On The Structure of Parametric Tournaments with Application to Ranking from Pairwise Comparisons

Vishnu Veerathu[*]
Cohesity Inc.
ofvishnuveerathu@gmail.com

Arun Rajkumar
Indian Institute of Technology, Madras
Robert Bosch Center for Data Science and AI
arunr@cse.iitm.ac.in

Abstract

We consider the classical problem of finding the minimum feedback arc set on tournaments (MFAST). The problem is NP-hard in general and we study it for important classes of tournaments that arise naturally in the problem of learning to rank from pairwise comparisons. Specifically, we consider tournaments classes that arise out of parametric preference matrices that can lead to cyclic preference relations. We investigate their structural properties via forbidden sub tournament configurations. Towards this, we introduce *Tournament Dimension* - a combinatorial parameter that characterizes the size of a forbidden configuration for rank r tournament classes i.e., classes that arise out of pairwise preference matrices which lead to rank r skew-symmetric matrices under a suitable link function. Our main result is a polynomial-time algorithm - `Rank2Rank` - that solves the MFAST problem for the rank 2 tournament class. This is achieved via a geometric characterization that relies on our explicit construction of a forbidden configuration for this class.

Building on our understanding of the rank-2 tournament class, we propose a very general and flexible parametric pairwise preference model called the local-global model which subsumes the popular Bradley-Terry-Luce/Thurstone classes to capture locally cyclic as well as globally acyclic preference relations. We develop a polynomial-time algorithm - `BlockRank2Rank`- to solve the MFAST problem on the associated Block-Rank 2 tournament class.

As an application, we study the problem of learning to rank from pairwise comparisons under the proposed local-global preference model. Exploiting our structural characterization, we propose `PairwiseBlockRank` - a pairwise ranking algorithm for this class. We show sample complexity bounds of `PairwiseBlockRank` to learn a good ranking under the proposed model. Finally, we conduct experiments on synthetic and real-world datasets to show the efficacy of the proposed algorithm.

1 Introduction

A tournament is a complete directed graph. Given a tournament $\mathbf{T}$, the classical feedback arc set on tournament (MFAST) problem asks for the minimum number of edges that must be removed (or whose orientation reversed) in $\mathbf{T}$ to make it acyclic [3]. The problem is known to be NP-hard for general tournaments [6]. We investigate the MFAST problem for several classes of tournaments which naturally occur in learning to rank from pairwise comparisons. In particular, we wish to study the MFAST problem on tournament classes which arise out of parametric pairwise preference classes.

Popular parametric preference models such as the Bradley-Terry-Luce (BTL) [4; 17] and Thurstone models [25] give rise to acyclic tournament matrices for which the MFAST problem is trivial. Our

[*]Cohesity, Inc, has neither endorsed nor sponsored this research or its publication in any way; the author's research opinions and findings are his own.

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

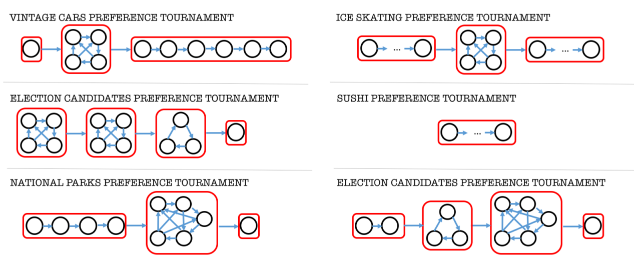

Figure 1: Several tournaments that arise out of real world datasets exhibit cyclic structure. (Source: `www.preflib.org`). Figure shows six such examples in varied application domains. Except the Sushi Preference Tournament, none of the others can be modelled using acyclic preference models like Bradley-Terry-Luce/Thurstone. The local-global model described in Section 6 and the associated Block- $\mathcal{T}_2^n$ class defined in this paper induces all these flexible tournament structures.

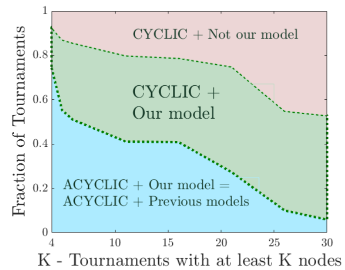

Figure 2: Out of 290 real world tournaments from `www.preflib.org`, figure shows the fraction of tournaments that are acyclic and the fraction of cyclic tournaments that satisfy the Block-Rank 2 model we propose.

goal is to identify and study non-trivial classes of parametric preference matrices which give rise to potentially cyclic tournaments. Cyclic tournaments arise naturally in several real world preference datasets. Figure 1 shows a few real world cyclic tournament structures and Figure 2 shows the fraction of tournaments out of 290 real world tournaments that are acyclic and those that are cyclic and can arise out of the models proposed in this paper. Clearly, simple models such as BTL/Thurstone are insufficient to model these tournaments.

Every pairwise preference matrix $\mathbf{P} \in (0,1)^{n \times n}$ can be associated to a tournament via a skew symmetric link function ϕ i.e, $\phi(\mathbf{P})$ is a skew symmetric matrix and the tournament $\mathbf{T}$ associated to $\mathbf{P}$ has an edge from i to j if and only if $\phi(P_{ij}) > 0$. While preference matrices themselves are always of high rank [11], several well studied pairwise preference classes give rise to skew symmetric matrices of low rank under a suitable link function. For example, the BTL model gives rise to a rank 2 skew symmetric matrix under the `logit` link whereas the Thurstone Model does so under a `probit` link [22]. It is important to understand the possible tournaments any parametric preference class can give rise to. Indeed, it might be natural and much easier for a modeller/domain expert to make a structural statement such as *The class of tournaments of interest should contain cycles of length at most* 4 as opposed to an algebraic statement like *The class of tournaments should be associated with a preference matrix which on a skew symmetric transformation leads to a rank* 8 *matrix*.

We study the structure of rank r tournament class i.e, tournaments which arise out of preference matrices that lead to some rank r matrix under a skew-symmetric transformation. This class can be interpreted as each node in the tournament having an embedding in $\mathbb{R}^r$ and the preference relation between nodes depends on a suitable notion of similarity between the embeddings (see Section 4 for details). It is known that such classes can model cyclic relations [22]. To investigate the MFAST problem for the rank r tournament classes, we first structurally understand these tournament via the notion of forbidden configurations [8]. Forbidden configurations in tournaments have been studied in other non-parametric contexts. The simplest example is the class of acyclic tournaments which has the 3 cycle as a forbidden configuration. A complicated forbidden configuration class using Möbius ladders was studied in [8; 9]. In both these cases, the MFAST problem is poly-time solvable.

Our Contributions. Our first contribution is to derive upper and lower bounds for the size of a forbidden configuration for any rank r tournament class. We do this by introducing a novel combinatorial parameter that we call the *Tournament Dimension*. When rank $r = 2$, we completely characterize the associated tournament class by identifying the exact forbidden configuration. We use this forbidden configuration to understand the geometry of this tournament class and propose an algorithm (`Rank2Rank`) to solve the MFAST problem in polynomial time for this class of tournaments.

Our next contribution is to propose a flexible class of tournaments called the Block Rank 2 class which builds on the rank 2 class. Tournaments in this class can arise from the *local-global* model - a general pairwise preference model that we propose. The model can capture locally cyclic relations as well as globally acyclic relations and is well suited for several practical applications including those in Figure 1. The model subsumes several popular pairwise ranking models including the BTL and

Thurstone models. We show that the MFAST problem on the associated Block Rank 2 tournament class is polynomial time solvable and propose an algorithm (`BlockRank2Rank`) for the problem.

Our third contribution is to study an application of our work to the problem of learning to rank from pairwise comparisons. We propose a matrix completion based algorithm called `PairwiseBlockRank` for this problem and derive sample complexity bounds for the same under the proposed local-global preference model. Finally, we evaluate our algorithms on real world and synthetic data-sets and show improved performance over several existing algorithms.

2 Related Work

Several works have considered learning from transitive pairwise models especially focusing on the Bradley-Terry-Luce (BTL) model[12; 20; 21; 24; 14]. As we focus on cyclic relations, we discuss only the models which can lead to cyclic preferences. While the BTL model can be seen as using a 1 dimensional embedding of the item using a score vector, studies have considered higher dimensional embeddings. A 2 dimensional embedding was considered in [5]. While their model can give rise to cyclic tournaments, it is not clear what type of cycles are possible. [18; 19] propose the Majority vote model which is a random utility model (RUM) with a d dimensional feature embedding for each item. The Majority vote model is powerful enough to produce arbitrarily long cycles and can express any probability sub-matrix over a fixed triplet. The Rank 2 model we study is as powerful as the Majority vote model. Furthermore, we give a complete structural characterization of the tournaments of our model which is not known for the Majority vote model. The Blade-Chest inner (BCI) model [7] embeds each item into two d dimensional vectors (blade vector and a chest vector) and a score vector $\mathbf{s}$ where the probability of i being preferred over j depends on $< i_{\text{chest}}, j_{\text{blade}} > - < i_{\text{blade}}, j_{\text{chest}} > + s_i - s_j$. The rank-2 model can be seen as a special case of the BCI model where $d = 1$ and $\mathbf{s}$ is a constant vector. When d is $\mathcal{O}(n)$, the BCI model can give rise to any tournament as there are $\mathcal{O}(n^2)$ parameters and so the MFAST problem becomes intractable. We focus on the special case of $d = 1$ where there are only $2n$ parameters and propose algorithms to obtain optimal rankings. Finally, we discuss the low rank pairwise rank (LRPR) class of models which result in a low rank matrix under a transformation using a suitable link function [22]. Previous work [12,22] have proposed matrix completion based algorithms to obtain optimal ranking for the LRPR type models assuming transitivity of preferences. In this work, we make no such assumptions.

3 Preliminaries

A Tournament is a complete directed graph. We use $i \succ_{\mathbf{T}} j$ to denote that there is a directed edge from node i to j in the tournament $\mathbf{T}$. We call a set of edges F_σ in a tournament $\mathbf{T}$ as the feedback arc set of permutation σ w.r.t $\mathbf{T}$ if $\sigma(i) > \sigma(j)$ (j ranked ahead of i in σ) but $i \succ_{\mathbf{T}} j$. Indeed, if one reverses the orientation of the F_σ edges in $\mathbf{T}$, one gets an acyclic tournament whose topological sort would yield σ. We call a permutation σ^* as *MFAST-Optimal* for $\mathbf{T}$ if $\sigma^* \in \arg\min_\sigma |F_\sigma|$.

We call $\mathbf{P} \in (0,1)^{n \times n}$ a pairwise preference matrix if $P_{ij} + P_{ji} = 1 \; \forall i \neq j$. We assume $P_{ij} \neq 0.5 \; \forall i, j$. We will say $\mathbf{T}_{\mathbf{P}}$ is a tournament associated with $\mathbf{P}$ if a directed edge from node i to j is present in $\mathbf{T}_{\mathbf{P}}$ if and only if $P_{ij} > 0.5$. We call $\mathbf{P}$ a *rank r preference matrix w.r.t a link function ϕ* if rank($\phi(\mathbf{P})$) = r where the function ϕ is applied element-wise to $\mathbf{P}$. We will call ϕ a skew symmetric link function if $\phi(\mathbf{P})$ is a skew symmetric matrix for any pairwise preference matrix $\mathbf{P}$. Examples of such ϕ include the `logit` and `probit` links where $\texttt{logit}(x) = \log(x/(1-x))$ and $\texttt{probit}(x) = \Phi^{-1}(p)$ where Φ is the standard normal cdf. We use $\mathcal{T}_r^{n,\phi}$ to denote the class of all tournaments on n nodes which are associated with a rank r preference matrix w.r.t ϕ.

Our goal is the study the tournament class $\mathcal{T}_r^{n,\phi}$ where ϕ is a skew symmetric link function. Skew-symmetric matrices are naturally associated to tournaments where the edge directions in the tournament are determined by the sign of the corresponding matrix entry. Our results would apply for any skew symmetric ϕ and so we will drop the ϕ in $\mathcal{T}_r^{n,\phi}$ when it is clear from the context. As skew symmetric matrices have even rank, $\mathcal{T}_r^{n,\phi}$ is non-empty only for even r. We wish to investigate the structural constraints on these tournament classes that arise out of the algebraic (rank) restriction that define them. We note that it was known earlier that for the special case when $r = 2$, $\mathcal{T}_2^{n,\phi}$ contains both cyclic and acyclic tournaments [22]. However, nothing further was known about this class.

Algorithm 1 `Tournament Game`

Input: Integers k, d
Player-1 chooses a labelled dataset $\{(\mathbf{x}_1, y_1), \dots (\mathbf{x}_k, y_k)\}$ where $\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^k \in \mathbb{R}^d$ induces a Tournament, $y_i \in \{+1, -1\}$
Player-2 chooses a Tournament preserving mapping f w.r.t $\mathcal{D}$
if $\exists \mathbf{w} \in \mathbb{R}^d$ such that $\texttt{sign}(\mathbf{w}^T f(\mathbf{x}_i)) = y_i \;\; \forall i \in [k]$ **then**
  Player-2 wins the game
else
  Player-1 wins the game
end if

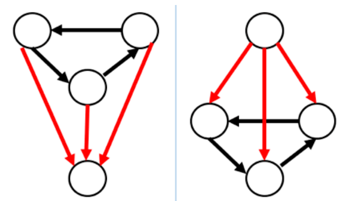

Figure 3: Forbidden configurations for any induced tournament on 4 nodes in $\mathcal{T}_2^n$

Definition 1. *A class of tournaments $\mathcal{T}$ is said to have a forbidden configuration of size ℓ if ℓ is the smallest integer such that there exists a tournament $\mathbf{T}^\ell$ on ℓ nodes which does not appear as an induced sub-tournament for any $\mathbf{T} \in \mathcal{T}$*

In other words, no induced sub-tournament on l nodes of $\mathbf{T} \in \mathcal{T}$ is isomorphic to $\mathbf{T}^\ell$. For example, the class of all acyclic tournaments $\mathcal{T}^{\texttt{acyclic}}$ has C^3 - the 3-cycle as a forbidden configuration. Taking advantage of this forbidden configuration, one can immediately come up with an algorithm to solve the MFAST problem on $\mathcal{T}^{\texttt{acyclic}}$ - indeed, a topological sort solves this problem. Our interest is to understand if there are similar forbidden configurations for the class $\mathcal{T}_r^{n,\phi}$. Towards this, we start the next section by defining a novel combinatorial parameter called the Tournament dimension.

4 Tournament Dimension

Let d be an even integer. We say a set of points $\mathcal{D} = \{\mathbf{x}_1, \dots, \mathbf{x}_k\} \in \mathbb{R}^d$ *induces a tournament* $\mathbf{T}$ if $\mathbf{x}_i$ is a d-dimensional embedding of node i and a directed edge from node i to j exists in $\mathbf{T}$ if and only if $\mathbf{x}_i^T A^{\texttt{rot}} \mathbf{x}_j > 0$. Here, $A^{\texttt{rot}} \in \mathbb{R}^{d \times d}$ is the block diagonal matrix with $\frac{d}{2}$ blocks where each block of size 2 is a rotation matrix $[0\ -1; 1\ 0]$ [2]. Note that $\mathbf{x}_i^T A^{\texttt{rot}} \mathbf{x}_j = -\mathbf{x}_j^T A^{\texttt{rot}} \mathbf{x}_i \; \forall i, j$ and the tournament inducing property of $\mathcal{D}$ excludes datapoints such that $\mathbf{x}_i^T A^{\texttt{rot}} \mathbf{x}_j = 0$. We note that d being even is not a restriction on the embedding. If d is odd, one can increase the embedding to $d + 1$ dimensions w.l.o.g where the last component is 1 for all $\mathbf{x}_i$.

Lemma 1. *Let $\mathcal{D} = \{\mathbf{x}_1, \dots, \mathbf{x}_k\} \in \mathbb{R}^d$ induce a tournament $\mathbf{T}$. Then $\mathbf{T} \in \mathcal{T}_d^k$. Furthermore, for every $\mathbf{T} \in \mathcal{T}_d^k$ there exists a dataset $\mathcal{D}$ with k vectors in $\mathbb{R}^d$ that induces $\mathbf{T}$.*

We call a mapping $f : \mathbb{R}^d \to \mathbb{R}^d$ as *Tournament-preserving* w.r.t. a tournament inducing dataset $\mathcal{D}$ if $(f(\mathbf{x}_i)^T A^{\texttt{rot}} f(\mathbf{x}_j) > 0) \iff (\mathbf{x}_i^T A^{\texttt{rot}} \mathbf{x}_j > 0) \;\; \forall \mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}$. We define the *Tournament game* between two players as described in Algorithm 1.

Definition 2. *The Tournament dimension for d - $\texttt{TourDim}(d)$ - is the largest value of k for which Player 2 always has a winning strategy in the Tournament game (Algorithm 1).*

Remark. $\texttt{TourDim}(d) = k$ implies that Player 1 has a winning strategy for all values $\geq k + 1$. In particular, there is a tournament $\mathbf{T} \in \mathcal{T}_d^k$ induced by the dataset $\mathcal{D}$ and a labelling $\{y_1, \dots, y_k\}$ strategically chosen by Player 1 such that there does not exist a $\mathbf{T}$-preserving mapping for which Player 2 can find a $\mathbf{w} \in \mathbb{R}^d$ that labels all points in $\mathcal{D}$ correctly. The inability to find such a $\mathbf{w}$ is equivalent to the inability to add one more node to $\mathbf{T}$ whose edge direction to node i of $\mathbf{T}$ is specified by the label y_i. Thus, there must be a forbidden configuration on $k + 2$ nodes.

The Theorem below formalizes the above remark.

Theorem 2. *$\mathcal{T}_r^n$ has a forbidden configuration of size $\texttt{TourDim}(r) + 2$ for every even integer $r < n$ and every $n \geq \texttt{TourDim}(r) + 2$.*

[2]The matrix $A^{\texttt{rot}}$ is fundamental to skew-symmetric matrices. In fact, any skew-symmetric bilinear form can be represented under a suitable basis using $A^{\texttt{rot}}$ and additional zero diagonal blocks as necessary [23].

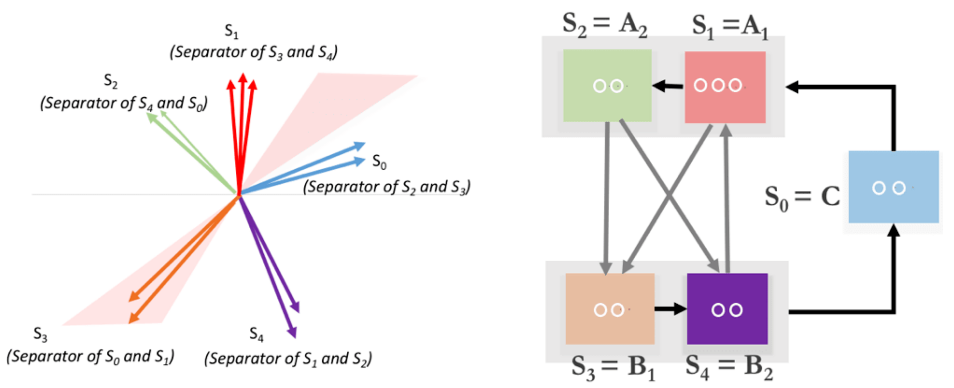

Figure 4: (Left) Geometry of the embeddings of nodes in $\mathcal{T}_2^n$ where $-S_3 \subset \texttt{cone}(S_0 \cup S_1)$ is highlighted. (Right) Structure of Tournament belonging to $\mathcal{T}_2^n$ for the example on the left. The sets $\{S_0, \ldots, S_4\}$ are colored as per the left figure and mapped to $\{A_1, A_2, B_1, B_2, C\}$ as in Theorem 6.

We bound the size of the forbidden configuration as follows

Theorem 3. *Let $\mathcal{T}_r^n$ have a forbidden configuration of size ℓ. Then $r + 2 \leq \ell \leq 2^r + r + 1$*

Remark. *We believe that the upper bound in Theorem 3 is not tight in general and leave it as an open problem to improve this for a general r.*

Corollary 1. *$\mathcal{T}_2^n$ has a forbidden configuration of size 4.*

While the above Corollary gives the size of the forbidden sub configuration, it does not specify what exactly the configuration is. As the rank 2 class occurs naturally in several learning to rank problems, we investigate this further in the next section.

5 Rank 2 Tournaments - $\mathcal{T}_2^n$

In this section we focus on the Rank 2 tournament class $\mathcal{T}_2^n$. To motivate this class, consider the following generalisation of the popular Bradley-Terry-Luce (BTL) preference model where for a skew symmetric link function ϕ, we define $\phi(P_{ij}) = u_i v_j - v_i u_j$. The model is parameterized by two vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$. When one of these vectors is the all ones vectors and when the link function ϕ is the `logit` function, the model reduces to the standard BTL model.

In general for any $\mathbf{P}$, if ϕ is a skew symmetric link function that results in a rank 2 matrix, then $\phi(\mathbf{P}) = \mathbf{u}\mathbf{v}^T - \mathbf{v}\mathbf{u}^T$ for some $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$. Such a model can be interpreted as each node i having a two dimensional embedding $\mathbf{h}_i = [u_i, v_i]$. Here, the direction of the edge between nodes i and j in the associated tournament depends on the sign of $u_i v_j - v_i u_j = \mathbf{h}_i^T A^{\texttt{rot}} \mathbf{h}_j$ where $A^{\texttt{rot}} \in \mathbb{R}^{2\times 2}$ is the rotation matrix $[0 - 1; 1\, 0]$. With this background, we now proceed to characterize the class $\mathcal{T}_2^n$.

Geometric Characterization of $\mathcal{T}_2^n$: We start with the following key lemma:

Lemma 4. *(**Forbidden configurations**) Let $n \geq 4$. Every induced tournament of $\mathcal{T}_2^n$ on a subset of four nodes forbids the configurations in Figure 3.*

The above lemma can be used to geometrically characterize the two dimensional embeddings of the nodes in $\mathcal{T}_2^n$. For this characterization, we need the following definition.

We call a set of vectors $S \subset \mathbb{R}^2$ a *separator* of two sets of vectors $U, V \subset \mathbb{R}^2$ if the convex cone of $U \cup V$ contains $-S$ i.e., the set of vectors in S rotated by 180 degrees is a subset of $\texttt{cone}(U \cup V)$

Theorem 5. *Let $\mathbf{T} \in \mathcal{T}_2^n$ where each node i has a two dimensional embedding $\mathbf{h}_i \in \mathbb{R}^2$. Then either $\mathbf{T} \in \mathcal{T}^{\texttt{acyclic}}$ or $\exists k \geq 1$ such that the vectors $\mathbf{h}_i$ can be partitioned into $2k+1$ ordered sets $\{S_0, \ldots, S_{2k}\}$ where for each $i = 0, \ldots, 2k$, $S_i \subset \mathbb{R}^2$ is a separator of the sets $S_{i+k \ mod(2k+1)}$ and $S_{i+k+1 \ mod(2k+1)}$*

Figure 4 (left) shows how the embeddings of nodes that result in $\mathcal{T}_2^n$ would look like geometrically. As one can observe, the example tournament has 5 sets of vectors and every set of vectors of the same color acts as a separator for two other sets of different colors. This is the crucial geometric insight that we obtain which characterizes this tournament class. We note that when the number of sets (colors) is just 1, the model leads to acyclic tournaments. Indeed, tournaments arising out of the BTL/Thurstone models have this property and lead to $\mathbf{T} \in \mathcal{T}^{\text{acyclic}} \subsetneq \mathcal{T}_2^n$. However, $\mathcal{T}_2^n$ is much more richer than the simple $\mathcal{T}^{\text{acyclic}}$ class.

Structural Characterization of $\mathcal{T}_2^n$: With the help of the geometric characterization, we next give the structural description of all tournaments in $\mathcal{T}_2^n$. For two sets of nodes A and B, We use $A \succ_{\mathbf{T}} B$ to denote that every node in A has an outgoing edge to every node in B in the tournament $\mathbf{T}$. Also, $A \prec_{\mathbf{T}} B$ whenever $B \succ_{\mathbf{T}} A$

Theorem 6. *Let $n \geq 4$ and $\mathbf{T} \in \mathcal{T}_2^n$. Then either $\mathbf{T} \in \mathcal{T}^{acyclic}$ or there exists $k \geq 1$ such that the n nodes can be partitioned into $2k+1$ sets $\{A_1, A_2, \ldots, A_k, B_1, B_2, \ldots, B_k, C\}$ where*
(a) The induced sub-tournament on each of the $2k+1$ sets belong to $\mathcal{T}^{acyclic}$
(b) $A_i \succ_{\mathbf{T}} A_j, B_i \succ_{\mathbf{T}} B_j \quad \forall i < j$
(c) $A_i \succ_{\mathbf{T}} B_j \; \forall i \geq j, A_i \prec_{\mathbf{T}} B_j \; \forall i < j$
(d) $B_i \succ_{\mathbf{T}} C \succ_{\mathbf{T}} A_j \quad \forall i, j$

Figure 4 (right) gives a graphical representation of the structure of $\mathcal{T}_2^n$ described in Theorem 6. The following theorem sheds light on the rich expressivity of the $\mathcal{T}_2^n$ tournament class.

Theorem 7. ***(Long Cycles and Arbitrary Triplets)*** *Given any arbitrary ordered set of indices $\{i_1, i_2, \ldots, i_k\}, k \leq n$, there exists a $\mathbf{T} \in \mathcal{T}_2^n$ such that $i_1 \succ_{\mathbf{T}} i_2 \succ_{\mathbf{T}} \ldots \succ_{\mathbf{T}} i_k \succ_{\mathbf{T}} i_1$. Furthermore, Given any $p_1, p_2, p_3 \in (0,1)$ and indices $i_1, i_2, i_3 \in [n]$, there exists a pairwise preference matrix $\mathbf{P}$ whose associated $\mathbf{T}_{\mathbf{P}} \in \mathcal{T}_2^n$ where $\mathbf{P}(i_1, i_2) = p_1, \mathbf{P}(i_2, i_3) = p_2, \mathbf{P}(i_3, i_1) = p_3$.*

5.1 Polynomial Time Algorithm for MFAST for $\mathcal{T}_2^n$

We present a polynomial time algorithm for MFAST for the $\mathcal{T}_2^n$ class. The algorithm `Rank2Rank` (Algorithm 2) takes as input a tournament $\mathbf{T}$ and outputs a permutation σ^* that is MFAST-optimal for $\mathbf{T}$ whenever $\mathbf{T} \in \mathcal{T}_2^n$. The algorithm finds such a σ^* by relying crucially on the structural understanding gained in the previous section via forbidden configurations. In the following, given a set of nodes A and a tournament $\mathbf{T}$, we represent the induced tournament on A by $\mathbf{T}$ as $\mathbf{T}_A$.

Algorithm 2 `Rank2Rank-R2R`

Input: A tournament $\mathbf{T}$
if $\mathbf{T} \in \mathcal{T}^{\text{acyclic}}$ **then**
  Obtain $\sigma*$ by a Topological sort of $\mathbf{T}$
else
  Find $\{a, b, c\}$ s.t $(a \succ_{\mathbf{T}} b \succ_{\mathbf{T}} c \succ_{\mathbf{T}} a)$
  $A := \{a\} \cup \{i : i \succ_{\mathbf{T}} b \text{ and } c \succ_{\mathbf{T}} i\}$
  $B := \{b\} \cup \{i : i \succ_{\mathbf{T}} c \text{ and } a \succ_{\mathbf{T}} i\}$
  $C := \{c\} \cup \{i : i \succ_{\mathbf{T}} a \text{ and } b \succ_{\mathbf{T}} i\}$
  $\sigma_0 = [\texttt{R2R}(\mathbf{T}_A), \texttt{R2R}(\mathbf{T}_B), \texttt{R2R}(\mathbf{T}_C)]$
  Let σ_h be the permutation obtained by a cyclic shift of σ_0 by h positions.
  Let σ^* be the permutation among $\sigma_h \; \forall h$ which has the least size of the feedback arc set w.r.t $\mathbf{T}$ i.e., $\sigma^* = \arg\min_h |F_{\sigma_h}|$
end if
Output σ^*

Algorithm 3 `BlockRank2Rank - BR2R`

Input: A tournament $\mathbf{T}$
if $\mathbf{T} \in \mathcal{T}^{\text{acyclic}}$ **then**
  Obtain σ^* by a Topological sort of $\mathbf{T}$
else
  Find $\{a, b, c\}$ s.t $(a \succ_{\mathbf{T}} b \succ_{\mathbf{T}} c \succ_{\mathbf{T}} a)$
  $S^+ = \{i : i \succ_{\mathbf{T}} \{a, b, c\}\}$
  $S^- = \{i : i \prec_{\mathbf{T}} \{a, b, c\}\}$
  $S = [n] \backslash \{S^+ \cup S^-\}$
  $\sigma^* = [\texttt{BR2R}(\mathbf{T}_{S^+}), \texttt{R2R}(\mathbf{T}_S), \texttt{BR2R}(\mathbf{T}_{S^-)}]$
end if
Output σ^*

Theorem 8. *If $\mathbf{T} \in \mathcal{T}_2^n$ is given as input to the `Rank2Rank` (Algorithm 2), the output σ^* produced by the algorithm is MFAST-optimal for $\mathbf{T}$. Furthermore, `Rank2Rank` has a time complexity of poly(n).*

Geometric Sweep Interpretation: One can also come up with a geometric sweep algorithm to solve the MFAST problem for $\mathcal{T}_2^n$ inspired by the geometric embedding based interpretation given earlier. Here, we assume that the embeddings of the items are given as input. The algorithm begins by fixing an arbitrary anchor item i and circular sweeps the $\mathbb{R}^2$ plane in the counter clockwise direction from the embedding of i thus creating a permutation σ_0 of items corresponding to the order in which they are encountered in the sweep. Starting to sweep from different anchors would give rise to n different cyclic shifts of σ_0. We argue in the appendix that the optimal σ^* that solves the MFAST problem w.r.t $\mathbf{T}$ must necessarily be one of these n permutations. We note that while the Geometric sweep algorithm needs the embeddings to be known, the `Rank2Rank` algorithm does not need the same.

6 Block $\mathcal{T}_2^n$ Class

Often in practical applications, items can be clustered into blocks such that within block they exhibit a certain *local* pairwise preference whereas across blocks they exhibit a *global* pairwise preference. For example, consider the game of tennis where the top 3 players might have a cyclic preference among themselves whereas they strictly beat every other player in the bottom $n-3$. To capture such effects, we next propose the *local-global* pairwise preference model.

6.1 Local-Global Preference Model

The local-global pairwise preference model is parameterized by three vectors $\mathbf{s}, \mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ and a ordered partition of $[n]$ given by $\{H_1, ..., H_b\} \in 2^{[n]}$ where $\cup_{k=1}^b H_k = [n]$ and $H_i \cap H_j = \emptyset \; \forall i \neq j$. Let $g(i)$ denote the partition in which item i is present. We assume that $s_i > s_j \iff g(i) < g(j)$ i.e., the parameter vector $\mathbf{s}$ aligns with the ordering of the partitions. The pairwise preference probability of item i being preferred over item j under a skew symmetric link function ϕ is given as

$$\phi(P_{ij}) \;= \begin{cases} u_i v_j - v_i u_j \text{ if } \exists \ell \text{ s.t } i,j \in H_\ell \\ s_i - s_j \text{ otherwise} \end{cases}$$

Figure 5 shows some flexible tournament structures that arise out of the local global model and Figure 1 shows several real world tournaments that have associated tournaments arising out of this model.

One can interpret the above model as follows: The set of $[n]$ items is divided into b blocks/partitions. Restricted to each block, the *local* pairwise preference matrix has an associated tournament that belongs to $\mathcal{T}_2^n$ which can potentially contain cycles. However, across partitions, the preference matrix behaves like a standard BTL type model parameterized by the score vector $\mathbf{s}$ and hence the associated *global* block level tournament relation belongs to $\mathcal{T}^{\texttt{acyclic}}$. The tournament associated to the entire set of n nodes thus has a *block* structure. Indeed, the model is determined by $3n$ parameters where $2n$ parameters determine the intra block local structure and n parameters the inter block global structure.

Remark. *The standard BTL model and it's generalization with $2n$ parameters are special cases of the above model where the number of blocks $b = 1$ and ϕ is the `logit` link.*

We refer to the class of tournaments that arise out of a local-global model as the $\texttt{Block} - \mathcal{T}_2^n$ class.

Theorem 9. *If $\mathbf{T} \in$ `Block`-$\mathcal{T}_2^n$ is given as input to the `BlockRank2Rank` (Algorithm 3), the output σ^* produced by the algorithm if MFAST-optimal for $\mathbf{T}$. Furthermore, `BlockRank2Rank` has a time complexity of poly(n).*

7 Learning to Rank From Pairwise Comparisons under `Block`-$\mathcal{T}_2^n$

We now consider the application of the models discussed so far to the problem of learning to rank from pairwise comparisons under the local-global pairwise preference model which lead to tournaments in `Block`-$\mathcal{T}_2^n$. Towards this, we propose the `PairwiseBlockRank` algorithm (Algorithm 4). Given a dataset of pairwise comparisons, the algorithm constructs the empirical pairwise comparison matrix. It applies the link function ϕ to construct a empirical skew-symmetric matrix with missing entries (corresponding to pairs that are not compared in the dataset). The algorithm then applies a matrix completion routine to complete the skew-symmetric matrix. Once complete, the associated tournament is computed and the ranking obtained using the `BlockRank2Rank` algorithm.

Matrix completion routines (such as [16]) typically require an upper bound on the rank of the matrix being completed. The following Lemma establishes that this depends on b, the number of blocks.

Lemma 10. *Let $\mathbf{T} \in$ `Block`-$\mathcal{T}_2^n$ with b blocks. Then $\mathbf{T} \in \mathcal{T}_r^n$ for some $r \leq 4b$.*

The following theorem establishes sample complexity of learning for recovering the blocks correctly.

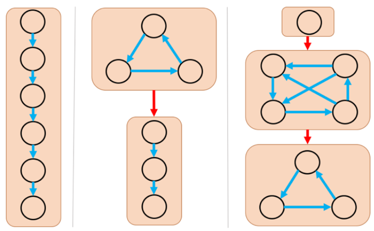

Figure 5: Example of a few Flexible Tournament Structures arising out of the *local-global* model with $1, 2$ and 3 blocks.

Algorithm 4 `PairwiseBlockRank`

```
Input: D = {(i_k, j_k), y_k}, k = {1, ..., m}, link function φ, rank r
Compute Empirical pairwise preference matrix:
if (i, j) = (i_k, j_k) for some k then
    P̂_ij = Σ_{k=1}^m 𝕀(i_k=i, j_k=j, y_k=1) / 𝕀(i_k=i, j_k=j); P̂_ji = 1 − P_ij
else
    P̂_ij = P̂_ji = 0
end if
M̂ = Matrix-Completion(φ(P̂), r)
Construct Tournament T from M̂ where (i, j) ∈ E whenever M̂_ij > 0
σ* = BlockRank2Rank - BR2R(T)
Output σ*
```

Theorem 11. *(**Block Recovery**) Let the local global model be parameterized by vectors* $\mathbf{u}, \mathbf{v}, \mathbf{s} \in \mathbb{R}^n$ *and ordered partitions* $\{H_1, \ldots, H_b\}$ *and a skew symmetric link function* ϕ*. Let* m *pairs be chosen uniformly at random from all subsets of* $\binom{n}{2}$ *of size* m *and let each pair be compared* K *times according to the corresponding preference matrix* $\mathbf{P}$*. Let* $\theta = \min_{ij} |u_i v_j - v_i u_j|$ *where* (i, j) *are in the same partition and let* $\Delta = \min |s_i - s_j|$ *where* (i, j) *are in different partitions. Let* $0 < \epsilon < n^2(\frac{\theta}{\Delta})^2$*. Then with probability at least* $1 - \frac{2}{n^3}$*, if* m *is* $O(nblog(n)))$ *and* K *is* $O(\frac{b \log(n)}{\epsilon \Delta^2})$*, the* `PairwiseBlockRank` *algorithm on running with the dataset generated as above, returns a ranking* σ *that respects the block structure i.e.,* $\sigma(i) < \sigma(j)$ *whenever* $i \in H_k$ *and* $j \in H_\ell \; \forall \ell > k$*.*

8 Experiments

Datasets: We use the following real world datasets (with n items to be ranked and m pairwise comparisons): DoTA [1] ($n = 757, m = 10442$), Tennis [2] ($n = 742, m = 23806$), Sushi-A [15] ($n = 10, m = 100000$), Sushi-B [15] ($n = 100, m = 25000$), Jester [13] ($n = 100, m = 891404$).

Setting + Performance measure: Given a set of pairwise comparisons, we do a $70 : 30$ train:test split. We run the algorithms on the train data to obtain a ranking σ. We test the performance of σ on the test set by computing the ratio of *upsets*(the pairs (i, j) for which $\sigma(j) < \sigma(i)$ but the fraction of times i being preferred over j in the test set is ≥ 0.5) and the number of unique pairs in the test data.

Algorithms:
MC + Copeland: We first complete the empirical preference matrix $\hat{\mathbf{P}}$ using a matrix-completion (MC) routine with rank = 2 and then run the standard Copeland procedure [10] to get a ranking.

MC + Borda: Same as Copeland but the standard Borda algorithm [21] is used instead of Copeland to obtain a ranking from the completed ranking.

Rank Centrality: A spectral ranking algorithm [20] that ranks based on computing a stationary distribution of a Markov chain associated with $\hat{\mathbf{P}}$.

Blade-Chest: A maximum likelihood based algorithm for the Blade-Chest model [7] where we set the dimension of the emdedding to be 8 (Other choices perform poorly, see Supplementary). To obtain a ranking from the estimated MLE, we run the BlockRank2 algorithm of this paper (Other choices perform poorly, see Supplementary).

PariwiseBlockRank: The algorithm described in this paper. To make it more suitable for real-world data, in each recursive call, we use either the ranking given by the algorithm or the Copeland procedure depending on whichever is better for the given data.

The results are presented in Table 1 averaged over 20 splits. As can be seen, the PairwiseBlockRank algorithm performs better or comparably with the best algorithm for most of the datasets. In some cases, it produces similar results as the Copeland algorithm. Furthermore, the Blade-Chest model

Algorithm| Dataset	MC + Copeland	MC + Borda	Rank Centrality	Blade-Chest	Pairwise Block-Rank
DOTA	0.311 ±0.01	0.368 ±0.03	**0.215** ±0.01	0.228 ±0.01	0.23 ±0.03
Tennis	0.314 ±0.005	0.314±0.006	0.285 ±0.008	0.311 ±0.006	**0.281** ±0.006
Sushi-A	0.036±0.02	0.18 ±0.02	0.042 ±0.01	0.06±0.01	**0.034**±0.02
Sushi-B	**0.176**±0.006	0.188±0.005	**0.176** ±0.006	0.203±0.006	**0.176**±0.006
Jester	**0.05** ±0.001	0.23 ±0.002	**0.05** ±0.001	0.15 ±0.022	**0.05** ±0.001

Table 1: Results on real-world datasets. Algorithms with the best results are in boldface.

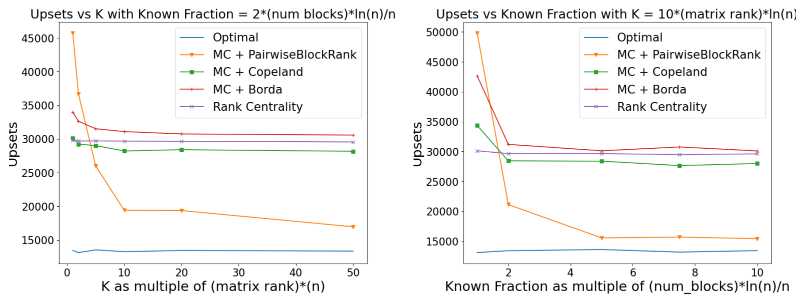

Figure 6: Results for $\mathbf{P}$ under local-global model with 3 blocks. Refer Section 8 for details.

performs poorly for all datasets. The Rank-centrality algorithm performs competitively. To understand this better, we run the following synthetic experiments to compare with rank centrality algorithm.

Synthetic Experiments: The synthetic data was generated to follow the local-global Model. We considered the number of items $n = 600$, and ran experiments on $b = 3$ (see Supplementary for other choices) block cases with equal sized blocks. From the constructed underlying probability preference matrix, we sampled entries according to two parameters, K and known fraction β. Known fraction (β) represents the fraction of total unique pairs that are uniformly chosen for comparison and K represents the number of Bernoulli comparisons for every such pair chosen. We did two types of experiments, one where we fixed the known fraction β and varied K and other where we fixed K and varied β. Thus for every experiment $K.\beta.\binom{600}{2}$ comparisons are made in total. Figure 6 shows the results from varying K and the known fraction. The results are averaged across 10 runs. As can be seen, for the MC + `PairwiseBlockRank` algorithm, the pairwise disagreements quickly approach the optimal value when the known fraction is fixed, and K increases. For the other algorithms, the quality of ranking does not improve with increase in K. Similar trend is observed when the known fraction β is increased while fixing K. The results clearly indicate that with increasing samples, the proposed algorithm quickly converges to the optimal ranking whereas the others perform poorly.

9 Conclusion

In this work, we study the class of parametric preference matrices which induce cyclic tournaments. We propose algorithms that solve the MFAST problem on rank 2 classes and build on the rank 2 class to propose a flexible model called the local-global model. For the general rank r class, we initiate understanding the associated tournament class via the notion of tournament dimension. Going forward, we would like to understand the forbidden configurations of higher rank tournaments and pin down the difficulty in solving the MFAST problem using this approach.

Broader Impact We introduce a flexible parametric model that can capture potentially acyclic relations. The broader impact is that we look at the classical MFAST problem from a novel parametric viewpoint. We believe this can lead to broader impact in fundamental understanding of the hardness of this problem. Algorithms might be subject to bias if the data is inherently biased. The current approach does not focus on removing algorithmic bias that can arise in this fashion.

References

[1] Dota data: `http://www.datdota.com`.

[2] Tennis data: `https://github.com/danielkorzekwa/atpworldtour-api`.

[3] Noga Alon. Ranking tournaments. *SIAM Journal on Discrete Mathematics*, 20(1):137–142, 2006.

[4] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[5] David Causeur and François Husson. A 2-dimensional extension of the bradley–terry model for paired comparisons. *Journal of statistical planning and inference*, 135(2):245–259, 2005.

[6] Pierre Charbit, Stéphan Thomassé, and Anders Yeo. The minimum feedback arc set problem is np-hard for tournaments. *Combinatorics, Probability and Computing*, 16:01–04, 2007.

[7] Shuo Chen and Thorsten Joachims. Modeling intransitivity in matchup and comparison data. In *Proceedings of the ninth acm international conference on web search and data mining*, pages 227–236, 2016.

[8] Xujin Chen, Guoli Ding, Wenan Zang, and Qiulan Zhao. Ranking tournaments with no errors i: Structural description. *Journal of Combinatorial Theory, Series B*, 141:264–294, 2020.

[9] Xujin Chen, Guoli Ding, Wenan Zang, and Qiulan Zhao. Ranking tournaments with no errors ii: Minimax relation. *Journal of Combinatorial Theory, Series B*, 142:244–275, 2020.

[10] Don Coppersmith, Lisa Fleischer, and Atri Rudra. Ordering by weighted number of wins gives a good ranking for weighted tournaments. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 776–782, 2006.

[11] D de Caen. The ranks of tournament matrices. *The American mathematical monthly*, 98(9): 829–831, 1991.

[12] David F Gleich and Lek-heng Lim. Rank aggregation via nuclear norm minimization. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 60–68, 2011.

[13] Ken Goldberg, Theresa Roeder, Dhruv Gupta, and Chris Perkins. Eigentaste: A constant time collaborative filtering algorithm. *information retrieval*, 4(2):133–151, 2001.

[14] Bruce Hajek, Sewoong Oh, and Jiaming Xu. Minimax-optimal inference from partial rankings. *arXiv preprint arXiv:1406.5638*, 2014.

[15] Toshihiro Kamishima. Nantonac collaborative filtering: recommendation based on order responses. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 583–588, 2003.

[16] Raghunandan H Keshavan, Andrea Montanari, and Sewoong Oh. Matrix completion from noisy entries. *The Journal of Machine Learning Research*, 11:2057–2078, 2010.

[17] R Duncan Luce. *Individual choice behavior: A theoretical analysis*. Courier Corporation, 2012.

[18] Rahul Makhijani. Social choice random utility models of intransitive pairwise comparisons. *arXiv preprint arXiv:1810.02518*, 2018.

[19] Rahul Makhijani and Johan Ugander. Parametric models for intransitivity in pairwise rankings. In *The World Wide Web Conference*, pages 3056–3062, 2019.

[20] Sahand Negahban, Sewoong Oh, and Devavrat Shah. Rank centrality: Ranking from pair-wise comparisons, 2015.

[21] Arun Rajkumar and Shivani Agarwal. A statistical convergence perspective of algorithms for rank aggregation from pairwise data. In *International Conference on Machine Learning*, pages 118–126. PMLR, 2014.

[22] Arun Rajkumar and Shivani Agarwal. When can we rank well from comparisons of o (n\log (n)) non-actively chosen pairs? In *Conference on Learning Theory*, pages 1376–1401. PMLR, 2016.

[23] Georg Schökle. Skew symmetric bilinear forms. URL http://www.math.uni-konstanz.de/algebra/ProseminarLA-WS20-21/Handout_Schoeckle_updated.pdf.

[24] Nihar Shah, Sivaraman Balakrishnan, Joseph Bradley, Abhay Parekh, Kannan Ramchandran, and Martin Wainwright. Estimation from pairwise comparisons: Sharp minimax bounds with topology dependence. In *Artificial Intelligence and Statistics*, pages 856–865. PMLR, 2015.

[25] Louis L Thurstone. A law of comparative judgment. *Psychological review*, 34(4):273, 1927.

Checklist

Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 4 and the *Broader Impact* section after conclusion.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See the *Broader Impact* section after conclusion.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] The proofs are included in the supplementary material due to lack of space.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Refer to 8 and supplementary.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] Obtained data from source which gives permission for usage when cited.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A] No personal data is used.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Appendix

Link to Source Code An anonymous link to the source code is present here.

Theorem 12. *(**Transitivity**) Let* $\mathbf{P} \in$. *Let* i, j, k *be three items with embeddings* $h_i, h_j, h_k \in \mathbb{R}^2$ *respectively. If there exits a* $d \in \mathbb{R}^2$ *such that* $h_a^T d > 0 \; \forall a \in \{i, j, k\}$, *then* $\mathbf{T_P}(\{i, j, k\})$ *is acyclic.*

Proof. Let θ_1 be the counterclockwise angle between i and j and θ_2 be the angle between j and k. W.l.o.g assume that $i \succ_{\mathbf{P}} j$ and $j \succ_{\mathbf{P}} k$. Then $0 < \theta_1, \theta_2 < \pi$. The relationship between i and k is determined by $\theta_1 + \theta_2$. However, the existence of a $d \in \mathbb{R}^2$ satisfying the condition of the theorem implies that the embeddings of the three items $\{i, j, k\}$ lie in the same half space. This implies that $0 < \theta_1 + \theta_2 < \pi$ which further implies that $i \succ_{\mathbf{P}} k$. The result follows. □

Proof of Lemma 1

Proof. Let $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_k\}$ induce a tournament $\mathbf{T}$. As $T_{ij} = \text{sign}(\mathbf{x}_i^T A^{\texttt{rot}} \mathbf{x}_j)$ by definition, $\mathbf{T}$ is associated with the skew symmetrix matrix

Proof of Theorem 2

Proof. Assume that the $\texttt{TourDim}(d) = k$. Thus Player 1 has a winning strategy for $k+1$ i.e., there exists a dataset $\mathcal{D}$ of size $k+1$ that induces a tournament $\mathbf{T} \in \mathcal{T}_d^{k+1}$ such that Player 2 cannot produce a $\mathbf{T}$ preserving mapping f such that one can find a $\mathbf{w}$ where $f(\mathbf{x}_i) = y_i$ for all i. Note that the ability of Player 2 to find such a $\mathbf{w}$ is equivalent to the ability of adding one more node to the tournament $\mathbf{T}$ whose direction with respect to the node i of $\mathbf{T}$ is given by the label y_i where the new tournament would belong to $\mathcal{T}_d^{k+2}$. This is true as this new tournament with the extra node is induced by the dataset $\{\mathcal{D} \cup (A^{\texttt{rot}})^{-1}\mathbf{w}\}$ (Note that $A^{\texttt{rot}}$ is invertible). As Player 2 is unable to produce such a mapping and taking into account that both $\mathcal{D}$ (and so $\mathbf{T}$) and y_i's were chosen strategically by Player 1, this implies that no tournament in $\mathcal{T}_d^{k+2}$ can contain $\mathbf{T}$ as an induced sub configuration. Finally consider any $n \geq k+1$ and let $\mathbf{T} \in \mathcal{T}_d^n$. As $\mathbf{T}$ restricted to any subset of $k+1$ nodes must belong to $\mathcal{T}_d^{k+1}$, the result follows. □

Theorem 3: The bounds for ℓ must be as follows: $r + 2 \leq \ell \leq 2^r + r + 1$ and not r and 2^r as stated in the submission.

Proof of Theorem 3

Proof. By definition $\ell = \texttt{TourDim}(r) + 2$. Thus we need to show that $\texttt{TourDim}(r) \geq r$. Assume not. Then Player 1 has a winning strategy for r i.e., there exists a dataset $\mathcal{D}$ of size r such that there does not exist a $\mathbf{T}$ preserving mapping such that one can find a $\mathbf{w}$ with $f(\mathbf{x}_i) = y_i$ for all i. However, it is always possible to perturb the datapoints randomly by a small ϵ amount such that the perturbed $\mathcal{D}$ still induces $\mathbf{T}$ and the data points are linearly independent with probability 1. For such a perturbed dataset, there must exist a $\mathbf{w}$ that achieves the labeling given by y_i - just solve for $\mathbf{X}\mathbf{w} = \mathbf{y}$ where $\mathbf{X} \in \mathbb{R}^{d \times d}$ contains the data points in the perturbed $\mathcal{D}$ as its columns. Thus, we arrive at a contradiction. This proves the lower bound.

To prove the upper bound, fix an arbitrary tournament $\mathbf{T} \in \mathcal{T}_r^{r+1}$ on $r+1$ nodes where each node has an embedding $\mathbf{x}_i \in \mathbb{R}^r$. Then $\mathbf{x}_{r+1} = \sum_{i=1}^r c_i \mathbf{x}_i$ for some constants c_is. Assume wlog that $c_i \neq 0 \;\; \forall i$ (If some set of $c_i = 0$, then the bound we get would be tighter). Now consider adding an extra node to $\mathbf{T}$ whose directions with the $r+1$ nodes are given by the label vector $y \in \{\pm 1\}^{r+1}$ where $y_{r+1} = -1$. If this new tournament is not a forbidden configuration for $\mathcal{T}_r^{r+2}$, then there must exist some $z \in \mathbb{R}^r$ such that $\text{sign}(z^T A^{\texttt{rot}} \mathbf{x}_i) = y_i$ for all $i = 1, \ldots, r+1$. We argue that such a z invalidates a particular sign pattern for the coefficients c_i's. Indeed, the following is true:

$$z^T A^{\texttt{rot}} \mathbf{x}_{r+1} = \sum_{i=1}^r c_i (z^T A^{\texttt{rot}} \mathbf{x}_i)$$

$y_{r+1} = \text{sign}(z^T A^{\texttt{rot}} \mathbf{x}_{r+1}) = -1 \implies \text{sign}(c_i) = y_i \; \forall i$ is not possible as it would lead to a contradiction since $\text{sign}(z^T A^{\texttt{rot}} \mathbf{x}_i) = y_i$.

As the choice for y_i for $i = 1, \ldots, r$ is arbitrary, one can keep attempting to add nodes to the tournament with all possible sign patterns with respect to the first r nodes. Indeed after adding 2^r such nodes exhausting all sign patterns for $y_1, \ldots, y_r$, one invalidates all possible sign patterns for the coefficients $c_1, \ldots c_r$. But this is a contradiction. Thus, starting from an arbitrary tournament on $\mathbf{T} \in \mathcal{T}_r^{r+1}$ we arrive at a forbidden configuration for $\mathcal{T}_r^{r+1+2^r}$ nodes.

Finally, for a general $n \geq r + 1 + 2^r$, if $\mathbf{T} \in \mathcal{T}_r^n$, any induced tournament on $r + 1 + 2^r$ nodes must belong to $\mathcal{T}_r^{r+1+2^r}$ and so the above argument suffices. □

Proof of Corollary 1

Proof. From Theorem 3, it follows that the forbidden configuration for $\mathcal{T}_2^n$ must be of size at least 4. We will show that this is indeed exactly equal to 4 by showing that $\texttt{TourDim}(2) = 2$. To show this, we need to demonstrate a strategy for player 1 to win with 3 data points in $\mathbb{R}^2$. Let the data points selected by player 1 induce the 3 cycle as the tournament and the corresponding labels be $y_1 = y_2 = y_3 = 1$. Any $\mathbf{T}$ preserving mapping should necessarily map the datapoints to some $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3 \in \mathbb{R}^2$ where the counterclockwise angle between $\mathbf{x}_1$ and $\mathbf{x}_2$ is less than 180 degrees and $-\mathbf{x}_3 \in \text{cone}\{\mathbf{x}_1, \mathbf{x}_2\}$. If not, one cannot induce the 3-cycle as a tournament. Then for any $\mathbf{w} \in \mathbb{R}^2$ such that $\text{sign}(\mathbf{w}^T\mathbf{x}_1) = \text{sign}(\mathbf{w}^T\mathbf{x}_2) = 1$, it must necessarily be the case that $\text{sign}(\mathbf{w}^T\mathbf{x}_3) = -1$. However $y_3 = 1$ and so Player 1 has a winning strategy.

Thus $\texttt{TourDim}(2) <= 2$. But it is trivial to verify that $\texttt{TourDim}(2) \geq 2$ as Player 2 always has a winning strategy with just 2 data points. Thus $\texttt{TourDim}(2) = 2$ and so from Theorem 2 it follows that $\mathcal{T}_2^n$ has a forbidden configuration of size 4. □

Proof of Lemma 4

Proof. Consider the sub-tournament of four items $\{a_1, a_2, a_3, a_4\}$ labelled $\mathbf{T}$. If $\mathbf{T}$ corresponds to an acyclic graph, there is nothing to prove. Therefore only the case where T contains at least 1 cycle is considered. Without loss of generality, let the items which form a cycle have indices a_1, a_2 and a_3. Let the corresponding embeddings of each item i to be $[u_i, v_i]$. For a_4 to not violate the rank 2 assumption, it too must have a corresponding embedding by which its interactions with the existing items are defined. Let the embedding of a_4 be $[u_4, v_4]$. Without loss of generality, let $a_1 \succ_{\mathbf{P}} a_2 \succ_{\mathbf{P}} a_3 \succ_{\mathbf{P}} a_1$. Let the angle between the embeddings of items a_1, a_2 be α and items a_2, a_3 be β. Since $a_3 \succ_{\mathbf{P}} a_1$, we have $\alpha + \beta > \pi$. Now consider the angle between a_1, a_4 to be θ. The following cases are possible.

$$\theta \in (0, \alpha) \implies a_1 \succ a_4 \succ a_2$$

$$\theta \in (\alpha, \alpha + \beta) \implies a_2 \succ a_4 \succ a_3$$

$$\theta \in (\alpha + \beta, 2\pi) \implies a_3 \succ a_4 \succ a_1$$

In none of these cases can the item a_4 succeed, or precede all of the 3 existing items. As the set of items chosen was arbitrary, the result follows. □

Proof of Theorem 5

This theorems follows naturally from Equation 1 obtained as part of the proof for Theorem 6. Given Equation 1, it follows that $S_{i+k+1 \ \mathbf{mod}(2k+1)} \succ S_i \succ S_{i+k \ \mathbf{mod}(2k+1)}$.

Proof of Theorem 6

Proof. This proof in constructive in nature. Consider the set of items in the tournament to be represented as a_i. For every item a_i in the tournament, define S_i to be the maximal set of items associated with a_i such that $|S_i| < n$ and S_i satisfies the following

$$\forall a_k \in S_i, a_j \notin S_i, a_k \succ_{\mathbf{P}} a_j \text{ iff } a_i \succ_{\mathbf{P}} a_j$$

By definition S_i is non-empty as it contains a_i and is not equal to $[n]$ as $|S_i| < n$. It can be easily observed that the item to set mapping is an equivalence relation. Let the set of all unique constructed sets be named $Q = \{S_1, S_2, ...S_\ell\}$ for some ℓ . Note that these sets are disjoint due to the equivalence relation present between the items and the sets in Q. Let V_i be a vector corresponding to any one of the items in the set chosen arbitrarily. This vector representation cannot capture the exact probabilities between the items but has the following property.

$$\sin\theta > 0 \iff S_i \succ_{\mathbf{P}} S_j \text{ where } \theta \text{ is the counterclockwise angle between } V_i \text{ and } V_j$$

Consider an ordering of Q determined by the corresponding V_i, such that the V_i are in counterclockwise order. Consider the set of sets corresponding to V_i that precede V_1 to be Q_1 and the set of sets which correspond to vectors preceded by V_1 to be Q_2. It is clear from Theorem 12 that the sets in Q_1 have a transitive relationship amongst them. The same applies for Q_2. Consider two consecutive sets (consecutive defined w.r.t to the angle between the representative vectors) $\{V_1\} \cup Q_1$, say X_1 and X_2. From the maximal property of the sets, we know that there must exist another set X such that

$$X_1 \succ X \succ X_2$$

or

$$X_2 \succ X \succ X_1$$

We now introduce the term 'set separator' to refer to the set X i.e, X is the separator of X_1 and X_2. There is a transitive relationship among the items of $V_1 \cup Q_1$, therefore X must lie in Q_2. Similarly, the separator of any two consecutive sets in $\{V_1\} \cup Q_2$ must lie in Q_1. Also, the set separator for a unique pair of successive sets must be unique. To prove this by contradiction, consider 2 pairs of successive sets in Q_1, X_1, X_2 and X_3, X_4. Let $Y \in Q_2$ be the separator for X_1, X_2 and X_3, X_4 such that $X_1 \succ X_2 \succ X_3 \succ X_4$. This leads to one of the forbidden configurations shown in Figure 4 being obtained for the subset $\{X_1, X_2, Y, X_3\}$. Since each pair of successive sets in Q_1 has a corresponding set in Q_2, and vice versa, the number of sets in Q_1 and Q_2 must be equal. Let k be the number of items in Q_1. The total number of sets present in Q must be $2k + 1$. Let the sets in Q now be ordered as $\{S_1, S_2 \ldots S_{2k+1}\}$ such that the sets are in counterclockwise order w.r.t. their corresponding vectors. In the above construction, the set Q is split using the vector V_1. Similarly, the set Q can by split by using any vector V_i(Corresponding to set S_i). Since the proof above generalizes to any split, the following relationship between the sets holds true.

$$S_i \succ S_j \iff i - j(mod 2k + 1) \leq k \tag{1}$$

Proof of Part(a) Using the claims proven in **??**, it is sufficient to prove that all the items belonging to a single set lie in the same half-plane. This is trivially true since all items in a set S_i lie in the same half-plane split by when the plane is split by any of the items due to the maximal property of the sets.

Proof of Parts (b) (c) (d)) We now construct the sets $A_1, A_2, \ldots, A_k, B_1, B_2, \ldots, B_k, C$. Let C by the set S_1. Consider

$$A_i = S_{i+1}$$

and

$$B_i = S_{i+k+1}$$

Since $A_1, A_2, \ldots, A_k$ and $B_1, B_2, \ldots, B_k$ are consecutive sets which lie in the same half place when S_1 is used to split the plane, the transitive properties on the sets represented by A_i and B_i holds. From our construction of sets $A_1, A_2, \ldots, A_k$ and $B_1, B_2, \ldots, B_k$ and the properties of the sets S_i proven above, parts (c) and (d) hold true. □

Proof of Theorem 7

Proof. We show this using construction. A $\mathbf{T} \in \mathcal{T}_2^n$ with a cycle of any length, k, can be constructed as follows. Let the embedding corresponding to item i_k be $[\cos\frac{2j\pi}{k}, \sin\frac{2j\pi}{k}]$. It is clear that $i_t^T A^{\texttt{rot}} i_{t+1} > 0 \forall t \in 1..k-1$ and $i_k^T A^{\texttt{rot}} i_1$.

To show this model can represent a size 3 cycle with any fixed pairwise probabilities, fix an arbitrary triplet $\{i, j, k\}$. Let $q_\ell = \ln(p_\ell/(1-p_\ell))\ \forall \ell \in \{1,2,3\}$ and the embedding, $[u_i, v_i]$, corresponding to each item i. By setting $u_i = 1, v_i = 0, u_j = 0, v_j = q_1, u_k = -q_2/q_1$ and $v_k = -q_3$, the arbitrary triplet can be satisfied for all values of $p_\ell \neq 0.5, \ell \in \{1,2,3\}$. By definition of $\mathcal{P}$, values of 0.5 are disallowed and the result follows. □

Proof of Theorem 8

Proof. First we aim to prove that the optimal ranking for $\mathbf{T} \in \mathcal{T}_2^n$ is one of the n cyclic shifts of one of the permutations of the items. Furthermore, the permutation of the items considered is a counterclockwise ordering the corresponding vectors, and is proven using induction. Using Theorem 6, we can partition the set of all items into $2k+1$ groups. We can first show that the items of a single group must appear in consecutive positions in one of the optimal rankings. This is proven as follows.

Consider there exists an optimal ranking with items which belong to the same group not occurring consecutively. Consider two items belonging to the same group, which have items from other groups present in between them in the ranking. Consider these items to be a_1, a_2, with a_1 present above in the rankings. Consider the number of upsets that the two items are involved in to be u_1 and u_2. If $u_1 \leq u_2$, a_2 can be placed right after a_1 in the ranking, creating a better or equivalent ranking in terms of upsets. Similarly if $u_1 \geq u_2$, a_1 can be placed directly above a_2 in the rankings to create an equivalent or better ranking. Therefore there exists an optimal ranking which has all items in the same group consecutively.

This theorem is then reduced to finding a ranking of groups, which is proven using induction on k.

Base Case

Consider the base case with $k = 1$. Let there be 3 groups, C, A_1, B_1. We can say that the optimal ranking cannot be any of the following

$$CB_1A_1$$
$$A_1CB_1$$
$$B_1A_1C$$

since all three rankings can be made better by swapping the second and third ranked groups. Therefore the 3 possible optimal rankings are

$$CA_1B_1$$
$$A_1B_1C$$
$$B_1CA_1$$

which are cyclic shifts of each other.

Inductive Step

One property of rankings which is useful for the inductive step proof is as follows. Let there be $2k+1$ groups $G = \{g_1, g_2 \dots g_{2k+1}\}$. Label the optimal ranking with the condition that g_i be placed first in the ranking as R_i. The ranking R_i with g_i removed must be the optimal ranking for $G \setminus \{g_i\}$. This can be shown using contradiction i.e, if there was a better ranking for $G \setminus g_i$, that ranking with g_i appended to the front would be better than R_i.

We now assume the theorem is true for size $2k-1$ instances and aim to prove for the same for size $2k+1$ instances. Consider g_1 as the first group in the ranking. This creates a certain number of upsets, for the purposes of ranking the remaining groups, 2 of the remaining groups can be merged into a single group. This follows from the observation in the proof of Theorem 6 that each group also 'separates' two groups. This can be considered an instance of the size $2k-1$ problem. Therefore the set of optimal rankings with g_1 as the first group in the rankings is made up of g_1 as the first group and a cyclic sweep of the remaining items to fill the remaining positions. Therefore the optimal permutation must be among the sets created by considering each of the $2k+1$ groups as the first group in the rankings. Let $R_{i,j}$ represent the ranking which has group g_i as the first group and the remaining groups present as a cyclic sweep from g_j. Consider the case of $R_{1,k}$. Let x_i represent the number of items in group g_i. If $R_{1,k}$ is a better ranking than $R_{k,k+1}$, it implies that

$$\sum_{i=k}^{n+1} x_i > \sum_{i=n+2}^{2n+1} x_i \tag{2}$$

by considering the shift of g_1 in the two rankings. The difference in the number of upsets between $R_{1,2}$ and $R_{1,k}$ is given by

$$x_2(-x_k - x_{k+1} \ldots - x_{n+2} + x_{n+3} \ldots x_{2n+1}) + x_3(-x_k - x_{k+1} \ldots - x_{n+3} + x_{n+4} \ldots x_{2n+1}) \ldots$$

Using Equation 2, it can be seen that each of the above terms are negative for any $j \leq n+1$, making $R_{1,2}$ the better ranking. Any $j > n+1$ cannot be considered as an optimal ranking since the first group as per the ranking must precede the second(otherwise switching them would decrease the upsets). Since either $R_{1,2}$ or $R_{k,k+1}$(both counterclockwise orderings) is better than $R_{1,k}$ whenever $k \leq n+1$($R_{1,k}$ cannot be the optimal ranking when $k > n+1$), and since this can be generalised for any $R_{i,j}$, it is shown that one of the counterclockwise orderings of the items is the optimal ranking. This shows that a geometric sweep of items when embeddings are known provides an optimal ranking. Using Theorem 12, we it can be proven that R2R can find an optimal ranking without using the embeddings. Considering a cycle a, b, c is found, let A, B, C be defined as in Algorithm 2. From this it is clear that the counterclockwise ordering of arms will follow $A \succ B \succ C \succ A$, but the ordering within each set remains unknown. Since Theorem 12 can be applied on A, B, C by setting d as c, a, b respectively, all 3 sets must be total orders (acyclic). Therefore, the counterclockwise sweep order is obtainable by performing a topological sort. Concatenating the topological sorts of A, B, C should result in a permutation which corresponds to a counterclockwise sweep being created. This is done in Algorithm 2 since the recursive calls of the algorithm on sets A, B, C will result in topological sorts being returned. Since the algorithm is able to find a counterclockwise sweep, and considers all cyclic shifts of the permutation, it is optimal. □

Proof of Polynomial Time Complexity The proof given here is stronger than the stated theorem, here we show that `R2R` has a polynomial running time for any instance. `R2R` first aims to find a cyclic triplet. This can be done in polynomial time in multiple ways. If a cyclic triplet is not found, the algorithm is polynomial time since a Topological sort is sufficient to obtain the optimal ranking. If a cycle is found, the recurrence relation, $T(n) = 3T(\frac{n}{3}) + P(n)$ where $P(n)$ is a polynomial which represents the time taken to check and compare the upsets of all n cyclic permutations. By solving this recursive equation, it can be shown that $T(n)$ is polynomial in nature.

Proof of Theorem 9

Proof. This algorithm is based on Theorem 8. `BR2R` first tries to find a cyclic triplet, these 3 items must belong to the same block, B, since there cannot be cycles or items which belong to different blocks.

If a cycle is not found, the time complexity analysis and optimality is trivial. The optimality is guaranteed by the properties of Topological sorting, and the time complexity is polynomial in n.

If a cycle is found, the remaining items can be put into 3 groups. The first group is the set of items that precede all 3 items in the cyclic triplet. These items cannot belong to B since one of the forbidden structures, as seen in 4, would be formed. This is also true for the second group, which consists of all the items which are preceded by all 3 items in the cyclic triplet. The third group of items consists of items which belong to the block B. Since the blocks are ordered, an optimal ranking can be constructed once the optimal rankings are known for each of the 3 groups by simply appending the rankings. Therefore we can construct the recurrence relationship for the time complexity of `BR2R`

$$T(n) = P(|B|) + T(n - |B| - |F|) + T(|F|)$$

where $|F|$ is the size of the group of items which precede all 3 items of the cyclic triplet. By assuming the $T(n)$ is not sub-linear

$$T(n) = P(|B|) + T(n - |B|)$$

Since $|B| >= 3$, $T(n)$ a polynomial, making `BR2R` a polynomial time algorithm. Since the ranking obtained by the above algorithm follows the block structure i.e, arms belonging to higher blocks are ranked higher, and optimal ranking is used to rank arms in the same block, the ranking over all arms produced must be optimal as well. □

Proof of Lemma 10

Proof. Consider the definition of $\mathbf{T} \in \texttt{Block-}\mathcal{T}_2^n$.

$$\Bigg\{ \mathbf{T} \in \texttt{Block-}\mathcal{T}_2^n : \exists \mathbf{s}, \mathbf{u} \in \mathbb{R}^n, \text{ a partition } \{H_k\}_{k=1}^b \in [n] \text{ s.t}$$
$$\phi(T_{ij}) = \left\{ \begin{array}{l} u_i v_j - v_i u_j \text{ if } \exists \ell \text{ s.t } i, j \in H_\ell \\ s_i - s_j \text{ otherwise} \end{array} \right\} \Bigg\}$$

The $\mathbf{T}$ defined above can be expressed in the form $Q - Q^T$ where Q is

$$Q_{ij} = \left\{ \begin{array}{l} u_i v_j \text{ if } \exists \ell \text{ s.t } i, j \in H_\ell \\ s_i \text{ otherwise} \end{array} \right\} \Bigg\}$$

The rank of Q can be established as $2b$ as follows. Consider a sub-matrix of Q defined as

$$Q_k = [Q_{ij}] \forall i \in [n] \forall j \in H_k$$

Each column of this matrix can be expressed as a linear combination of two vectors U_k, S_k which are defined as

$$U_k = \left\{ \begin{array}{l} u_i \text{ if } i \in H_k \\ 0 \text{ otherwise} \end{array} \right.$$

$$S_k = \left\{ \begin{array}{l} s_i \text{ if } i \notin H_k \\ 0 \text{ otherwise} \end{array} \right.$$

A column whose index in the original matrix Q is j can be expressed as $U_k v_j + s_k$. Since each sub-matrix is of rank 2, Q must be of rank $2b$. Since the rank of the sum of two matrices is upper bounded by the sum of the ranks of the two matrices, the rank of $\mathbf{T}$ must be less than or equal to $4b$. Therefore $\mathbf{T} \in \mathcal{T}_r^n$ for some $r \leq 4b$. □

Proof of Theorem 11

Proof. To prove this theorem, we assume the matrix completion subroutine used in Algorithm 4 is the OPTSPACE algorithm of (16). From Lemma 10 we know that the rank of a local global model $\mathbf{P}$ with b blocks is at most $4.b$ and this is also passed as an input to the Matrix completion routine. The proof of the theorem closely follows the proof of Theorem 13 in (22)). Specifically, from Equation 8 in the proof of the theorem, under similar sample complexity $O(nb\log(n)$ samples each compared $O(b\log(n)$ times and for any $0 < \epsilon < \frac{1}{2}$, the Frobenius norm difference of the completed matrix and the true low rank matrix can be bounded as follows:

$$\|\hat{\mathbf{M}} - \mathbf{M}\|_F^2 \leq \frac{n^2 \epsilon \min(\theta^2, \Delta^2)}{4} \leq \frac{n^2 \epsilon \Delta^2}{4}$$

where $\mathbf{M} = \phi(\mathbf{P})$

Under the condition that $\epsilon \leq \frac{\theta^2}{n^2 \Delta^2}$, this further can be bounded by

$$\|\hat{\mathbf{M}} - \mathbf{M}\|_F^2 \leq \frac{\theta^2}{4}$$

This implies that for every (i, j) pair where i and j are in different blocks, it must necessarily be the case that $|\hat{M}_{ij} - M_{ij}| \leq \frac{\theta}{2}$. This implies then that in the reconstructed tournament using $\hat{\mathbf{M}}$,

$i \succ_{\mathbf{T}(\hat{\mathbf{M}})} j$ if and only if $i \succ_{\mathbf{T_P}} j$. This guarantees that the tournament $\mathbf{T}(\hat{\mathbf{M}})$ would be the same as $\mathbf{T_P}$ upto the blocks. Once the tournament is reconstructed, the algorithm runs the BR2B routine on $\mathbf{T}(\hat{\mathbf{M}})$. The BR2B algorithm starts by looking for a cyclic triplet if one exists. As a cycle cannot exist across blocks, the BR2R algorithm will necessarily find a cycle (if one exists) within a block. Thus the sets S^+ and S^- would contain complete blocks in every recursive call. This further implies that the final concatenated ordering produced by the BR2R algorithm respects the blocks. Thus, the result follows. □