r)$, the utility that a gives to bundle R . Hence, we study cardinal monotonic additive preferences. An *allocation task* is a triple $(\mathcal{A}, \mathcal{R}, \mathcal{U})$, where \mathcal{U} is a collection of utility functions, one per agent, as specified in the paragraph above.

B Details on Datasets and Experiments

We combined various statistical cultures and the real-world data to construct datasets that we focus on in the paper. Due to the obvious tradeoff between the computation time of our experiment and the size of the datasets, we kept the number of synthetic instances rather low to better observe the real-life related instances.

For the 5×5 *small* dataset consisting of instances with 5 agents and 5 goods, we generated 4 instances according to: 1- and 2-Euclidean models; the attributes models with 2 and 5 attributes; the resampling and Dirichlet-resampling models with all values $\{0.2, 0.4, 0.6\}$ of parameter p and all values $\{0.2, 0.8\}$ of parameter ϕ . Furthermore, we added 20 instances sampled (as described in the previous section) from the Island data, 20 instances from the Candies data, and all 16 Spliddit instances. Finally, we put the respective CON, IND, SEP, WSEP, and BCON instance. To verify our experiment for more real-world-inspired data, we also constructed the 5×5 *large* dataset, where we increased the number of Island and Candies instances to 500 (we wanted to keep the Spliddit instances as they are provided, without sampling, so we could not put more instances to the large dataset). Analogously, we constructed the 3×6 *small* and 3×6 *large* datasets consisting of instances with 3 agents and 6 goods. The only exception being that in the 3×6 large dataset, we took 250 instance of the Island and Candies data. However, since there were enough Spliddit instances of this size, we also took 250 Spliddit instances.

We generated each of the datasets multiple times (note that generating real-life inspired data is a random process) and repeat all our experiments. The obtained results were qualitatively the same. Due to the limitations of the Spliddit data, where there is very few instances with more than 5 agents, we decided not to present experiments for larger number of agents.

C Deferred Details from Section 2

C.1 Valuation Distance

Denote $\Pi(\mathcal{A}, \mathcal{A}')$ by $\Pi_{\mathcal{A}}$ and $\Pi(\mathcal{R}, \mathcal{R}')$ by $\Pi_{\mathcal{m}}$. Then, consider some $\pi_a \in \Pi_{\mathcal{A}}$ and some $\pi_{goods} \in \Pi_{\mathcal{m}}$, which we call, respectively, an *agent matching* and a *good matching*. For some distance δ on nonnegative real numbers, we let

$$D_{\delta}(\mathcal{T}, \mathcal{T}', \pi_a, \pi_{goods}) := \sum_{a \in \mathcal{A}} \sum_{r \in \mathcal{R}} \delta(u_a(r), u_{\pi_a(a)}(\pi_{goods}(r)))$$

and refer to $D_{\delta}(\mathcal{T}, \mathcal{T}', \pi_a, \pi_a)$ as the δ -distance between \mathcal{T} and \mathcal{T}' witnessed by π_a and π_{goods} . The δ -distance between \mathcal{T} and \mathcal{T}' , denoted by $d_{\delta}(\mathcal{T}, \mathcal{T}')$, is then the minimal δ -distance between \mathcal{T} and \mathcal{T}' witnessed over all pair of matchings; formally:

$$d_{\delta}(\mathcal{T}, \mathcal{T}') := \min_{\pi'_a \in \Pi_{\mathcal{A}}, \pi'_{goods} \in \Pi_{\mathcal{m}}} D(\mathcal{T}, \mathcal{T}', \pi'_a, \pi'_{goods}).$$

We are now ready to define our *valuation distance* that, intuitively, is the smallest sum of difference in agents valuations of the goods achievable over all possible matchings of agents and goods.

Definition 2. Given two task allocations $\mathcal{T} = (\mathcal{A}, \mathcal{R}, \mathcal{U})$ and $\mathcal{T}' = (\mathcal{A}', \mathcal{R}', \mathcal{U}')$, its *valuation distance* $d_v(\mathcal{T}, \mathcal{T}')$ is:

$$d_v(\mathcal{T}, \mathcal{T}') := d_{\ell_1}(\mathcal{T}, \mathcal{T}') := \min_{\pi_a, \pi_{goods}} \sum_{a \in \mathcal{A}} \sum_{r \in \mathcal{R}} |u_a(r) - u_{\pi_a(a)}(\pi_{goods}(r))|.$$

It is easy to see that a valuation is an isomorphic distance. Naturally, if the tasks are isomorphic, then there exists some pair of agent and good matchings that witness distance 0. On the other hand, if there is no such pair, there is no possibility that the valuation is 0. The property of being an isomorphic distance, however, comes at a cost of computational intractability.

Theorem 5. Given two task allocations \mathcal{T} and \mathcal{T}' and an real number d , deciding whether $d_v(\mathcal{T}, \mathcal{T}') \leq d$ is **NP-hard**.

Proof. We give a polynomial-time many-one reduction from an NP-hard problem d_{Spear} -ISOMORPHIC DISTANCE. In this problem we are given two ordinal elections $E = (C, V)$ and $E' = (C', V')$ such that $|C| = |C'|$ and $|V| = |V'|$ and an integer k . Assuming that for some voter a and candidate b (where both a and b are part of the same election), we denote by $\text{pos}_a(b)$ the position of candidate b according to the ranking of a , we ask whether there exist two permutations $\rho: C \rightarrow C'$ and $\phi: V \rightarrow V'$ such that

$$D(\rho, \phi) := \sum_{v \in V} \sum_{c \in C} \left| \text{pos}_v(c) - \text{pos}_{\phi(v)}(\rho(c)) \right| \leq k.$$

Given the instance I of d_{Spear} -ISOMORPHIC DISTANCE as described above, our reduction constructs an instance I' of our problem as follows. We construct allocation task \mathcal{T} using election E from the original instance as follows. Task \mathcal{T} consists of $|V|$ agents, denoted as \mathcal{A} , $a(v_1)$ to $a(v_{|V|})$ representing voters and $|C|$ goods, denoted as \mathcal{R} , $r(c_1)$ to $r(c_{|C|})$ representing candidates. Taking a normalizing factor $F = 1 + 2 + \dots + |C| = \binom{|C|}{2}$, for each voter $v \in V$ we set the corresponding agent's utility function to be $u_{a(v)}(r(c)) = \text{pos}_v(c)/F$, for each $c \in C$. It can be easily verified that the values of the utility function of each agent in \mathcal{T} sum to 1. By analogously constructing allocation task \mathcal{T}' using election E' and setting the distance d in question regarding instance I to value $d := k/F$, we obtain a new instance I' of our problem.

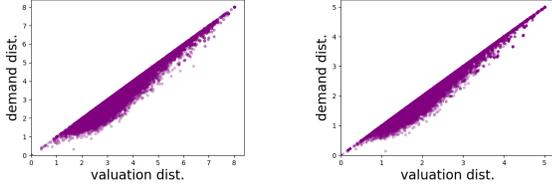


Figure 6: Correlation between our distances for the small 5x5 dataset (left) and the small 3x6 dataset (right).

We show that for each pair of permutations $\rho: C \rightarrow C'$ and $\phi: V \rightarrow V'$ such that $D(\rho, \phi) \leq k$, there are two permutations π_a and π_{goods} such that witness that $d_v(\mathcal{T}, \mathcal{T}') \leq d$. Since we also show that the opposite direction is true as well, we obtain that the reduction is correct.

Suppose that we have ρ and ϕ that meet the above assumption. Consider the following π_a and π_{goods} . For each voter $v \in V$ and candidate $c \in C$, let $\pi_a(a(v)) = a(\phi(v))$ and $\pi_{goods}(r(c)) = r(\phi(c))$. In words, permutation π_a maps agents exactly as permutation ϕ maps their respective voters, and so does permutation π_{goods} with respect to goods and candidates. Now, in the series of algebraic transformations, let us analyze the relation of $D(\rho, \phi)$ and d :

$$\begin{aligned} D(\rho, \phi)/F &= \frac{\sum_{v \in V} \sum_{c \in C} \left| \text{pos}_v(c) - \text{pos}_{\phi(v)}(\rho(c)) \right|}{F} = \\ &= \sum_{v \in V} \sum_{c \in C} \left| \frac{\text{pos}_v(c)}{F} - \frac{\text{pos}_{\phi(v)}(\rho(c))}{F} \right| = \\ &= \sum_{v \in V} \sum_{c \in C} \left| u_{a(v)}(r(c)) - u_{\phi(v)}(\rho(c)) \right| = \\ &= \sum_{a' \in \mathcal{A}} \sum_{r' \in \mathcal{R}} \left| u_{a'}(r') - u_{\pi_a(a')}(\pi_{goods}(r')) \right| = \\ &= \sum_{a' \in \mathcal{A}} d_{\ell_1}(u_{a'}, \pi_{goods}(u_{\pi_a(a')})) = d. \end{aligned}$$

So, clearly, if $D(\rho, \phi) \leq k$, then $d_v(\mathcal{T}, \mathcal{T}')$ witnessed by π_a and π_{goods} is smaller than $k/F = d$. On the other hand, if there exist π_a and π_{goods} that witness $d_v(\mathcal{T}, \mathcal{T}') \leq d$, then one can construct ρ and ϕ for which $D(\rho, \phi) \leq dF = k$. \square

The computational hardness of the task of computing the valuation distance comes from the fact that one seeks an optimal value depending on two matchings simultaneously. It turns out that this intuitive understanding can be well supported by a formal claim. We show that for a given either the agent matching or the good matching, the optimal value of the distance as witnessed by the given matching can be computed in polynomial-time.

Theorem 6. *Given two task allocations $\mathcal{T}, \mathcal{T}'$, a (fixed) agent matching π_a , a real number d , deciding whether $d_v(\mathcal{T}, \mathcal{T}')$ as witnessed by π_a is at most d is polynomial-time solvable. The same holds for the case of a given good matching.*

Proof. For two allocation tasks $\mathcal{T} = (\mathcal{A}, \mathcal{R}, \mathcal{U})$, $\mathcal{T}' = (\mathcal{A}', \mathcal{R}', \mathcal{U}')$, with $\mathcal{R} = \{r_1, r_2, \dots, r_n\}$, $\mathcal{R}' = \{r'_1, r'_2, \dots, r'_n\}$, and an agent matching $\pi_a \in \Pi_{\mathcal{A}}$, we give a polynomial-time algorithm that computes a good matching π_{goods} minimizing

$$D(\pi_{goods}) := \sum_{a \in \mathcal{A}} \sum_{r \in \mathcal{R}} \left| u_a(r) - u_{\pi_a(a)}(\pi_{goods}(r)) \right|.$$

In words, the algorithm computes the minimal achievable distance as witnessed by the given agent matching π_a .

The algorithm constructs a complete bipartite weighted graph G consisting of vertices $x_1, x_2, \dots, x_{|\mathcal{R}|}$ of one partition and consisting of vertices $y_1, y_2, \dots, y_{|\mathcal{R}'|}$ of the other partition. For each pair $i \in [|\mathcal{R}|]$, $j \in [|\mathcal{R}'|]$, the weight $w(\{x_i, y_j\})$ of edge $\{x_i, y_j\}$ is $\sum_{a \in \mathcal{A}} \left| u_a(r_i) - u_{\pi_a(a)}(r'_j) \right|$. Finally, the algorithm looks for a minimum weight perfect matching (which always exists) in G .

Let M be some perfect matching in G . Clearly, this perfect matching corresponds to exactly one good matching $\pi'_{goods} \in \Pi_m$. Let us now compute the weight $w(M)$ of M :

$$\begin{aligned} w(M) &= \sum_{\{x_i, y_j\} \in M} \sum_{a \in \mathcal{A}} \left| u_a(r_i) - u_{\pi_a(a)}(r'_j) \right| = \\ &= \sum_{a \in \mathcal{A}} \sum_{r \in \mathcal{R}} \left| u_a(r) - u_{\pi_a(a)}(\pi'_{goods}(r)) \right| = D(\pi_{goods}). \end{aligned}$$

Since our algorithm finds the minimum-weight matching, the correctness follows.

The algorithm runs in polynomial time because finding a minimum-weight matching is well-known polynomial-time solvable task and building the bipartite graph is quadratic with respect to the number of goods (which is polynomially bounded in the input size). The proof for the case of a given good matching is analogous. \square

C.2 Demand Distance

For some good $r \in \mathcal{R}$, let $\text{dem}(r) = (u_{a_1}(r), u_{a_2}(r), \dots, u_{a_n}(r))$ be a *demand vector* of r . By $\overrightarrow{\text{dem}}(r)$, let us denote the *sorted demand vector* that results from ordering $\text{dem}(r)$ descending. For the sake of readability, we often drop “sorted,” when it is implied by the context. Consider some good matching $\pi_{goods} \in \Pi_m$. For some distance δ on nonnegative real vectors, let

$$\hat{D}_{\delta}(\mathcal{T}, \mathcal{T}', \pi_{goods}) := \sum_{r \in \mathcal{R}} \delta(\overrightarrow{\text{dem}}(r), \overrightarrow{\text{dem}}(\pi_{goods}(r))).$$

Similarly to the valuation distance, the *demand distance* is then the minimum of the above-defined formula over all possible good matchings taking δ to be d_{ℓ_1} .

Definition 3. Given two task allocations $\mathcal{T} = (\mathcal{A}, \mathcal{R}, \mathcal{U})$ and $\mathcal{T}' = (\mathcal{A}', \mathcal{R}', \mathcal{U}')$, its *demand distance* $d_d(\mathcal{T}, \mathcal{T}')$ is:

$$\begin{aligned} d_v(\mathcal{T}, \mathcal{T}') &:= \min_{\pi_{goods}} \hat{D}_{d_{\ell_1}}(\mathcal{T}, \mathcal{T}', \pi_{goods}) := \\ &= \min_{\pi_{goods}} \sum_{r \in \mathcal{R}} d_{\ell_1}(\overrightarrow{\text{dem}}(r), \overrightarrow{\text{dem}}(\pi_{goods}(r))). \end{aligned}$$

$$\text{IND}_m := \begin{bmatrix} 1/m & 1/m & \cdots & 1/m \\ 1/m & 1/m & \cdots & 1/m \\ \vdots & \vdots & \ddots & \vdots \\ 1/m & 1/m & \cdots & 1/m \end{bmatrix} \quad \text{SEP}_m := \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \quad \text{CON}_m := \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \cdots & 0 \end{bmatrix}$$

Figure 7: Matrices representing characteristic instances: Indifference (IND), Separability (SEP), and Contention (CON).

Observe that to compute the demand distance between two allocations tasks, one needs to find (only) a single optimal matching. Hence, this distance can be computed in polynomial using standard matching algorithms.

Theorem 7. *Given two task allocations \mathcal{T} and \mathcal{T}' and a real number d , deciding whether $d_d(\mathcal{T}, \mathcal{T}') \leq d$ is polynomial-time solvable.*

Proof. We give an algorithm that first constructs a weighted bipartite graph (representing the task of computing the demand distance) and then computes its minimum weight perfect matching, which represents the optimal good matching.

To be specific, our algorithm proceeds as follows. For each $r \in \mathcal{R} \cup \mathcal{R}'$, it first computes $\overrightarrow{\text{dem}}(r)$. Then, it constructs a bipartite graph G with one partition consisting of the goods \mathcal{R} and the other one of the goods \mathcal{R}' . For every pair $(r, r') \in \mathcal{R} \times \mathcal{R}'$, the algorithm adds an edge $\{r, r'\}$ to G and sets its weight $w(r, r') := d_{\ell_1}(\overrightarrow{\text{dem}}(r), \overrightarrow{\text{dem}}(r'))$. Finally, the algorithm finds a minimum weight perfect matching, say M , of G .

Since M is a perfect matching (and $|\mathcal{R}| = |\mathcal{R}'|$), it is clear that M represents a good matching π_{goods} such that for each $\{r, r'\} \in M$, $\pi_{\text{goods}}(r) = \pi_{\text{goods}}(r')$. Hence, the total weight $w(M)$ of M can be expressed as

$$w(M) := \sum_{\{r, r'\} \in M} d_{\ell_1}(\overrightarrow{\text{dem}}(r), \overrightarrow{\text{dem}}(r')) = \sum_{r \in \mathcal{R}} d_{\ell_1}(\overrightarrow{\text{dem}}(r), \overrightarrow{\text{dem}}(\pi_{\text{goods}}(r))).$$

As a result, a minimum weight perfect matching in graph G yields a good matching that witnesses the demand distance and the weight of this matching is exactly the requested demand distance.

Note that computing a minimum weight perfect matching is polynomial-time solvable. Thus, our algorithm also runs in polynomial time. \square

Note that the procedure described in the proof of Theorem 7 is constructive and, in fact, solves the optimization variant of the problem of computing the demand distance between two allocation tasks.

Proposition 1. *Let U and U' be two allocations instances with n agents and m goods. Then, the valuation distance $d_v(U, U')$ and the demand distance $d_d(U, U')$ are at most $2n - \frac{2n}{m}$.*

allocation task \mathcal{T}					allocation task \mathcal{T}'				
	r_1	r_2	r_3	r_4		r'_1	r'_2	r'_3	r'_4
a_1	2	4	6	8	a'_1	2	4	6	8
a_2	3	3	6	8	a'_2	3	3	6	8
a_3	6	8	6	0	a'_3	6	8	0	6
a_4	8	6	0	6	a'_4	8	6	6	0

Demand vectors for \mathcal{T}					Demand vectors for \mathcal{T}'				
	r_1	r_2	r_3	r_4		r'_1	r'_2	r'_3	r'_4
	8	8	6	8		8	8	6	8
	6	6	6	8		6	6	6	8
	3	4	6	6		3	4	6	6
	2	3	0	0		2	3	0	0

Figure 8: The task allocations demonstrating a zero demand distance (note the demand vectors and apply matching r_i to r'_i , for all $i \in \{1, 2, 3, 4\}$) but a non-negative valuation distance (to verify, perform an exhaustive check).

D Deferred Details from Section 4.2

D.1 Wide Separability

We propose two ways of generalizing this instance to m not divisible by n , setting $\ell := \lfloor m/n \rfloor$: In the first variant, WSEP, each agent values ℓ goods at $1/\ell$, and thus $m \bmod n$ goods have no value for any agent. In the second variant, WSEPF, each agent values the ℓ goods that no other agent values at n/m , and the final $m \bmod n$ goods have a value of $\frac{1-\ell n/m}{m \bmod n}$ for every agent. WSEP always lies on the East and WSEPF on the South border; if $n \mid m$, they coincide, meeting in the South-East corner.

E Deferred Proofs

Theorem 1 (“West”). σ_2 is at least 0. An instance lies on this boundary iff all agents have the same utility vector. In particular, IND, CON, and their convex combinations lie on this boundary.

Proof. Singular values are always nonnegative real numbers. It is well-known that the second singular value is 0 iff the matrix has rank 1 (or zero), i.e., if all rows are linearly dependent. Since all row sums are 1, this is equivalent to all rows being identical. Since IND and CON each have only identical rows, they have $\sigma_2 = 0$. The same property is inherited by their convex combinations, which, by the continuity of the singular values, must trace the entire boundary between IND and CON. \square

Theorem 2 (“South”). σ_1 is at least $\sqrt{n/m}$. An instance lies on this boundary iff all columns of its utility matrix have an equal sum (namely, n/m). In particular, IND, WSEPF, and their convex combinations lie on this boundary.

Proof. By Eq. (1),

$$\begin{aligned}\sigma_1 &= \max_{\vec{v}_1 \in \mathbb{R}^m, \|\vec{v}_1\|=1} \|U \vec{v}_1\| \\ &\geq \left\| U \begin{pmatrix} 1/\sqrt{m} \\ 1/\sqrt{m} \\ \vdots \\ 1/\sqrt{m} \end{pmatrix} \right\| \\ &= \left\| \begin{pmatrix} 1/\sqrt{m} \\ 1/\sqrt{m} \\ \vdots \\ 1/\sqrt{m} \end{pmatrix} \right\| && \text{(by row stochasticity)} \\ &= \sqrt{n/m}.\end{aligned}$$

Note that the vector being multiplied with U in the second row has dimension m (thus norm 1), but the vector in the third row has dimension n .

We now show that, whenever this inequality is tight, all column sums must be n/m . This extends a widely known proof (presentation adapted from [user1551 on Math Stack Exchange](#)) showing that, among square matrices, a row-stochastic matrix has $\sigma_1 = 1$ iff it is doubly stochastic, i.e., its column sums are also equal to one. In the following,

we set $\mathbf{1}_t$ to denote the vector $\begin{pmatrix} 1 \\ \vdots \\ i \end{pmatrix} \in \mathbb{R}^t$, and denote the vector dot product by $\langle \cdot, \cdot \rangle$.

$$\begin{aligned}m &= \frac{m}{n} \langle \mathbf{1}_n, \mathbf{1}_n \rangle \\ &= \frac{m}{n} \langle \mathbf{1}_n, U \mathbf{e}_m \rangle && \text{(by row stochasticity)} \\ &= \left\langle U^T \left(\frac{m}{n} \mathbf{1}_n \right), \mathbf{1}_m \right\rangle \\ &\leq \left\| U^T \left(\frac{m}{n} \mathbf{1}_n \right) \right\| \cdot \|\mathbf{1}_m\| && \text{(Cauchy-Schwartz)} \\ &\leq \sigma_1(U^T) \frac{m}{n} \|\mathbf{1}_n\| \|\mathbf{1}_m\| \\ &&& \text{(property of Operator Norm)} \\ &= \sigma_1(U) \frac{m}{n} \|\mathbf{1}_n\| \|\mathbf{1}_m\| && (\sigma_1(A) = \sigma_1(A^T)) \\ &= \sqrt{\frac{n}{m}} \frac{m}{n} \|\mathbf{1}_n\| \|\mathbf{1}_m\| && \text{(by assumption)} \\ &= \sqrt{\frac{n}{m}} \frac{m}{n} \sqrt{n} \sqrt{m} \\ &= m.\end{aligned}$$

Since both ends of the inequality chain are equal, all terms along the chain must be equal. Since the Cauchy-Schwartz step was an equality, we know that $U^T (m/n \mathbf{1}_n)$ is a scalar multiple of $\mathbf{1}_m$; since $\|U^T (m/n \mathbf{1}_n)\| \|\mathbf{1}_m\| = m$, we now that $\|U^T (m/n \mathbf{1}_n)\| = \sqrt{m}$; finally, we now that $U^T (m/n \mathbf{1}_n)$ is nonnegative. Taking these facts together, we conclude that $U^T (m/n \mathbf{1}_n) = \mathbf{1}_m$, which means that all column sums of U are equal to n/m .

It is easy to see that IND, WSEPF, and their linear interpolations all have column sums of n/m , which concludes the claim. \square

Theorem 3 (“North”). σ_1 is at most $\sqrt{n - \sigma_2} \leq \sqrt{n}$. An instance lies on this boundary iff each agent values a single good, and if at most two goods are valued by any agent. In particular, IND and, if n is even, BIC lie on this boundary.

Proof. Setting $\sigma_1, \sigma_2, \dots, \sigma_r$ for the singular values of some matrix, it is well known that $\sum_{t=1}^r \sigma_t^2$ equals the square of the matrix’ Frobenius norm, i.e., equals the sum of squares across the entries of the matrix. That is, for a matrix U ,

$$\sum_{t=1}^r \sigma_t^2 = \sum_{i=1}^n \sum_{j=1}^m u_{i,j}^2.$$

Since all entries of our matrix are between 0 and 1, the i th row’s contribution to the right-hand side is $\sum_{j=1}^m u_{i,j}^2 \leq \sum_{j=1}^m u_{i,j} = 1$, and this inequality is tight iff agent i values one item at 1 and all others at 0. It follows that

$$\sigma_1^2 + \sigma_2^2 \leq \sum_{t=1}^r \sigma_t^2 \leq n,$$

where the inequality is tight exactly iff (a) all agents single-mindedly value a single good (which makes the second inequality tight) and (b) all σ_t for $t \geq 3$ are zero. Part (b) is the case iff U has rank at most 2. Assuming part (a), U ’s rank is exactly the number of distinct goods which some agent values single-mindedly. Taking square, we obtain the desired inequality with its necessary-and-sufficient condition.

Clearly, this condition is satisfied by IND and by BIC if n is even. Note that there is a natural way to interpolate between these two, where one good is single-mindedly valued by t agents and a second good by $n - t$ agents. Clearly, each of these points lies on the boundary. If one linearly interpolates between successive values of t , the interpolations have one agent who values two items, which removes this interpolation point from the boundary. But of course this operation does not move the instance far from the boundary. Figure 4 shows this interpolation as the blue line following the upper boundary of the map. \square

Theorem 4 (“East”). σ_2 is at most σ_1 . If U , after row and column permutation, has the block matrix structure $\begin{pmatrix} A & 0 & 0 \\ 0 & A & 0 \\ 0 & 0 & B \end{pmatrix}$ for rectangular matrices A, B and $\sigma_1(A) \geq \sigma_1(B)$, this is sufficient for lying on the boundary. (If B has height 0, we set $\sigma_1(B) = 0$.) In particular, WSEP, BIC, and a suitable interpolation lie on this boundary.

Proof. $\sigma_2 \leq \sigma_1$ holds by definition of the singular values.

Let U be a utility matrix with the block matrix structure from the theorem statement (since the singular values are invariant to row and column permutations, it suffices to consider such matrices directly). One important property of singular values we have not used yet is that the singular values of a matrix U are the square roots of the eigenvalues of the Gram matrix $U^T U$ (or, equivalently, for $U U^T$).

Given the block matrix structure,

$$\begin{pmatrix} A & 0 & 0 \\ 0 & A & 0 \\ 0 & 0 & B \end{pmatrix} \begin{pmatrix} A & 0 & 0 \\ 0 & A & 0 \\ 0 & 0 & B \end{pmatrix}^T = \begin{pmatrix} AA^T & 0 & 0 \\ 0 & AA^T & 0 \\ 0 & 0 & BB^T \end{pmatrix}.$$

It is well known that the eigenvalues of such a block diagonal matrix are simply the eigenvalues of the blocks AA^T , AA^T , and BB^T combined (with multiplicity preserved), which means that the singular values of U are just the singular values of A , A , and B combined. In particular, all singular values of A will appear in U at least twice. By $\sigma_1(A) \geq \sigma_1(B)$, the largest singular value is one of these duplicated singular values, which implies that $\sigma_1(U) = \sigma_2(U) = \sigma_1(A)$.

After reshuffling the columns, WSEP looks as follows (setting again $\ell := \lfloor m/n \rfloor$):

$$\begin{pmatrix} 1/\ell & \dots & 1/\ell & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/\ell & \dots & 1/\ell & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1/\ell & \dots & 1/\ell & 0 & 0 & \dots & 0 \\ & & & & & & & & & & \ddots & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/\ell & \dots & 1/\ell \end{pmatrix}$$

where the lines indicate the division of the blocks. The symmetry of the instance ensures that (unless B has height 0) $\sigma_1(B) = \sigma_1(A)$, so WSEP lies on this boundary.

For BIC, the situation is even simpler:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & \dots & 0 \\ \vdots & 0 & 0 & 0 & \dots & 0 \\ 1 & 0 & 0 & 0 & \dots & 0 \\ \hline 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & \vdots & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & 0 & \dots & 0 \\ \hline 0 & 0 & 1 & 0 & \dots & 0 \end{pmatrix}$$

In this case, the singular values are easy to calculate: $\sigma_1(A) = \sqrt{\lfloor n/2 \rfloor}$ which is at least $\sigma_1(B) = 1$.

Interpolating between both matrices without leaving the boundary is not straight-forward. For this, we first linearly interpolate from WSEP to SEP. For some $0 < \theta < 1$, this means that each agent approves one good at $\theta + (1 - \theta)/\ell$ and $\ell - 1$ goods at $(1 - \theta)/\ell$. For this interpolation, the block structure remains the same as the one discussed for WSEP and preserves the same symmetry, which is why the interpolation stays on the boundary.

The interpolation from SEP to BIC proceeds in discrete steps as follows: For $1 \leq r \leq \lfloor n/2 \rfloor$, r many agents only value the first good, r agents value only the second good, and the remaining agents value each a separate good. One verifies that this recovers SEP for $r = 1$ and BIC for $r = \lfloor n/2 \rfloor$, and that each of these stages can be represented in the block matrix shape, where A is a column of r ones, and B is the identity matrix with possibly zero columns attached to its right. Then, $\sigma_1(A) = \sqrt{r}$ which is at least $\sigma_1(B) = 1$, which means that this interpolation step lies on the boundary. By linearly interpolating between successive steps, one

obtains (after reordering) matrices of the shape

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ \vdots & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ \theta & 1 - \theta & 0 & 0 & 0 & 0 & \dots & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & \vdots & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & \theta & 1 - \theta & 0 & 0 & \dots & 0 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & \dots & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & \ddots & 0 \end{pmatrix},$$

which one verifies also lie on the boundary. As a result, we can continuously⁹ interpolate from WSEP to BIC while staying on the right boundary. It follows that this interpolation traces the entire right boundary between WSEP and BIC. \square

Proposition 1. *Let U and U' be two allocations instances with n agents and m goods. Then, the valuation distance $d_v(U, U')$ and the demand distance $d_d(U, U')$ are at most $2n - \frac{2n}{m}$.*

Lemma 1 (Rearrangement Inequality). *Let \vec{x} and \vec{y} be two size n vectors whose entries are sorted non-increasingly. Then, for every permutation σ of $[n]$, it holds that*

$$\sum_{i \in [n]} \min(x_i, y_i) \geq \sum_{i \in [n]} \min(x_i, y_{\sigma(i)}).$$

Proof. The proof works recursively as follows: Assume that the smallest entry of \vec{y} would not be in the last position n , but in the position i^* . Consider two cases.

First, $x_n \leq y_{i^*}$. Since by definition $y_n \geq y_{i^*}$, swapping y_{i^*} and y_n cannot decrease the overall sum.

Second, $x_n > y_{i^*}$. Then, also $x_{i^*} > y_{i^*}$ (by \vec{x} being non-increasing). That is, swapping y_{i^*} and y_n leads the the minimum in position n to become y_{i^*} (the previous minimum of position i^*). Yet, the minimum in position i^* is now at least $\min(y_n, x_{i^*})$, which must be at least $\min(y_{i^*}, x_n)$, since $y_n \geq y_{i^*}$ and $x_{i^*} \geq x_n$.

The last position is correct, that is, remove these entries and recurse. \square

Proof of Proposition 1. We first show the proof for the valuation distance. It is an adaption of the proof of Lemma 2 in the full version (arXiv:2205.00492 [cs.GT]) of [Boehmer et al. \(2022\)](#).

Assume towards a contradiction that we have two allocations instances U and U' with $d_v(U, U') > 2n - \frac{2n}{m}$. Assume w.l.g. that $d_v(U, U') = \sum_{i \in [n], j \in [m]} |u_{i,j} - u'_{i,j}|$, otherwise we could permute rows and column (relabel the goods and agents) of one of the allocation instances.

⁹While we reordered the matrix in between, this is just for exposition.

Observe that

$$\begin{aligned}
d_v(U, U') &= \sum_{i \in [n], j \in [m]} |u_{i,j} - u'_{i,j}| \\
&= \sum_{i \in [n], j \in [m]} (\max(u_{i,j}, u'_{i,j}) - \min(u_{i,j}, u'_{i,j})) \\
&= \sum_{i \in [n], j \in [m]} (u_{i,j} + u'_{i,j}) - 2 \sum_{i \in [n], j \in [m]} \min(u_{i,j}, u'_{i,j}) \\
&= \sum_{i \in [n]} 2 - 2 \sum_{i \in [n], j \in [m]} \min(u_{i,j}, u'_{i,j}) \\
&= 2n - 2 \sum_{i \in [n], j \in [m]} \min(u_{i,j}, u'_{i,j}). \tag{2}
\end{aligned}$$

If $d_v(U, U') > 2n - \frac{2n}{m}$, then it must hold that:

$$\sum_{i \in [n], j \in [m]} \min(u_{i,j}, u'_{i,j}) < n/m. \tag{3}$$

For each permutation σ of $[m]$, we have $d_v(U, U') \leq \sum_{i \in [n], j \in [m]} |u_{i,j} - u'_{i,\sigma(j)}|$ (this being incorrect would violate our assumption that $d_v(U, U') = \sum_{i \in [n], j \in [m]} |u_{i,j} - u'_{i,j}|$). Consequently, for every permutation σ of $[m]$, and reasoning analogously to Eq. (2) and Eq. (3), we get

$$\sum_{i \in [n], j \in [m]} \min(u_{i,j}, u'_{i,\sigma(j)}) < n/m.$$

Observe that if $x, y \in [0, 1]$, then it holds that $x \cdot y \leq \min(x, y)$. Since for each $i \in [n]$ and $j \in [m]$, we have $u_{i,j}, u'_{i,j} \in [0, 1]$, for each permutation σ of $[m]$, it holds that:

$$\sum_{i \in [n], j \in [m]} u_{i,j} \cdot u'_{i,\sigma(j)} < n/m. \tag{4}$$

We define a family $\Psi := \{\sigma^{(k)} \mid k \in [m]\}$ of permutations by $\sigma^{(k)}(j) := (j + k - 1 \bmod m) + 1$. By summing up Eq. (4) (on both sides) for each permutation from Ψ we obtain:

$$\sum_{\sigma \in \Psi} \sum_{i \in [n], j \in [m]} u_{i,j} \cdot u'_{i,\sigma(j)} < n/m \cdot |\Psi|,$$

which can be rearranged to

$$\sum_{i \in [n], j \in [m]} u_{i,j} \cdot \sum_{\sigma \in \Psi} (u'_{i,\sigma(j)}) < n/m \cdot |\Psi|. \tag{5}$$

Since $\sum_{j \in [m]} u_{i,j} = \sum_{j \in [m]} u'_{i,j} = 1, \forall i \in [n]$, it holds that:

$$\sum_{\sigma \in \Psi} (u'_{i,\sigma(j)}) = \sum_{k \in [m]} (u'_{i,\sigma^{(k)}(j)}) = \sum_{\ell \in [m]} u'_{i,\ell} = 1.$$

Hence (and with $|\Psi| = m$), from Eq. (5) we get

$$\sum_{i \in [n], j \in [m]} u_{i,j} < n,$$

which contradicts the fact that $\sum_{j \in [m]} u_{i,j} = \sum_{j \in [m]} u'_{i,j} = 1, \forall i \in [n]$. Hence, we have $d_v(U, U') \leq 2n - \frac{2n}{m}$.

To see that also $d_d(U, U') \leq 2n - \frac{2n}{m}$, we upper-bound $d_d(U, U') \leq d_v(U, U')$.

Assume towards a contradiction that the demand distance between two instances would be greater than the valuation distance. Let $V = \overrightarrow{\text{dem}}_U(1) \cdots \overrightarrow{\text{dem}}_U(m)$ and $V' = \overrightarrow{\text{dem}}_{U'}(\sigma(1)) \cdots \overrightarrow{\text{dem}}_{U'}(\sigma(m))$ be the matrices resulting from the column-wise concatenation of the demand vectors of U and U' , respectively, using some permutation σ of the columns.

Recall Eq. (2). Since the demand distance is upper-bounded by the entry-wise sum of ℓ^1 -distances between V and V' , the same reasoning as above holds. Thus,

$$\sum_{i \in [n], j \in [m]} \min(v_{i,j}, v'_{i,j}) < \sum_{i \in [n], j \in [m]} \min(u_{i,j}, u'_{i,j}), \tag{6}$$

Note that, Eq. 6 holds for any column permutation used to define V' , so in particular also when we use the same which we used for the valuation distance. In other words, we can assume that, up to permutation of the entries, U and V as well as U' and V' have the same column vectors.

For Eq. 6 to hold, it would need to hold that

$$\sum_{i \in [n]} \min(v_{i,j^*}, v'_{i,j^*}) < \sum_{i \in [n]} \min(u_{i,j^*}, u'_{i,j^*}) \tag{7}$$

for some column $j^* \in [m]$. Due to the rearrangement inequality (Lemma 1), however, we know that $\sum_{i \in [n]} \min(v_{i,j^*}, v'_{i,j^*}) \geq \sum_{i \in [n]} \min(u_{i,j^*}, u'_{i,j^*})$; a contradiction to Eq. 7. \square

F Additional Experimental Results

In this section, we expand upon Sections 3.2 and 4.3 by (i) presenting the remaining comparisons of the different maps for the different features, (ii) introducing new features, and (iii) disaggregating plots of the distributions of instance sources.

A comparison of the distance-embedding and the explicit map regarding the minimax envy, the maximum Nash welfare, and the maximum utilitarian welfare for both the 3×6 and 5×5 instances can be seen in Figures 9, 10, and 11, respectively; these show that the distance-embedding and explicit provide similar information regarding these features.

The same observation holds for some other features: Figure 12 shows whether an instance permits an envy-free allocation—an information that can also be derived from the minimax envy—, while Figure 13 shows whether an instance allows an envy-free and Pareto-efficient allocation. Interestingly, the maps of the two features look identical—indeed they are for 5×5 , and only differ for 6 instances for 3×6 . We have also investigated whether an instance fulfills the max-min fair share criterion, which requires each agent to get a bundle with a utility no less than the maximum, over all allocations, of the utility of the bundle that has the lowest utility for the agent. We omit the corresponding maps, as each instance of our two instance set satisfies this criterion.

Furthermore, we consider the sum, over all agents, of the maximal envies, i.e. $\sum_{i \in \mathcal{A}} \max_{i' \neq i} u_i(S_{i'}) - u_i(S_i)$, which can be seen in Figure 14 and which shows a similar color gradient to utilitarian welfare, but reversed: the sum of the absolute envies (smoothly) decreases if an instance is closer to separability.

While the previous features are based on allocations, we also consider features that can be computed solely from the utility matrix. Figure 15 and Figure 16 show the “maximum demand” and “preference diversity”, respectively, which are introduced in Section 4.1: These results support the correlation between the features and the singular values mentioned in the latter section. Figure 17 shows the fraction of agents who are single-minded, i.e., value only one item: Roughly half of the map is covered by instances in which at most around 20% of the agents are single-minded.

In addition, we introduce the following measures: In order to measure the *diversity of demand* we create a vector of all total demands and compute a Gini coefficient of that vector, where the demand for the good i is defined as $\sum_{a \in \mathcal{A}} u_a(i)$. On the other hand, we compute Gini coefficient of each vote and take the average over all votes in order to measure the *pickiness*. The maps showing these two features can be seen in Figure 18 and Figure 19 (which shows one minus pickiness, because then the value is one for one extreme point and zero for the other extreme points, which is also the case for diversity of demand and preference diversity), which show that these measures also vary smoothly over the map.

Lastly, we highlight each of the different instance sources separately in Figures 20 to 26, which shows the observations about the instances sources on the distance-embedding map in Section 3.2 more clearly on both distance-embedding and explicit maps, both of which show similar distributions.

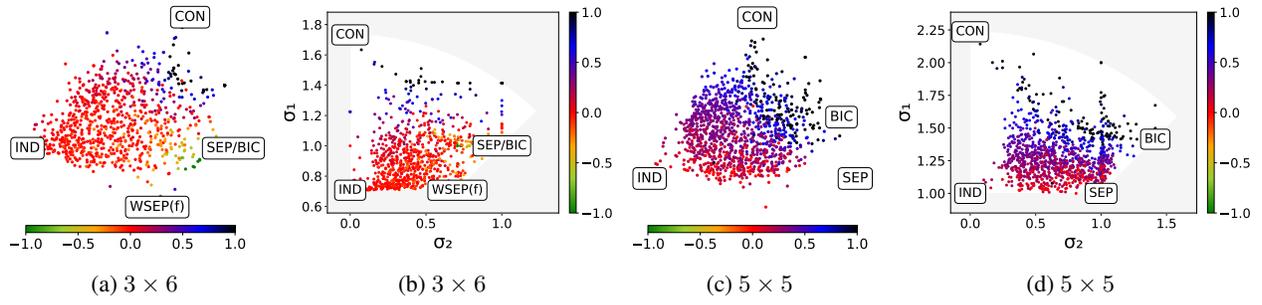


Figure 9: Distribution of the minimax envy on our distance-embedding (first and third from left) and explicit map (second and fourth from left).

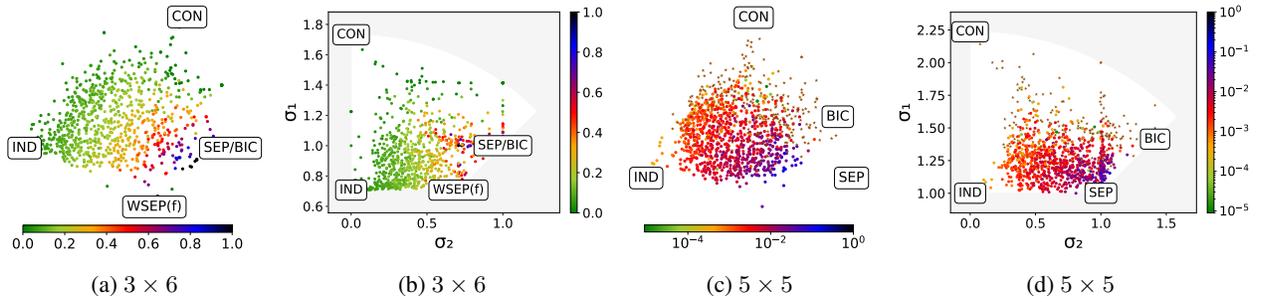


Figure 10: Distribution of the maximal Nash welfare on our distance-embedding (first and third from left) and explicit map (second and fourth from left). Brown stars represent the value 0 in the 5×5 maps.

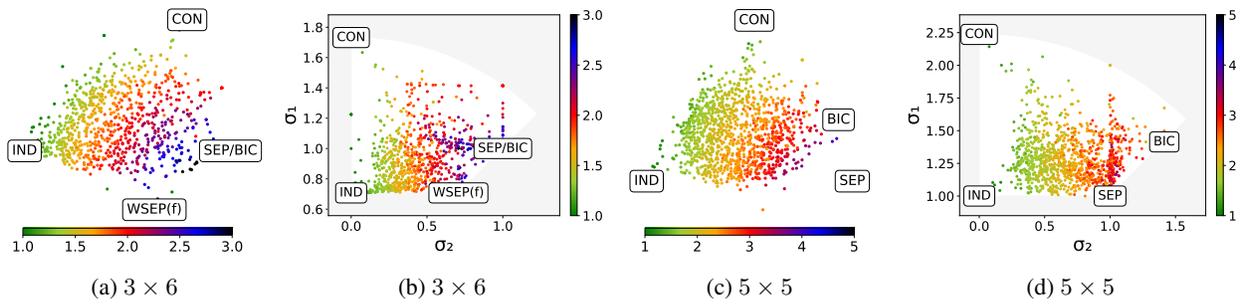


Figure 11: Distribution of the maximum utilitarian welfare on our distance-embedding (first and third from left) and explicit map (second and fourth from left).

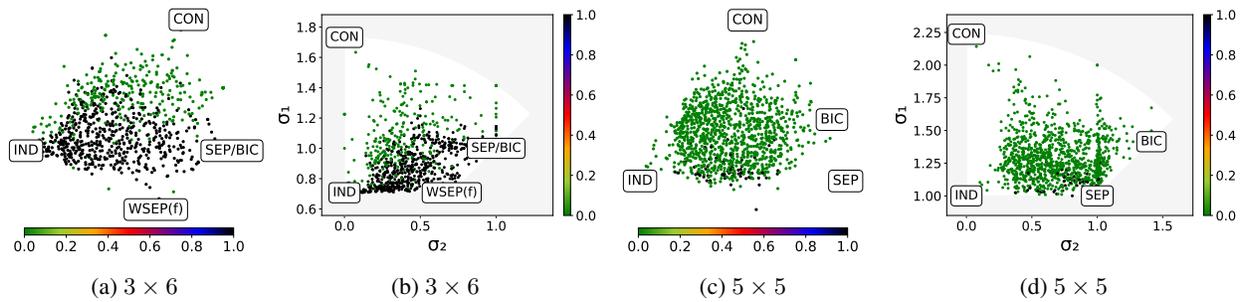


Figure 12: Distribution of whether an envy-free allocation exists on our distance-embedding (first and third from left) and explicit map (second and fourth from left).

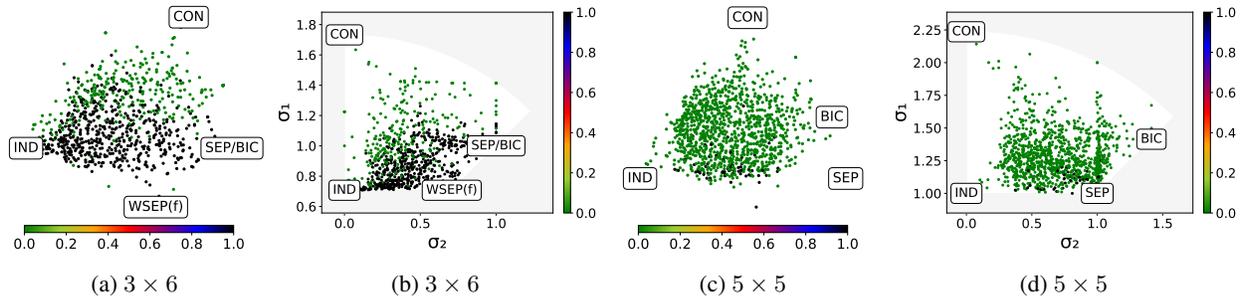


Figure 13: Distribution of whether an envy-free and Pareto efficient allocation exists on our distance-embedding (first and third from left) and explicit map (second and fourth from left).

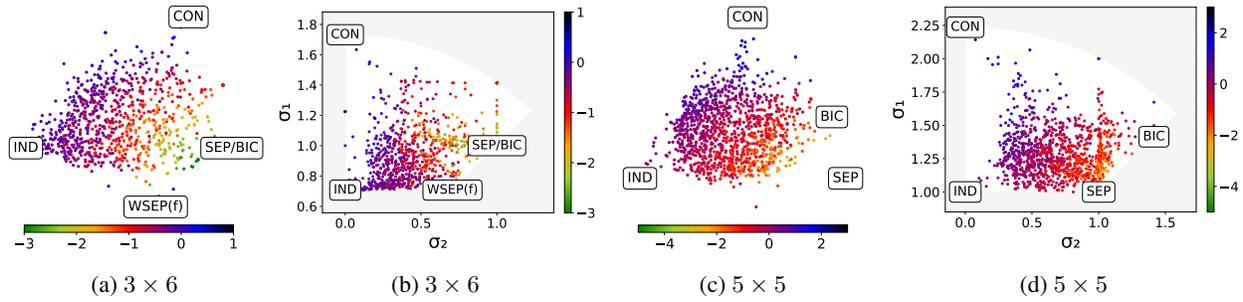


Figure 14: Distribution of the sum of the maximal envies on our distance-embedding (first and third from left) and explicit map (second and fourth from left).

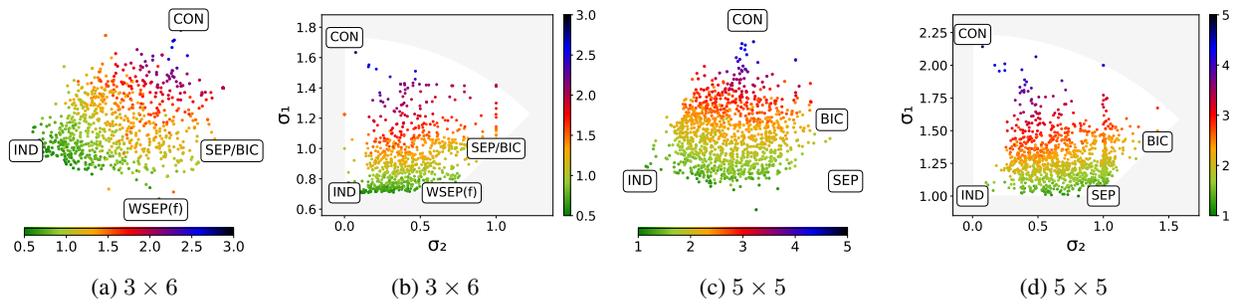


Figure 15: Distribution of the maximum demand on our distance-embedding (first and third from left) and explicit map (second and fourth from left).

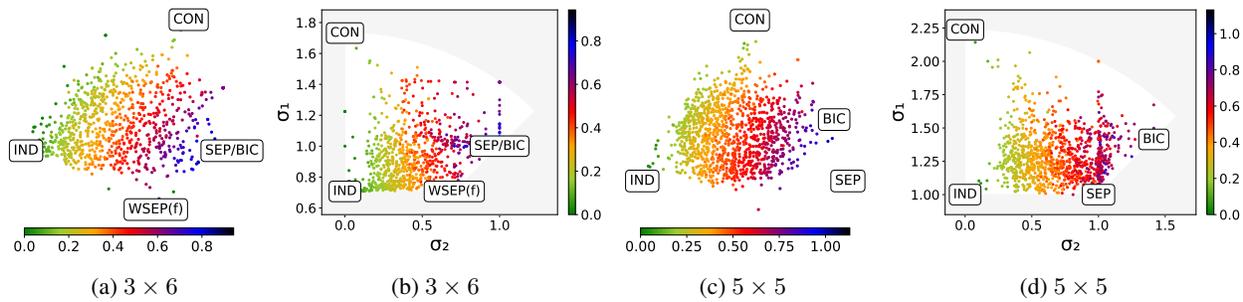


Figure 16: Distribution of the preference diversity on our distance-embedding (first and third from left) and explicit map (second and fourth from left).

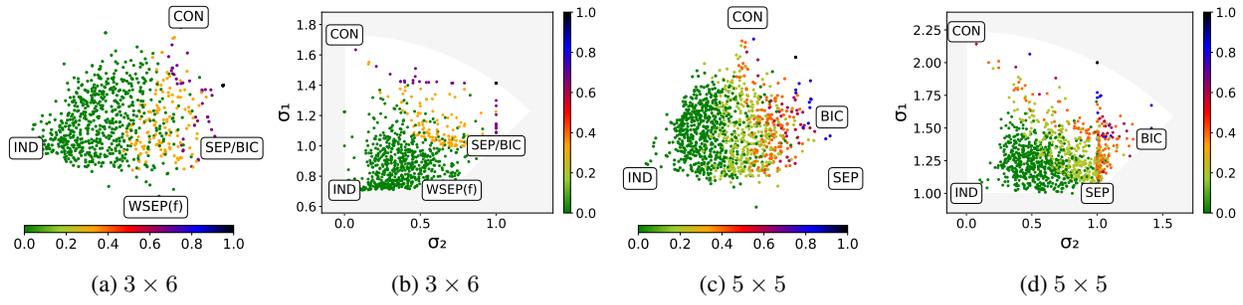


Figure 17: Distribution of the agents who are single-minded on our distance-embedding (first and third from left) and explicit map (second and fourth from left).

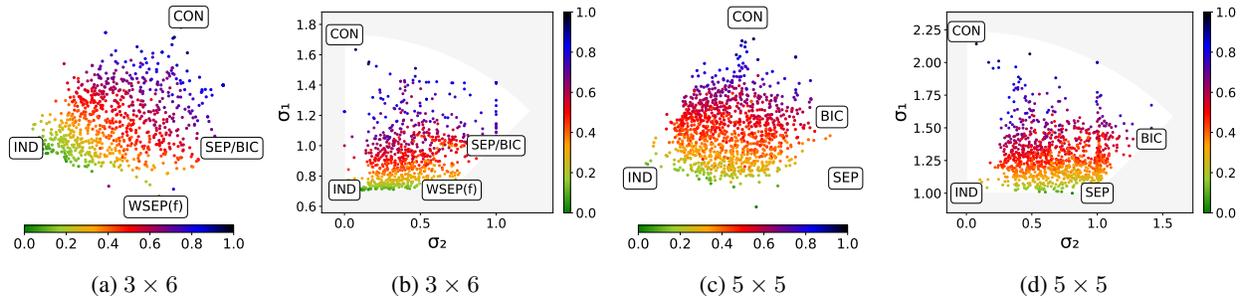


Figure 18: Distribution of the diversity of demand on our distance-embedding (first and third from left) and explicit map (second and fourth from left).

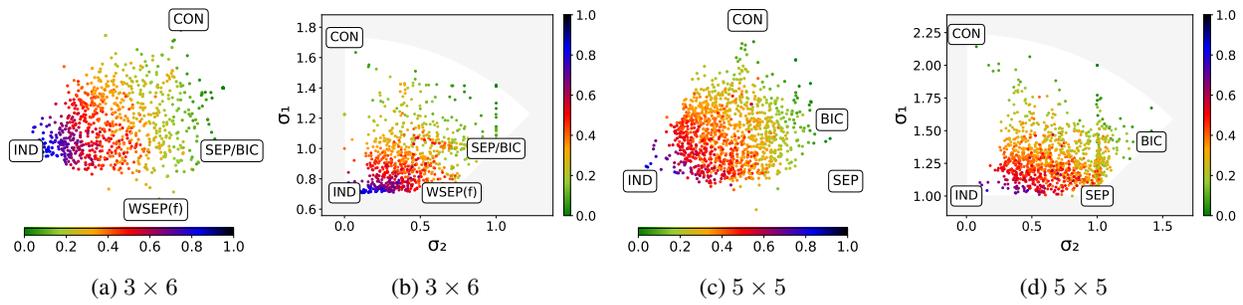


Figure 19: Distribution of one minus pickiness on our distance-embedding (first and third from left) and explicit map (second and fourth from left).

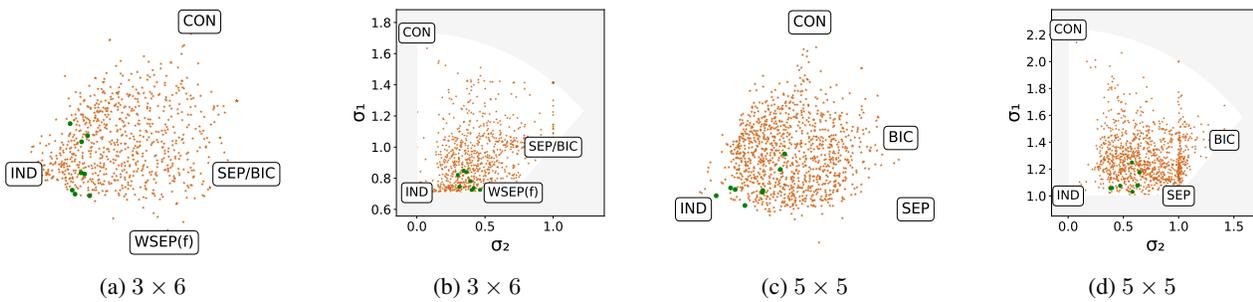


Figure 20: The instances from the euclidean distribution are marked as green dots, while all other instances are marked as brown stars.

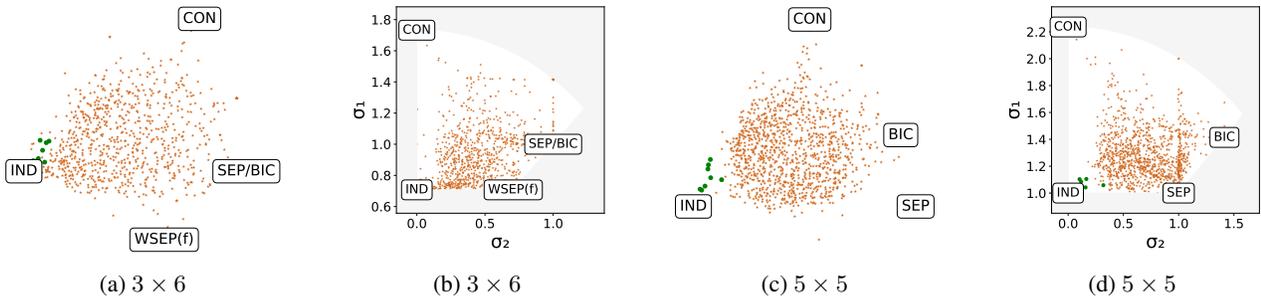


Figure 21: The instances from the attributes distribution are marked as green dots, while all other instances are marked as brown stars.

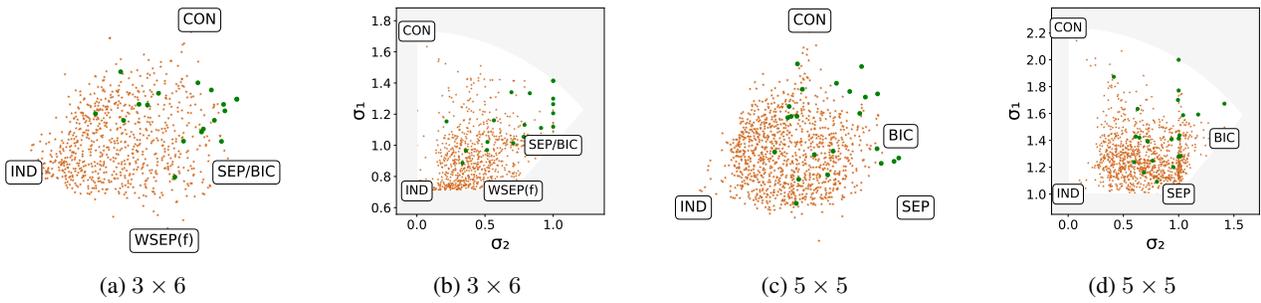


Figure 22: The instances from the Dirichlet-Resampling distribution are marked as green dots, while all other instances are marked as brown stars.

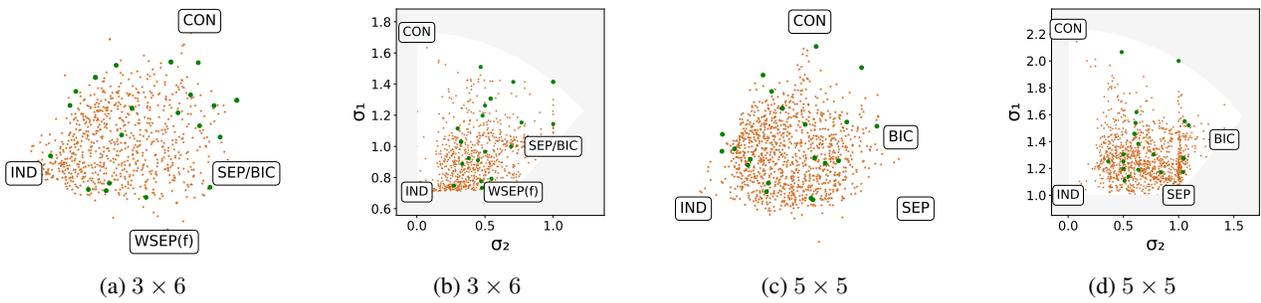


Figure 23: The instances from the resampling distribution are marked as green dots, while all other instances are marked as brown stars.

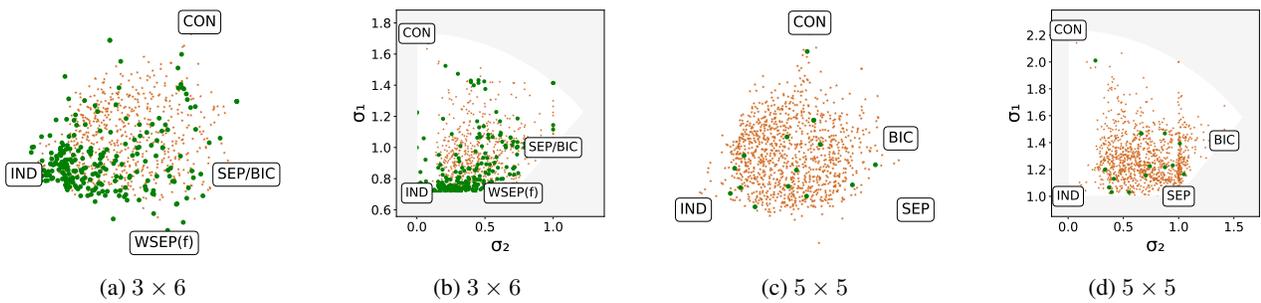


Figure 24: The instances from the Spliddit dataset are marked as green dots, while all other instances are marked as brown stars.

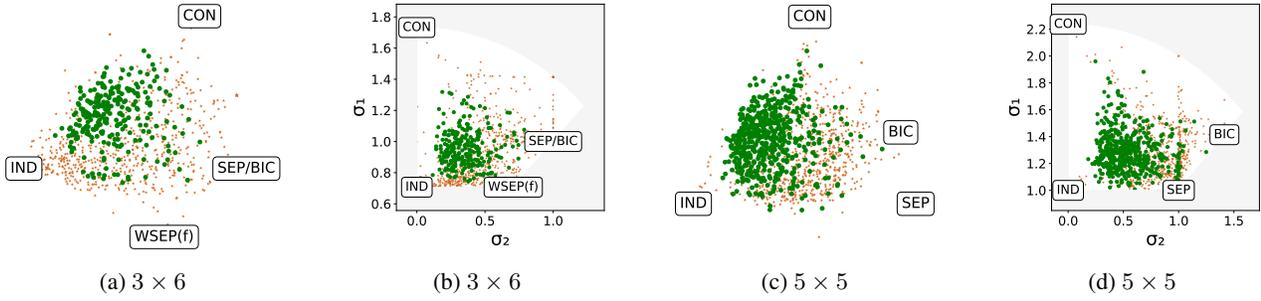


Figure 25: The instances from the island dataset are marked as green dots, while all other instances are marked as brown stars.

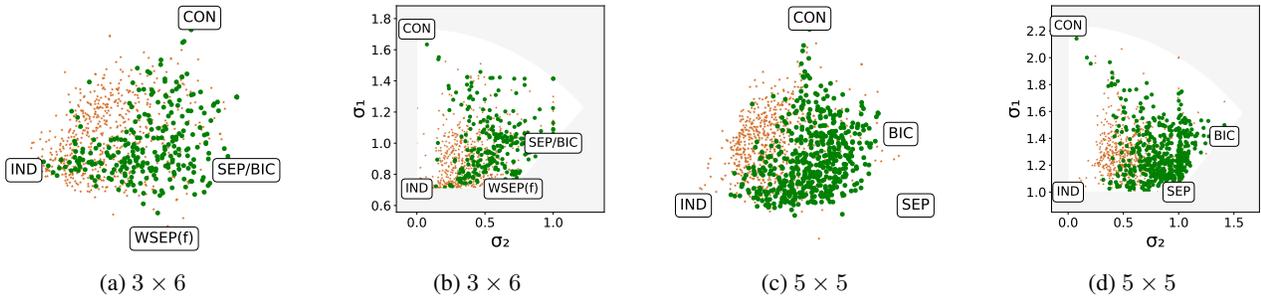


Figure 26: The instances from the candies dataset are marked as green dots, while all other instances are marked as brown stars.