On the Distance Between CP-nets

Andrea Loreggia
University of Padova
Padova, Italy
andrea.loreggia@gmail.com

Nicholas Mattei
IBM Research
Yorktown Heights, USA
n.mattei@ibm.com

Francesca Rossi
IBM Research & University of Padova
Yorktown Heights, USA
francesca.rossi2@ibm.com

K. Brent Venable
Tulane University & IHMC
New Orleans, USA
kvenabl@tulane.edu

ABSTRACT

Preferences play a key role in decision making by both single individuals and/or groups. In a multi-agent context, it is also important to know how to aggregate preferences to reach a collective decision. Moreover, being able to measure the distance between the preference of two individuals is important to identify the amount of disagreement and possibly reach consensus. In this paper we define a notion of distance between CP-nets, a formalism that can compactly encode conditional qualitative preferences. We consider the Kendall-tau distance between the partial orders induced by CP-nets, and we define two tractable approximations of that distance, which can be computed in time polynomial in the number of features of the CP-nets. We then perform experiments to demonstrate the quality of these approximations compared to the Kendall-tau distance. We also relate our two notions of distance to the distance rationalizability of sequential plurality voting for CP-nets.

KEYWORDS

CP-nets; preference reasoning; social choice

ACM Reference Format:
Andrea Loreggia, Nicholas Mattei, Francesca Rossi, and K. Brent Venable. 2018. On the Distance Between CP-nets. In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), Stockholm, Sweden, July 10–15, 2018,* IFAAMAS, 9 pages.

1 INTRODUCTION

Preferences are ubiquitous in real-life, and are central to decision making, whether the decision is made by a single individual or by a group. The study of preferences in computer science has led to important theoretical and practical results across a range of areas where decisions need to be made [17, 34]. CP-nets provide an effective compact way to qualitatively model and reason with preferences over outcomes when the preferences have a combinatorial structure [7]. Moreover, CP-nets provide a way to model not only subjective preferences, but also priorities and optimization criteria, thus allowing for a homogeneous modeling and reasoning framework where a seamless integration of several optimization and preference reasoning modalities are supported [35].

Besides modeling, learning, reasoning with, and aggregating preferences, it is often useful to be able to measure the distance between the preferences of two individuals, or between a group and an individual, in order to measure the amount of disagreement and possibly get closer to a consensus. A notion of distance can also be useful in the presence of exogenous priorities in addition to the subjective preferences of the decision maker. These priorities can be derived from ethical principles, feasibility constraints, or business values [6, 14, 37, 38]. When preferences and certain external priorities are in conflict, the priorities should override the subjective preferences of the decision maker. For example, in a hiring scenario, the preferences of the hiring committee members over the candidates should be measured against guidelines and laws e.g., ensuring gender and minority diversity. Therefore, it is essential to have principled ways to evaluate if preferences are compatible with a set of priorities, and to measure any deviations. Hence, the ability to precisely quantify the distance between preferences and external priorities provides a way to detect deviations, and possibly suggest more compliant decisions [27, 28].

In this paper we define a notion of distance (formally a distance function or metric) between CP-nets. CP-nets are a compact representation of a partial order over outcomes, so the ideal notion of distance would be a distance between the underlying partial orders of the CP-net. We generalize the classic Kendall's τ distance (KTD) [23], which counts the number of inverted pairs between two complete, strict linear orders. We add a penalty parameter p defined for partial rankings as in Fagin et al. [21], and use *KTD* as a baseline to compare partial orders. However, the size of the induced orders is exponential w.r.t. the CP-net, and we conjecture that computing a distance directly between the induced partial orders is intractable because of this possibly exponential expansion.

To achieve tractability, we define two distances between CP-nets, that we call *O-Legal CP-net Distance (*O-*CPD)* and *Induced CP-net Distance (I-CPD)* that do not require enumerating the entire partial orders of the underlying CP-nets but rather analyzes the dependency structure of the CP-nets and their CP-tables in order to compute the distance. These notions of distance are an approximation of the KTD, i.e., the true distance, between the induced partial orders of the CP-net. Our measures leverage the compact representation of the partial orders induced by the CP-net to achieve this approximation. *O*-CPD performs a little better in scenarios where CP-nets are *O*-legal [24], i.e., when the CP-nets share a topological ordering of their underlying features, while I-CPD does not require this assumption over the structure of the CP-nets.

An important and interesting property of both *O*-CPD and I-CPD is that when the KDT between two CP-nets is equal to zero

Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), M. Dastani, G. Sukthankar, E. André, S. Koenig (eds.), July 10–15, 2018, Stockholm, Sweden.

then both of our distance measures are also zero. This only happens when the two CP-nets have the same dependency structure and CP-tables and thus induce the same partial order. We achieve the same results when the (normalized) KDT is equal to one, i.e., when the two CP-nets have the same structure but completely reversed CP-statements. This condition means that the induced partial orders are inverted and hence $KTD = 1$. Unfortunately, when KTD is not equal to zero or one, then both O-CPD and I-CPD can be larger or smaller than the true value of KTD. In the case of O-CPD, errors are accumulated relative to the number of incomparable pairs in the induced partial order. When a pair is incomparable in either of the two CP-nets, or incomparable in one or not the other. Likewise, for I-CPD errors are only introduced due to incomparable pairs as we do not fully expand these parts of the induced partial order.

Hence, to provide bounds on the error of both O-CPD and I-CPD we quantify the number of incomparable pairs that can occur in a CP-net. We prove that it is polynomial to compute the number of incomparable pairs of outcomes in a separable CP-net. Non-separable CP-nets have fewer incomparable pairs of outcomes, since each dependency link eliminates at least one incomparable pair. However, these theoretical bounds are very loose. For this reason, we also perform an experimental analysis of the relationship between the two distance functions and KTD, which shows that the average error is never more than 10%.

Contribution. We define two novel distance functions between CP-nets that generalizes the Kendall τ distance between the underlying partial orders. We conjecture that this distance is hard to compute and define two approximation of this distance which can be computed in polynomial time. We provide bounds on this approximation based on the number of incomparable pairs in a CP-net and perform empirical experiments to show that our approximation is never more than 10% away from the true distance, i.e., the Kendall-tau distance between the induced partial orders. We also show an interesting link between our novel metrics and distance rationalizability for voting rules defined over CP-nets.

2 PRELIMINARIES

Our preferences may apply to one or more of the individual components, rather than to an entire decision. For example, if we need to choose a car, we may prefer certain colors over others, and we may prefer certain brands over others. We may also have conditional preferences, such as in preferring red cars if the car is a convertible. For these scenarios, the CP-net formalism [7] is a convenient and expressive way to model preferences [15, 22, 35]. CP-nets indeed provide an effective compact way to qualitatively model preferences over outcomes (that is, decisions) with a combinatorial structure. The CP-net formalism is intuitively easy to understand and provides efficient optimization reasoning [1, 12]. Moreover, in a collective decision making scenario, several CP-nets can be aggregated, e.g., using voting rules [13, 16, 29], to find compromises and reach consensus among decision makers.

2.1 CP-nets

CP-nets [7] (for Conditional Preference networks) are a graphical model for compactly representing conditional and qualitative preference relations. They are sets of *ceteris paribus* preference statements (cp-statements). For instance, the cp-statement *"I prefer red wine to white wine if meat is served."* asserts that, given two meals that differ *only* in the kind of wine served *and* both containing meat, the meal with red wine is preferable to the meal with white wine. Formally, a CP-net has a set of features (often called variables) $F = \{X_1, \dots, X_n\}$ with finite domains $D(X_1), \dots, D(X_n)$. We use X_i to denote a feature and x_i to denote the literal assigned to feature i. For each feature X_i, we are given a set of *parent* features $Pa(X_i)$ that can affect the preferences over the values of X_i. This defines a *dependency graph* in which each feature X_i has $Pa(X_i)$ as its immediate predecessors. Consequently, for each feature X_i we also define a set of *successor* features $Succ(X_i)$ which is the set of features that directly depend on X_i. An *acyclic* CP-net is one in which the dependency graph is acyclic. Given this structural information, one needs to specify the preference over the values of each feature X_i for *each complete assignment* on $Pa(X_i)$. This preference is assumed to take the form of a total or partial order over $D(X_i)$. A cp-statement has the general form $x_1 = v_1, \dots, x_n = v_n : x_i = a_1 \succ \dots \succ x_1 = a_m$, where $Pa(X_i) = \{x_1, \dots, x_n\}$, $D(X_i) = \{a_1, \dots, a_m\}$, and $\succ$ is a total order over such a domain. The set of cp-statements regarding a certain feature X_i is called the cp-table for X_i.

Consider a CP-net whose features are A, B, C, and D, with binary domains containing f and $\overline{f}$ if F is the name of the feature, and with the cp-statements as follows: $a \succ \overline{a}$, $b \succ \overline{b}$, $(a \wedge b) : c \succ \overline{c}$, $(\overline{a} \wedge \overline{b}) : c \succ \overline{c}$, $(a \wedge \overline{b}) : \overline{c} \succ c$, $(\overline{a} \wedge b) : \overline{c} \succ c$, $c : d \succ \overline{d}$, $\overline{c} : \overline{d} \succ d$. Here, statement $a \succ \overline{a}$ represents the unconditional preference for $A = a$ over $A = \overline{a}$, while statement $c : d \succ \overline{d}$ states that $D = d$ is preferred to $D = \overline{d}$, given that $C = c$.

A *worsening flip* is a change in the value of a feature to a less preferred value according to the cp-statement for that feature. For example, in the CP-net above, passing from $abcd$ to $ab\overline{c}d$ is a worsening flip since c is better than $\overline{c}$ given a and b. One outcome α is *better* than another outcome β (written $\alpha \succ \beta$) if and only if there is a chain of worsening flips from α to β. This induces a preorder over the outcomes, which is a partial order if the CP-net is acyclic.

In general, finding the optimal outcome of a CP-net is NP-hard [7]. However, in acyclic CP-nets, there is only one optimal outcome and this can be found in linear time by sweeping through the CP-net, assigning the most preferred values in the cp-tables. For instance, in the CP-net above, we would choose $A = a$ and $B = b$, then $C = c$, and then $D = d$. In the general case, the optimal outcomes coincide with the solutions of a set of constraints obtained replacing each cp-statement with a constraint [9]: from the cp-statement $x_1 = v_1, \dots, x_n = v_n : x_1 = a_1 \succ \dots \succ x_1 = a_m$ we get the constraint $v_1, \dots, v_n \Rightarrow a_1$. For example, the following cp-statement (of the example above) $(a \wedge b) : c \succ \overline{c}$ would be replaced by the constraint $(a \wedge b) \Rightarrow c$.

2.2 O-legality and Linearization

When CP-nets are used for collective decision making we are typically given a collection of CP-nets, P, called a profile in the voting literature, over m common features with binary domains. A popular restriction for a collection of CP-nets is the assumption of O-legality [24]. For a collection of CP-nets to be O-legal, there must exist a total order O over the m features in P. Hence, for every CP-net in

P, each feature X_i in all the CP-nets is independent with respect to features that follow it in O. Given a feature X_i the function $flw(X_i)$ returns the number of features following X_i in O.

Every acyclic CP-net is satisfiable, i.e., it is possible to produce a linearization of the partial order such that each cp-statement in the CP-net is satisfied [7]. One could compute a distance between two CP-nets by comparing a linearization of the partial orders induced by the two CP-nets, provided we use the same algorithm to linearize the partial order. However, this intuitively simple method is likely intractable, we can use it as a starting point.

We only consider linearization generated using Algorithm 1 [7]. This algorithm works as follows: Given an acyclic CP-net A over n features and a ordering O to which the CP-net A is O-legal, then there is at least one feature with no parents. If more than one feature has no parents, then choose the one that comes first in the ordering O; let X be such a feature. Let $x_1 \succ x_2$ be the ordering over $Dom(X)$ dictated by the cp-table of X. For each $x_i \in Dom(X)$, construct a CP-net, N_i, with the $n-1$ features $V - X$ by removing X from the initial CP-net, and for each feature Y that is a child of X, revising its CPT by restricting each row to $X = x_i$. We can construct a preference ordering $\succ_i$ for each of the reduced CP-nets N_i. For each N_i recursively identify the feature X_i with no parents and construct a CP-net for each value in $Dom(X_i)$ following the same algorithm until a CP-net has no features. We construct a preference ordering for A by ranking every outcome with $X = x_i$ as preferred to any outcome with $X = x_j$ if $x_i \succ x_j$ in CPT(X).

Algorithm 1 Linearization of a Partial Order induced by CP-net A

```
function LexO(A, O, Lin = [], o = None)    ▷ Where A is a CP-net, O is the O-legal order on A, Lin is the (initially empty) linearization computed by the function, and o is an outcome (initially none).
    if O = Null then
        Lin.append(o)
        return Lin
    end if
    v = pop(O)
    for value ∈ CPT_{A,o}(v) do
        temp = o + value (set value in outcome o)
        Lin = LexO(A, O, Lin, temp)
    end for
    return Lin
end function
```

In Algorithm 1, $CPT_{A,o}(v)$ returns the ordered values of feature v in CP-net A, given a partial assignment o to a subset of features. This linearization, $LexO(A, O)$, where A is a CP-net and O an O-legal order over the features of A, enforces that ordered pairs in the induced partial order are ordered the same in the linearization and that incomparable pairs are linearized using the cp-tables.

3 METRIC SPACES

A metric space is a pair (M, d), where M is a set of elements and d is a function (called distance or metric) $d : M \times M \to \mathbb{R}$, which satisfies the following properties:

(1) $d(A, B) \geq 0$;
(2) $d(A, B) = d(B, A)$;
(3) $d(A, B) \leq d(A, C) + d(C, B)$.
(4) $d(A, B) = 0$ if and only if $A = B$;

Throughout our work we assume that M is a set of CP-nets and we focus on defining distance functions between the elements of M. We assume that all CP-nets are acyclic and in minimal (non-degenerate) form, i.e., all arcs in the dependency graph have a real dependency expressed in the cp-statements, see the extended discussion by Allen et al. [2, 3]. In Definition 3.1 extend the Kendall τ (KT) distance [23] with a penalty parameter p defined for partial rankings by Fagin et al. [21] to the case of partial orders.

Definition 3.1. Given two CP-nets A and B inducing partial orders P and Q over the same unordered set of outcomes U:

$$KTD(A, B) = KT(P, Q) = \sum_{\forall i, j \in U, i \neq j} K^p_{i,j}(P, Q) \tag{1}$$

where i and j are two outcomes with $i \neq j$ (i.e., iterate over all unique pairs), we have:

(1) $K^p_{i,j}(P, Q) = 0$ if i, j are ordered in the same way or they are incomparable in both P and Q;
(2) $K^p_{i,j}(P, Q) = 1$ if i, j are ordered inversely in P and Q;
(3) $K^p_{i,j}(P, Q) = p$, $0.5 \leq p < 1$ if i, j are ordered in P (resp. Q) and incomparable in Q (resp. P).

KTD as defined in Definition 3.1 is a count, hence CP-nets with different numbers of features will have different possible maximum and minimum values. In order to make it scale invariant and thus comparable across CP-net pairs, we project it into $[0, 1]$ by normalizing: we divide the KTD value by the total number of pairs of outcomes which can increase the distance, i.e. the total number of pairs where outcomes are comparable in at least one CP-net.

In Definition 3.1 we must set $0.5 \leq p < 1$ to make $KTD(A, B)$ a distance function, indeed if $p < 0.5$ the distance does not satisfy the triangle inequality, as shown by Fagin et al. [21]. We also exclude $p = 1$ so that there is a penalty for two outcomes being considered incomparable in one and ordered in another CP-net. This allows us, assuming O-legality [24] of the CP-nets, to define for each CP-net a unique most distant CP-net.

An important question is the complexity of computing the distance between two CP-nets. We can extend a result from Santhanam et al. [36] to show that in general, the question is hard.

Proposition 3.2. *Given two CP-nets A and B deciding if $KTD(A, B) = 0$ cannot be computed in polynomial time unless $P = NP$.*

Proof. Santhanam et al. [36] show that it is NP-complete to verify equivalence for two CP-nets, i.e., deciding if two CP-nets induce the same ordering, can be reduced to the problem of checking if their KTD distance is 0. Hence, if we had a polynomial time algorithm for deciding if $KTD(A, B) = 0$ then we could decide the equivalence problem for acyclic CP-nets. □

We know from Boutilier et al. [7] that dominance testing for max-δ-connected CP-nets, i.e., CP-nets where the maximum number of paths between two features is polynomially bounded in the size of the CP-net is NP-complete. We know that O-legal, acyclic CP-nets are a class of max-δ-connected CP-nets because the O-legality constraint means that there are only a maximum of $n - 2$ paths between two nodes. Hence, it seems reasonable to conjecture that the same result holds also for O-legal CP-nets, though this question remains open.

3.1 Metric Spaces for O-Legal CP-nets

Due to the likely intractability of KTD we define a new distance for CP-nets which can be computed efficiently directly from the CP-nets without having to explicitly compute the induced partial orders. This new distance is defined as the Kendal Tau distance of the two $LexO$ linearization (Algorithm 1) of the partial orders.

Definition 3.3. Given two O-legal CP-nets A and B, with m features, we define:

$$O\text{-CPD}(A,B) = \frac{KT(LexO(A), LexO(B))}{2^{m-1}(2^m - 1)} \tag{2}$$

where $2^{m-1}(2^m - 1)$ represents the maximum KT distance between two total orders over 2^m outcomes.

Definition 3.3 requires a linearization but not necessarily the one provided in Algorithm 1. Observe that depending on the particular ordering O used, O-CPD can return a different distance value since the ensuing linearization may vary from ordering to ordering. We show that O-CPD is a distance over O-legal CP-nets.

Theorem 3.4. *Function* O-*CPD*(A, B) *is a metric.*

Proof. Properties 1-3 are directly derived from the fact that KTD is a distance function over total orders. Let us now focus on property 4. In our context, $A = B$ if and only if they induce the same partial order. Thus, if $A = B$ then O-CPD$(A, B) = 0$ since $LexO(A) = LexO(B)$. Let us now assume that $A \neq B$, i.e., A and B induce different partial orders. In principle, what could happen is that one partial order is a subset of the other. In such a case they would have the same $LexO$ linearizations and it would be the case that O-CPD$(A, B) = 0$, despite them being different. We need to show that this cannot be the case if A and B are O-legal. Let us first assume that A and B have the same dependency graph but that they differ in at least one ordering in one CP-table. It is easy to see that in such a case there is at least one pair of outcomes that are ordered in the opposite way in the two induced partial orders. Assume that A and B have a different dependency graph. Due to O-legality it must be that there is a least an edge which is present, say, in A and missing B. In this case by adding a non-redundant dependency we are reversing the order of at least two outcomes. □

We will now show how O-CPD(A, B) can be directly computed from CP-nets A and B, without having to compute the linearizations. The computation comprises of two steps. The first step, which we call, normalization, modifies A and B so that each feature will have the same set of parents in both CP-nets. This means that each feature will have, in both normalized CP-nets, a CP-table with exactly the same number of rows corresponding to the same assignment to its parents. The second step, broadly speaking, computes the contribution to the distance of each difference in these CP-table entries. We describe each step in turn.

Step 1: Normalization: Consider two CP-nets, A and B over m features $V = \{X_1, \dots, X_m\}$ each with binary domains. We assume the two CP-nets are O-legal with respect to a total order $O = X_1 < X_2 < \dots < X_{m-1} < X_m$. We note that O-legality implies that the X_i can only depend on a subset of $\{X_1, \dots, X_{i-1}\}$

Each feature X_i has a set of parents $Pa_A(X_i)$ (resp. $Pa_B(X_i)$) in A (resp. in B), and is annotated with a conditional preference table in each CP-net, denoted $CPT_A(X_i)$ and $CPT_B(X_i)$.

We note that, in general we will have that $Pa_A(X_i) \neq Pa_B(X_i)$. However, it is easy to extend the two CP-nets so that in both X_i will have the same set of parents $Pa_A(X_i) \cup Pa_B(X_i)$. This is done by adding redundant information to the CP-tables, which does not alter the induced ordering.

For example, let us consider $CPT_A(X_i)$, then we will add $2^q - 2^{Pa_A(X_i)}$ (where $q = Pa_A(X_i) \cup Pa_B(X_i)$) copies of each original row to $CPT_A(X_i)$, that is, one for each assignment to the features on which X_i depends in B but not in A. After this process is applied to all the features in both CP-nets, each feature will have the same parents in both CP-nets and its CP-tables will have the same number of rows in both CP-nets. We denote with A' and B' the resulting CP-nets. We note that normalization can be seen as the reverse process of CP-net reduction [4] which eliminates redundant dependencies in a CP-net.

Step 2: Distance Calculation: Given two normalized CP-nets A and B, let $diff(A, B)$ be the set of CP-table entries of B which are different in A and let $var(i) = j$ if CP-table entry i refers to feature X_j. Let $m = |V|$ and $flw(X)$ denote the number of features following X in order O. We define the two following quantities (expanded in Example 3.6):

$$nSwap(A, B) = \sum_{j \in diff(A,B)} 2^{flw(var(j)) + (m-1) - |Pa_B(var(j))|} \tag{3}$$

which counts the number of inversions that are caused by each different table entry and sums them up, and

$$maxSwap(A) = 2^{(m-1)} \cdot \sum_{X \in V} 2^{flw(X)} \tag{4}$$

which counts the number of total possible swaps, that is equal to having a CP-net that states the exact opposite of A. Observe that $maxSwap$ is only dependent on the O-legal order, since it counts the number of swaps to get a complete inversion of a linearization.

Theorem 3.5. *Given two normalized CP-nets A and B, we have:*

$$\text{O-}CPD(A, B) = \frac{nSwap(A, B)}{maxSwap(A)} \tag{5}$$

We provide an example that gives an intuition of how a difference in a CP-table entry affects the $LexO$ linearization.

Example 3.6. Consider a CP-net with three binary features, A, B, and C, with domains containing f and $\overline{f}$ if F is the name of the feature, and with the cp-statements as follows: $a \succ \overline{a}$, $b \succ \overline{b}$, $c \succ \overline{c}$. A linearization of the partial order induced by this CP-net can be obtained by imposing an order over the features, say Let feature ordering $O = A \succ B \succ C$. The $LexO(A)$ is as follows:

$$\overbrace{\overbrace{abc \succ ab\overline{c}}^{B1Zone} \succ \overbrace{a\overline{b}c \succ a\overline{b}\overline{c}}^{B2Zone}}^{A1Zone} \succ \overbrace{\overbrace{\overline{a}bc \succ \overline{a}b\overline{c}}^{B3zone} \succ \overbrace{\overline{a}\overline{b}c \succ \overline{a}\overline{b}\overline{c}}^{B4zone}}^{A2zone}$$

Now, consider changing only the cp-statement regarding A to $\overline{a} > a$. Then, the linearization of this new CP-net can be obtained by the previous one by swapping the first outcome in the *A1zone* with the first outcome in the *A2zone*, the second outcome in the *A1zone* with the second outcome in the *A2zone* and so on. Moreover, the number of swaps is directly dependent on the number of features that come after A in the total order.

From Theorem 3.5 we can see that $0 \leq O\text{-CPD}(A, B) \leq 1$, where m is the number of features. In particular:

- $O\text{-CPD}(A, B) = 0$ when the two CP-nets have the same dependency graph and cp-tables and so they are representing the same preferences;
- $O\text{-CPD}(A, B) = 1$ when the two CP-nets have the same dependency graph but cp-tables with reversed entries, so they are representing preferences that are opposite to each other.

Notice that features with different cp-statements in the representation give more value to the distance if they come first in the total order: the value decreases as the position in the total order increases. For instance it is easy to prove that if the cp-statement of the first feature in the total order differs, than $O\text{-CPD} \geq \frac{1}{2}$.

3.2 Relaxing O-Legality

In the real world, people can have preferences vary wildly and may not have the same topological ordering over the aspects that they consider important. Consequently, in some domains the assumption of O-legality may be too strong or unnatural. Therefore we define a metric which does not require O-legality.

This distance requires that the CP-nets are normalized using the process described in Section 3.1. This normalization procedure can create cycles in the set of CP-nets due to the induced dependencies in the graph of the features. However, this is irrelevant for our purposes as the process does not change the induced partial order. We merely need to normalize so that there are the same number of cp-entries in each CP-net.

Each difference in the CP-table corresponds to an inversion of an edge in the induced partial order, i.e. an inversion of preference between two outcomes. The distance counts this inversions and it also considers portions of the transitive closure by computing the number of pairs which are directly affected by these differences in the CP-tables.

Formally, the number of pairs of outcomes for which either the preference is inverted directly or inverted due to the transitive closure of the induced partial order can be computed as:

$$nInversion(A, B) = \sum_{j \in diff(A,B)} 2^{m-1-|Pa_B(var(j))|+|Succ_B(var(j))|} \tag{6}$$

The total number of possible inversions that can be directly counted in such a way is:

$$maxInversion(A, B) = \sum_{X \in V} 2^{m-1-|Pa_B(X)|+|Succ_B(X)|} \tag{7}$$

Definition 3.7. Given two normalized CP-nets A and B, with m features, we define:

$$I\text{-CPD}(A, B) = \frac{nInversion(A, B)}{maxInversion(A, B)}. \tag{8}$$

The function $I\text{-CPD}(A, B)$ is a distance over acyclic CP-nets that does not require the O-legality assumption. It is also interesting to notice that $I - CPD$ is independent from the ordering used to process the features and compute the distance. We can make the following statement whose proof is similar to that for Theorem 3.4 and we omit it for space.

THEOREM 3.8. *Function* I-$CPD(A, B)$ *is a metric.*

4 BOUNDING THE ERROR OF *O*-CPD

The reason for introducing O-CPD is to provide a distance over CP-nets which can be computed directly from their structures and which approximates their KTD.

To understand to what extent O-CPD can differ from KTD, let us consider two O-legal CP-nets, A and B, with induced partial orders P and Q, and two outcomes o and o'. From Definitions 3.1 and 3.3 it follows that:

- If o and o' are ordered in both P and Q then the pair will contribute in the same way, i.e., either with 0, if they are ordered in the same way, or 1, if they are ordered in the opposite way, to both $KTD(A, B)$ and $O\text{-CPD}(A, B)$.
- If o and o' are incomparable in both P and Q then the contribution of the pair to $KTD(A, B)$ is 0, while its contribution to $O\text{-CPD}(A, B)$ can be either 0 or 1 depending if the *LexO* linearization has linearized the pair in the same or opposite way in the two induced orderings.
- If o and o' are ordered in, say, P and incomparable in Q then the contribution of this pair to $KTD(A, B)$ is p while its contribution to $O\text{-CPD}(A, B)$ is either 0 or 1.

Summarizing, for each pair O-CPD can overestimate of at most 1 and under-estimate of at most p only if the pair is incomparable in at least one of the orderings. Thus, an absolute upperbound to the error that O-CPD makes can be estimated by counting the maximum number of incomparable pairs in an ordering induced by a CP-net. We will now compute this number.

Let us consider a separable CP-net S, that is, a CP-net over a set of m features V with binary domains and no dependencies between the features. Let P be the partial order induced by S over the set of outcomes U. A *chain* is a subset $U' \subseteq U$ such that for each $(x, y) \in U' \times U', x > y$ or $x < y$. We recall that the *height* of a partial order P, denoted $h(P)$, is the number of elements in the longest chain. We call *incomp*(P) the set of all the incomparable pairs of outcomes in P. In the following section we will call o_b the best outcome and o_w the worst outcome in P.

We start by observing that the height of a partial order induced by an acyclic CP-net corresponds to the length of the longest path from the best outcome to the worst outcome. This is a direct consequence of the fact that a CP-net induces a lattice.

PROPOSITION 4.1. *The height $h(P)$ of a partial order P induced by an acyclic CP-net coincides with the length of the longest path from o_b to o_w.*

We now observe that the number of incomparable pairs in a partial order is connected to its height. In fact, Mirsky's theorem states that the height of a partial order equals the cardinality of the minimum antichains partition that cover the partial order [33]. This result can be extended to partial orders induced by CP-nets.

Theorem 4.2. *Given two partial orders P and Q induced by two O-legal acyclic CP-nets defined over the same set of features V, if $h(P) > h(Q)$ then $incomp(P) < incomp(Q)$.*

Given Proposition 4.1 and Theorem 4.2 we can prove that separable CP-nets are indeed inducing partial orders with the maximal set of incomparables with respect to all other O-legal CP-nets.

Theorem 4.3. *A separable CP-net S induces a partial order P where the number of incomparable pairs of outcomes is maximal with respect of all the possible acyclic O-legal CP-nets over the same set of features.*

We provide the intuition behind the proof. Let's start by computing the partial order induced by the CP-net, starting from the best outcome o_b. We can now build the next level of outcomes by changing just one assignment for each feature, let's call $var(o_{i,j})$ the subset of features for which we change the value in the $j-th$ outcome of the $i-th$ level with respect to o_b. We get a subset of outcomes that differ for just one value from o_b. In the induced partial order all these outcomes are worst than o_b by definition. The cardinality of this subset is $\binom{n}{1}$. For each outcome we can now compute the subset of outcomes of the next level by changing the assignment of just one feature except the ones in $var(o_{i,j})$. Each outcome in level $i+1$ derived from $o_{i,j}$ is worst than it. For each level i the number of outcomes is $\binom{n}{i}$. In such a way there does not exist an outcome o' in level i' which is better than another outcome o'' in level i'', with $i' > i''$. Roughly speaking, we have shown that level partitioning is minimal and any other CP-net structure leads to an increment of height of the induced partial order. But due to Theorem 4.2, a completely separable CP-net has the maximal number of incomparable pairs with respect to any other CP-net.

This reasoning allows us to make the following statement.

Theorem 4.4. *The total number of incomparable pairs in any completely separable CP-net is:*

$$\sum_{i=1}^{m-1}\binom{m}{i}\cdot(\frac{1}{2}(\binom{m}{i}-1)+\sum_{l=1}^{i}\binom{i}{l}\sum_{j=l+1}^{m-1}\binom{m-1}{j}) \tag{9}$$

In fact, starting from the best outcome we can flip the value of a single feature to have a new outcome which is directly comparable with the best outcome. This can be done $\binom{m}{i}$ times (with $i=1$) to have all the possible outcomes that are directly comparable with the best outcome in the induced partial order. All the outcomes computed in such a way are incomparable with respect to each other since the CP-net is separable and since each outcome differs on two values from any other outcome at the same level (let's denote a level with the value of index i). So the number of incomparable pairs is $\binom{m}{i}\cdot(\frac{1}{2}(\binom{m}{i}-1)$. Iterating this computation by increasing the index i we have the number of incomparable pairs for each level. We now need to compute the number of incomparable pairs due to the transitive closure. At each level, the index i also represents the number of features with a different values with respect to the best outcome, we call $F \subset V$ the subset of such features. Let's consider two outcomes o at level i and o' at level j, with $j = i+1$. The two outcomes are incomparable if at least one feature in F and at least one more features not in F have different values. The proof is straightforward, if all the features in F for o and o' have the same values, then a single feature is changing from o to o' that makes the two outcomes comparable. We can choose $\binom{i}{l}$ different combinations for the features in F and $\binom{m-1}{j}$ of different combinations for features not in F. Iterating this process by increasing the index j allows us to compute the number of outcomes which are incomparable to o.

The number of incomparable pairs can be used to bound the error of O-CPD. It is easy to see that each non-fake arc added to the dependency graph of a CP-net reverses at least one arc in the induced partial order. This reduces of at least 1 the number of incomparable pairs, since we connect two outcomes of the same level in the induced partial order.

5 DISTANCE RATIONALIZABILITY

In this section we describe how a notion of distance between CP-nets can be used to inform preference aggregation. Multi-agent systems face the problem of finding a common outcome, which satisfies either a majority or all of a set of agents. In general, research in preference aggregation studies rules that synthesize the preferences of the agents into one or more selected outcome and characterizes the rules in terms of the properties they satisfy [11, 35].

In this context, Distance Rationalizability (DR) is a framework used for studying voting rules [8, 19, 20, 32]: given a consensus notion, a distance, and a voting rule on preference profiles, the voting rule is said to be distance rationalizable if it elects the same winner as the closest profile in the consensus class. Intuitively, this is a way to "rationalize" (justify) the behavior of the voting rule by proving that it returns the same winner of a profile in which voters agree and that is the most similar to the one given in input with respect to a particular metric.

Most of the results on distance rationalizability, have studied the case in which voters express their preferences via total orders [20]. However, when voters express their preference as CP-nets we face the problem of aggregating partial orders, which has been shown to be a hard problem [10]. This hardness can be addressed via Sequential Voting (SV) [25, 26]. Given a profile of O-legal CP-nets over the same n features, a sequential voting rule is a tuple of n voting rules. The sequential procedure applies the voting rules to the features following order O. In particular, in each step, a profile of total orders over the values of the feature under consideration is obtained from the CP-tables considering the row corresponding to the values of the parent features elected in previous steps.

We show that distance function O-CPD can be used to rationalize sequential plurality voting under two common consensus classes. We recall that in plurality voting, ballots consist of the single most preferred candidate and the candidate with the most votes win and tie breaking rules may need to be applied. In our context all features are binary and under sequential voting we select one value for each feature, hence Plurality is the natural choice to consider. Under non-sequential or non-binary settings other rules may be more appropriate. We will now show that distance function O-CPD can be used to rationalize sequential plurality voting on CP-nets with binary features under two common consensus classes. To the best of our knowledge, these are the first results which extends DR to voting with preferences represented compactly.

We extend the notion of consensus classes to profiles of CP-nets:

- **Strong Unanimity** S**:** consists of profiles of CP-nets where all individuals have the same CP-net, i.e. all have the same partial order over the outcomes;
- **Unanimity** U**:** consists of profiles of CP-nets where all individuals have the same best outcome, i.e. they all have a partial order with the same undominated outcome at the top; and
- **Majority** M**:** consists of profiles of CP-nets where there is a majority of CP-nets that have the same best outcome, i.e. there is a majority of partial orders that have the same best outcome at the top.

Plurality voting is distance rationalizable with respect to the consensus classes defined on total orders homologous to S and U and swap distance [18, 32].

The following definition of distance rationalizability is an adaptation to CP-nets of the one given in [20]:

Definition 5.1. A Sequential Voting rule (SV) is said to be *distance-rationalizable* with respect to a consensus class $X \in \{S, U, M\}$ if there is a distance d over the induced orders such that for each profile V of CP-nets, which is O-legal w.r.t. some order O, an outcome o is the winner under SV applied to V if and only if it is a winner in a nearest (with respect to d) election in X.

Informally, the distance d over orders is used to define a distance between profiles (that is, elections) by taking the sum of the distances between pairs of corresponding orders in the two elections. So the nearest election to V is the election in the particular consensus class X with smallest distance from V. Given a distance d over CP-nets, and two profiles of n CP-nets $E = (u_1, \ldots, u_n)$ and $E' = (v_1, \ldots, v_n)$ over the same set of features, we define a distance over profiles as the sum of the distances between corresponding pairs of CP-nets, $d'(E, E') = \sum_{i=1}^{n} d(u_i, v_i)$.

Theorem 5.2. *Sequential plurality voting is* O-CPD *DR with respect to the strong unanimity consensus class* S.

Intuitively, given a profile of CP-nets E, the profile E' in the consensus class S which minimizes O-CPD is the one where all the CP-nets have the same topology. In detail, for each feature X_i' the set of parents is the union of the parents of X_i in all the CP-nets in E. Moreover, the CP-tables associate to each joint assignment of the parent features the order appearing in the majority of the corresponding entries in the CP-nets.

Theorem 5.3. *Sequential plurality voting is* O-*CPD DR with respect to the unanimity consensus class* U.

In this case, in the consensus profile the CP-nets maintain their original topology. Let assume that the winner obtained by the sequential approach is outcome o. Then, to obtain a profile in U, only the rows of the CP-tables corresponding to o are changed so to put its assignment first.

Theorem 5.4. *Sequential plurality voting is* O-*CPD DR with respect to the majority consensus class* M.

This is true by design, as the nearest profile in M is E itself.

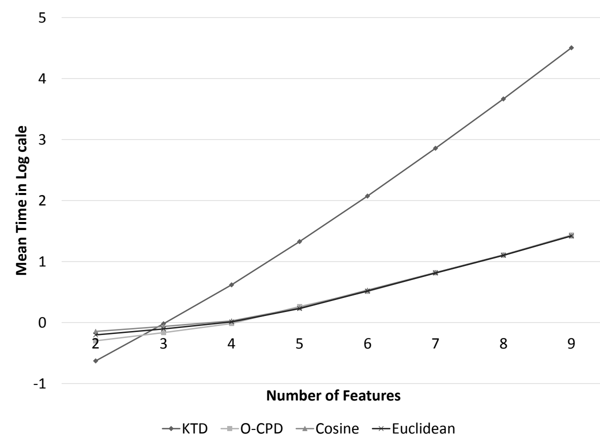

Figure 1: Mean time (log scale) to compute distance metrics for CP-nets with various numbers of features. For a pair of CP-nets with 9 features it takes ≈32929ms to compute KTD and only 27ms for O**-CPD or** I**-CPD.**

6 EMPIRICAL ANALYSIS

To support our theoretical results we run a set of experiments to see how our distances measures behave in practice. Due to the lack of real world data [30, 31], we generate CP-nets uniformly at random using the software described by Allen et al. [2, 3]. During this phase we generate CP-nets over n features with binary domains and $2 \leq n \leq 9$ parents. For each value of n we generate 1000 CP-nets using the default software settings: for $2 \leq n \leq 6$ the maximum number of parents is $n - 1$, while for $7 \leq n \leq 9$ the maximum number of parents is 5. We use these generated CP-nets to test different properties of the distance function.

Experiments are developed in Python and were run on a cluster with 2 Intel E5-2670 CPUS running at 2.60GHz, 128GB of memory, and a Tesla K20m GPU. For all experiments we set $p = 0.5$ when computing KTD. To compare our metrics we built a simple vector representation of a CP-net. To do this we first, given two CP-nets A and B, where A is the referee, normalize the two CP-nets to have the same number of cp-entries. Each cell of the vector representing A is equal to 1. Each i-th cell in the vector representing B is equal to 1 if the correspondent cp-entry in the two CP-nets are equal, -1 otherwise. Using this vector representation we can compute both the cosine similarity between the vectors and the Euclidean distance [5]. Using these two well known metrics as distance functions allows us to compare the performance of O-CPD with simple baselines.

Fig. 1 shows the average time required to compute KTD, O-CPD, cosine similarity, and the Euclidean distance. While KTD grows exponentially, the other mean times are all similar to each other. This suggests that simple distance functions have good performance on this combinatorial domain. Clearly, for small values of n, the running time for O-CPD is higher than that of KTD due to the normalization processing time. However, the mean time for KTD increases exponentially and is almost 4 orders of magnitude greater than the mean time of O-CPD when $n \geq 3$.

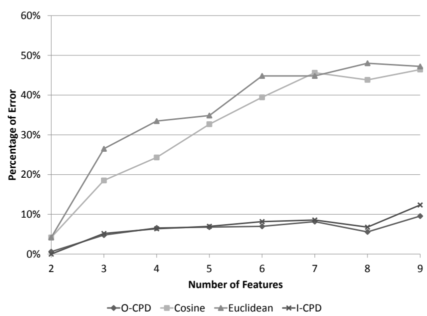

Figure 2: Percentage of incorrectly classified O-legal CP-nets versus KTD for the various distance metrics.

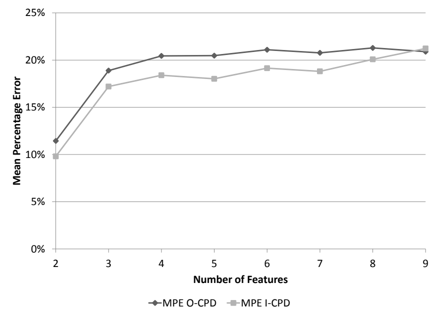

Figure 3: Mean percentage error of O-CPD and I-CPD with respect to the true value of KTD for O-legal CP-nets.

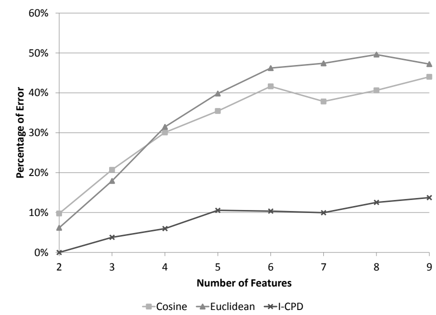

Figure 4: Percentage of incorrectly classified non-O-legal CP-nets versus KTD for the various distance metrics.

In addition to running time, we also checked how many times O-CPD, I-CPD, cosine similarity and the Euclidean distance are incorrect w.r.t. KTD. Formally, given a triple of CP-nets (A, B, C) over the same set of features we compute $kt1 = KTD(A, B)$, $kt2 = KTD(A, C)$, $l1 = O\text{-CPD}(A, B)$ and $l2 = O\text{-CPD}(A, C)$ (respectively for the other metrics) and count how many times $kt1 >= kt2$ but $l1 < l2$ or vice-versa. In other words, we count how many times B is closer to A with respect to C according to KTD, but the various distance metrics state the opposite.

Fig. 2 shows the percentage of incorrectly classified cases as we sweep n. We observe that our metrics strictly dominate the performance of the vector representations and this result is statistically significant. The error rates for O-CPD and I-CPD are relatively stable as we increase n, at about 9% and 13%, respectively. This is interesting because with a high value of n the number of incomparable pairs in the induced partial order increases: incomparable pairs are the main reason for the over-estimation and the under-estimation of KTD. Hence, O-CPD and I-CPD show good resilience to increasing numbers of incomparable pairs.

Looking at O-CPD and I-CPD more closely, we compute the Mean Percentage Error (MPE) in Fig. 3. The MPE is a scale invariant measure that gives an idea of how much our metrics vary from KTD. For each value of n we compute: $MPE = \frac{100}{t} \sum_{i=1}^{t} \frac{ktd_i - cpd_i}{ktd_i}$. Where t is the number of samples drawn randomly, ktd_i is the value of KTD for the i-th sample and cpd_i is the value of O-CPD or I-CPD for the i-th sample. Looking closely at Figure 3 we see that after $n \geq 3$ the error stays relatively constant.

Turning to I-CPD, we ran the same set of experiments on CP-nets which were not O-legal. The results of these experiments are show in Figures 4 and 5. We observe that I-CPD performs extremely well even in cases where there are a high number of incomparable pairs. In Figure 4 we see I-CPD still out preforms the other metrics on non-O-legal CP-nets and that the error is relatively stable. In Figure 5 we see that the MPE for I-CPD is *smaller* than for O-legal CP-nets and it actually stabilizes around 7%. In fact, as the number of features increase, we see a drop in the error rate for I-CPD, indicating that it works very well across a large number of cases.

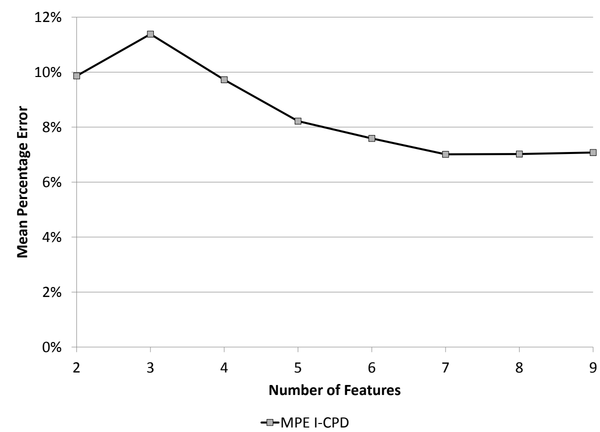

Figure 5: Mean percentage error of I-CPD with respect to the true value of KTD for non-O-legal CP-nets.

7 CONCLUSIONS

In this paper we defined two novel notions of distance between CP-nets. This is, to the best of our knowledge, the first attempt to define a tractable notion of distance between CP-nets. We give theoretical bounds and an experimental evaluation, showing that both are efficient and accurate. Using these metrics we extend the concept of Distance Rationalizability to partial orders and hence to Sequential Voting Rules defined for CP-nets, showing that our new distance can be used to rationalize the sequential plurality voting rule for CP-nets. There are a number of interesting extensions to consider for future work including tighter bounds and extending both metrics to work when CP-nets have different features.

REFERENCES

[1] T.E. Allen, M. Chen, J. Goldsmith, N. Mattei, A. Popova, M. Regenwetter, F. Rossi, and C. Zwilling. 2015. Beyond Theory and Data in Preference Modeling: Bringing Humans into the Loop. In *Proceedings of the4thInternational Conference on Algorithmic Decision Theory (ADT)*.

[2] T.E. Allen, J. Goldsmith, H.E. Justice, N. Mattei, and K. Raines. 2016. Generating CP-nets Uniformly at Random. In *Proceedings of the30thAAAI Conference on Artificial Intelligence (AAAI)*.

[3] T. E. Allen, J. Goldsmith, H. E. Justice, N. Mattei, and K. Raines. 2017. Uniform Random Generation and Dominance Testing for CP-Nets. *Journal of Artificial Intelligence Research* 59 (2017), 771–813.

[4] Krzysztof R. Apt, Francesca Rossi, and Kristen Brent Venable. 2008. Comparing the notions of optimality in CP-nets, strategic games and soft constraints. *Ann. Math. Artif. Intell.* 52, 1 (2008), 25–54.

[5] C. M. Bishop. 2006. *Pattern Recognition and Machine Learning*. Springer.

[6] Jean-François Bonnefon, Azim Shariff, and Iyad Rahwan. 2016. The social dilemma of autonomous vehicles. *Science* 352, 6293 (2016), 1573–1576.

[7] Craig Boutilier, Ronen Brafman, Carmel Domshlak, Holger Hoos, and David Poole. 2004. CP-nets: A Tool for Representing and Reasoning with Conditional Ceteris Paribus Preference Statements. *Journal of Artificial Intelligence Research* 21 (2004), 135–191.

[8] Craig Boutilier and Ariel D. Procaccia. 2012. A Dynamic Rationalization of Distance Rationalizability.. In *AAAI*. AAAI Press.

[9] Ronen I. Brafman and Yannis Dimopoulos. 2004. Extended Semantics and Optimization Algorithms for CP-Networks. *Computational Intelligence* 20, 2 (2004), 218–245.

[10] Franz-Josef Brandenburg, Andreas Gleißner, and Andreas Hofmeier. 2013. Comparing and Aggregating Partial Orders with Kendall tau Distances. *Discrete Math., Alg. and Appl.* 5, 2 (2013).

[11] F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia (Eds.). 2016. *Handbook of Computational Social Choice*. Cambridge University Press.

[12] Yann Chevaleyre, Frédéric Koriche, Jérôme Lang, Jérôme Mengin, and Bruno Zanuttini. 2011. Learning ordinal preferences on multiattribute domains: The case of CP-nets. In *Preference Learning*. Springer, 273–296.

[13] Vincent Conitzer, Jérôme Lang, and Lirong Xia. 2011. Hypercubewise preference aggregation in multi-issue domains. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI)*. 158–163.

[14] David Copp. 2005. *The Oxford Handbook of Ethical Theory*. Oxford University Press.

[15] C. Cornelio, J. Goldsmith, N. Mattei, F. Rossi, and K.B. Venable. 2013. Updates and Uncertainty in CP-nets. In *Proceedings of the 26th Australasian Joint Conference on Artificial Intelligence (AUSAI)*.

[16] C. Cornelio, U. Grandi, J. Goldsmith, N. Mattei, F. Rossi, and K.B. Venable. 2015. Reasoning with PCP-nets in a Multi-Agent Context. In *Proceedings of the 14th International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*.

[17] C. Domshlak, E. Hüllermeier, S. Kaci, and H. Prade. 2011. Preferences in AI: An overview. *Artificial Intelligence Journal* 175, 7 (2011), 1037–1052.

[18] E. Elkind, P. Faliszewski, and A. Slinko. 2010. On the role of distances in defining voting rules. In *Proceedings of the9thInternational Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*. 375–382.

[19] Edith Elkind, Piotr Faliszewski, and Arkadii Slinko. 2015. Distance rationalization of voting rules. *Social Choice and Welfare* 45, 2 (2015), 345–377.

[20] Edith Elkind, Piotr Faliszewski, and Arkadii M. Slinko. 2010. Good Rationalizations of Voting Rules.. In *AAAI*, Maria Fox and David Poole (Eds.). AAAI Press.

[21] Ronald Fagin, Ravi Kumar, Mohammad Mahdian, D. Sivakumar, and Erik Vee. 2006. Comparing Partial Rankings. *SIAM J. Discret. Math.* 20, 3 (March 2006), 628–648. https://doi.org/10.1137/05063088X

[22] J. Goldsmith, J. Lang, M. Truszczyński, and N. Wilson. 2008. The computational complexity of dominance and consistency in CP-nets. *Journal of Artificial Intelligence Research* 33, 1 (2008), 403–432.

[23] M. G. Kendall. 1938. A new measure of rank correlation. *Biometrika* 30, 1/2 (1938), 81–93.

[24] Jerome Lang and Lirong Xia. 2009. Sequential composition of voting rules in multi-issue domains. *Mathematical Social Sciences* 57, 3 (2009), 304–324.

[25] J. Lang and L. Xia. 2009. Sequential composition of voting rules in multi-issue domains. *Mathematical Social Sciences* 57, 3 (2009), 304–324.

[26] Minyi Li, Quoc Bao Vo, and Ryszard Kowalczyk. 2011. Majority-rule-based preference aggregation on multi-attribute domains with CP-nets.. In *AAMAS*, Liz Sonenberg, Peter Stone, Kagan Tumer, and Pinar Yolum (Eds.). IFAAMAS, 659–666.

[27] A. Loreggia, N. Mattei, F. Rossi, and K. B. Venable. 2018. Preferences and Ethical Principles in Decision Making. In *Proceedings of the 1st AAAI/ACM Conference on AI, Ethics, and Society (AIES)*.

[28] A. Loreggia, N. Mattei, F. Rossi, and K. B. Venable. 2018. Value Alignment Via Tractable Preference Distance. In *Artificial Intelligence Safety and Security*, R. V. Yampolskiy (Ed.). CRC Press.

[29] N. Mattei, M. S. Pini, F. Rossi, and K. B. Venable. 2013. Bribery in Voting with CP-nets. *Annals of Mathematics and Artificial Intelligence* 68, 1–3 (2013), 135–160.

[30] N. Mattei and T. Walsh. 2013. PrefLib: A Library for Preferences, HTTP://WWW.PREFLIB.ORG. In *Proceedings of the 3rd International Conference on Algorithmic Decision Theory (ADT)*.

[31] N. Mattei and T. Walsh. 2017. A PrefLib.Org Retrospective: Lessons Learned and New Directions. In *Trends in Computational Social Choice*, U. Endriss (Ed.). AI Access Foundation, Chapter 15, 289–309.

[32] Tommi Meskanen and Hannu Nurmi. 2008. *Closeness Counts in Social Choice*. Springer Berlin Heidelberg, Berlin, Heidelberg, 289–306. https://doi.org/10.1007/978-3-540-73382-9_15

[33] L. Mirsky. 1971. A Dual of Dilworth's Decomposition Theorem. *The American Mathematical Monthly* 78, 8 (1971), 876–877.

[34] G. Pigozzi, A. Tsoukiàs, and P. Viappiani. 2015. Preferences in Artificial Intelligence. *Annals of Mathematics and Artificial Intelligence* 77 (2015), 361–401.

[35] F. Rossi, K.B. Venable, and T. Walsh. 2011. *A Short Introduction to Preferences: Between Artificial Intelligence and Social Choice*. Morgan and Claypool. 1–102 pages.

[36] Ganesh Ram Santhanam, Samik Basu, and Vasant Honavar. 2013. Verifying Preferential Equivalence and Subsumption via Model Checking. In *Algorithmic Decision Theory - Third International Conference, ADT 2013, Bruxelles, Belgium, November 12-14, 2013, Proceedings (Lecture Notes in Computer Science)*, Patrice Perny, Marc Pirlot, and Alexis Tsoukiàs (Eds.), Vol. 8176. Springer, 324–335.

[37] Amartya Sen. 1974. *Choice, Ordering and Morality*. Blackwell, Oxford.

[38] Judith Jarvis Thomson. 1985. The Trolley Problem. *The Yale Law Journal* 94, 6 (1985), 1395–1415.