

Reasoning with PCP-Nets

Cristina Cornelio

Samsung AI, Cambridge, United Kingdom

C.CORNELIO@SAMSUNG.COM

Judy Goldsmith

University of Kentucky, Lexington, KY, USA

GOLDSMIT@CS.UKY.EDU

Umberto Grandi

Institut de Recherche en Informatique de Toulouse (IRIT)

University of Toulouse, France

UMBERTO.GRANDI@IRIT.FR

Nicholas Mattei

Department of Computer Science

Tulane University, New Orleans, LA, USA

NSMATTEI@TULANE.EDU

Francesca Rossi

IBM Research, T.J. Watson Research Center

Yorktown Heights, New York, USA

FRANCESCA.ROSSI2@IBM.COM

K. Brent Venable

IHMC and University of West Florida, Florida, USA

KVENABLE@TULANE.EDU

Abstract

We introduce PCP-nets, a formalism to model qualitative conditional preferences with probabilistic uncertainty. PCP-nets generalise CP-nets by allowing for uncertainty over the preference orderings. We define and study both optimality and dominance queries in PCP-nets, and we propose a tractable approximation of dominance which we show to be very accurate in our experimental setting. Since PCP-nets can be seen as a way to model a collection of weighted CP-nets, we also explore the use of PCP-nets in a multi-agent context, where individual agents submit CP-nets which are then aggregated into a single PCP-net. We consider various ways to perform such aggregation and we compare them via two notions of scores, based on well known voting theory concepts. Experimental results allow us to identify the aggregation method that better represents the given set of CP-nets and the most efficient dominance procedure to be used in the multi-agent context.

1. Introduction

Preferences are ubiquitous in our everyday life; whenever we make a decision we exploit our preferences over the candidate options. Moreover, if the decision is shared with others, we need to find ways to reconcile the possibly conflicting preferences of all the involved decision makers (agents). No area of our lives is free from explicit or implicit preference modeling, reasoning, and aggregation. For this reason, the study of how to model and reason with preference plays such an important role in automated decision making, multi-agent systems and artificial intelligence broadly defined, and many comprehensive surveys exist detailing this work, for example, Rossi, Venable, and Walsh (2011), Fürnkranz and Hüllermeier (2010), Brafman and Domshlak (2009), Chevaleyre, Endriss, Lang, and Maudet (2008), Pigozzi, Tsoukis, and Viappiani (2015) and Amor, Dubois, Gouider, and Prade (2016).

For many decision tasks that occur in, for example, e-commerce, web search, retailing, and personalized medicine, the number of options can outstrip an agent’s ability to view and express opinions over all the available choices (Adomavicius & Kwon, 2015). Hence, it is desirable in these instances to have an efficient formalism to model and reason with complex, interdependent preferences that are specified over decision features, rather than entire decisions. Consider for example expressing preferences over all cars vs. over car features such as make, model, engine, etc. The set of all cars has a combinatorial structure and grows very fast over the features, but preferences can and often are expressed over features, for example, *I prefer red cars to blue cars*. A number of compact preference representation languages have been developed in the literature to tackle the computational challenges arising from these scenarios; see the work of Amor et al. (2016) for a survey of compact graphical models. We mention specifically conditional preference structures (CP-nets) (Boutilier et al., 2004), soft constraints (Bistarelli et al., 1997; Rossi et al., 2011), and GAI-nets (Gonzales & Perny, 2004) as they have been used widely across computer science and other disciplines and possess complementary properties in terms of expressivity and reasoning efficiency.

Motivation and Semantics. In this paper we formalize and provide reasoning algorithms for *Probabilistic Conditional Preference networks (PCP-nets)*. PCP-nets are a generalization of CP-nets (Boutilier et al., 2004) that allow for modeling qualitative conditional preference statements with probabilistic uncertainty (Cornelio et al., 2013; Bigot et al., 2013; Cornelio et al., 2014, 2015). We decided to generalize CP-nets since CP-nets are arguably quite natural and intuitive to use in many circumstances (Chevalere et al., 2011; Allen et al., 2015), and can be used in many applications such as recommendation engines (Pu et al., 2011), including recommendation systems over features (Adomavicius & Kwon, 2015) or requiring aggregation (Beliakov, Calvo, & James, 2015); preference reasoning under constraints and/or ethical principles (Loreggia et al., 2018b, 2018a, 2019; Rossi & Mattei, 2019), aggregating the preferences of multiple agents (Lang & Xia, 2009; Rossi et al., 2004; Xia et al., 2011), and allow for expressing complex preferences over features of a decision. In addition, there is experimental evidence that qualitative preferences may more accurately reflect humans’ preferences in uncertain information settings (Roth & Kagel, 1995; Allen et al., 2015).

Often in real life the information is not complete, or not precise, or contains some level of uncertainty. In preference settings, an agent may be unsure about his preferences over certain items, or there could be noise due to lack of precision in the elicitation phase or in sensor collection, for example, a measurement error from remote sensors, or the uncertainty could come from aggregating the preferences of multiple agents. While CP-nets specify a preference ordering over the values of each feature of the considered objects, in a PCP-net we are given a probability distribution over all possible preference orderings of the features’ domain. For the car example in a CP-net we would state our preference ordering over the two colors (for example, that we prefer *white* to *red*), while in a PCP-nets we would say that we prefer *white* to *red* with a certain probability p , and *red* to *white* with probability $1 - p$. For example this probability can express the percentage of the time that we encounter a particular comparison: we may very much like Italian food (90% of the time) but occasionally (10% of the time) we prefer burger with fries. Another example is when we are reporting the preferences of someone else of which we are not certain.

It is interesting to think through the semantics of what the uncertainty and probabilities *mean*. If I prefer *red* to *white* with probability 0.8 how does one interpret this? It could be that it represents a strength of preference, though there are other formalisms that deal with this interpretation directly

(Boutilier, Bacchus, & Brafman, 2001). It could be that the context is not well specified, and that in 0.8 of the cases I observe $red > white$ while in 0.2 of the cases I observe the reverse; this is the model used by some psychologists to resolve intransitive observations in repeated two alternative forced choice experiments (Regenwetter, Dana, & Davis-Stober, 2010; Regenwetter & Davis-Stober, 2012; Regenwetter, Dana, & Davis-Stober, 2011; Marden, 1995). Or it could be that the PCP-net represents the aggregation of preferences over a large population (considering the agents independent from each other) that prefers $red > white$ 80% of the time; the idea that uncertain, qualitative information may come from multiple agents has been popular in the recommendation systems area (Price & Messinger, 2005; Faltings, Torrens, & Pu, 2004; Beliakov et al., 2015). Surprisingly, the same mathematics of PCP-net formalism works for *all of these interpretations*.

Summary of Results. Given a PCP-net, it is then interesting to study what it means to find the most preferred object and to respond to dominance queries involving two objects (determining if one outcome is more preferred than another according to the outcome ordering). For optimality, we notice that a PCP-net defines a probability distribution over a collection of *induced* CP-nets, that is, all those CP-nets that can be obtained from the PCP-net by choosing an ordering from each probability distribution over orderings in the PCP-net, and we use this observation to define two notions of optimality: the most probable optimal outcome and the optimal outcome of the most probable induced CP-net. Finding both optimal objects is computationally difficult in general, but it becomes easy if we impose some restriction on the connectivity of the dependency structure. For dominance queries, of the form “Is o_1 more preferred than o_2 ?”, the answer is a probability value, that is, the probability that o_1 is more preferred than o_2 , which by definition is the sum of the probabilities of those induced CP-nets where o_1 is preferred to o_2 . Finding such a value is computationally difficult, as dominance queries are already difficult in CP-nets. However, we propose and evaluate a tractable approximation method that returns an interval in which the exact value is positioned.

We then turn our attention to the use of PCP-nets in a multi-agent environment. This is a natural setting for PCP-nets, since, as noted above, a PCP-net models a collection of CP-nets. Thus we may think of individual agents specifying a CP-net that models their preferences, and a PCP-net that uses probabilities to reconcile the possibly conflicting preferences given by the individual CP-nets. This provides a formalism to perform collective reasoning tasks such as answering dominance queries, in addition to finding the most preferred outcome. Moreover, we can store or communicate between agents a single, compact structure instead of a possibly large collection of CP-nets. In contrast, previous efforts to tackle collective decision tasks over collections of CP-nets have primarily focused on the question of optimality (Li, Vo, & Kowalczyk, 2011) or dominance (Santhanam, Basu, & Honavar, 2010) and have not dealt with uncertainty. Generally, a voting method is defined in order to determine the most preferred alternative and these methods usually work at the level of the individual features in a sequential fashion (Lang & Xia, 2009; Xia et al., 2011; Li, Vo, & Kowalczyk, 2010; Li et al., 2011; Xia, Conitzer, & Lang, 2008; Maran, Maudet, Pini, Rossi, & Venable, 2013; Mattei, Pini, Rossi, & Venable, 2013, 2012). We propose various ways to obtain a PCP-net out of the given collection of CP-nets and then compare both optimality and dominance tasks theoretically and experimentally by considering the satisfaction level of the contributing CP-nets. For collective optimality, we prove that our best method gives the same result as the sequential voting procedure proposed by Lang and Xia (2009). For collective dominance we provide an approximation that is tractable in the case of CP-nets for which the dependency structure is a directed poly-tree or for-

est, that is, *polytree-structured* CP-nets (a polytree is a directed graph whose underlying undirected graph is a tree).

Extensions and Related Work. Since PCP-nets were first introduced in the work by Bigot et al. (2013) and Cornelio et al. (2013), there have been several generalizations and additional studies. The question of using CP-nets and PCP-nets for group decision making was further investigated in the work of Sikdar, Adali, and Xia (2017), closing several open questions. For example, in the work of Bigot, Mengin, and Zanuttini (2014) they show that PCP-nets can be learnt in both on-line and offline settings starting from a sequence of optimal items, by leveraging structure learning techniques for Bayesian Networks. Also, Fidha and Amor (2015) introduce *Probabilistic Weighted CP-nets* which allow for both uncertainty and differential preference weights, at the cost of a higher computational complexity. Weights are also added to CP-nets in the work of Wang, Zhang, Sun, Song, Guo, and Zhou (2012), allowing users to express more precisely the relative strength of preferences between two attributes. In the work of Lukasiewicz, Martinez, and Simari (2014) the notion of probabilistic preference logic networks are used to capture a more general notion of preferences than those modelled by CP-nets, again at the cost of higher computational complexity. Moreover, similar to our probabilistic extension of CP-nets, a probabilistic generalization of TCP-nets, which is an extension of CP-nets that represents the notion of relative importance and conditional relative importance between features (Brafman & Domshlak, 2002), has been recently proposed by Ahmed and Mouhoub (2018a). This work exploit directly our definition of G-net (see Definition 12) to compute the *most probable compatible TCP-net* and a generalization of the notion of Opt-Network (see Definition 13) to compute the most probable optimal outcome.

Organization. In Section 2 we introduce the main notions used throughout the paper including a background on CP-nets, Bayesian Networks, and voting. In Section 3 we formally define PCP-nets, while in Section 4 and 5 we address optimality and dominance tasks at the level of a single agent’s preferences (single network) and provide empirical experiments for our algorithms. In Section 6 we move to the multi-agent setting, showing how to compute optimality and dominance when we have a collection of preferences (networks) again using both theoretical and empirical tools. Finally, in Section 7 we summarize the contributions of the paper and we discuss possible lines for future work.

2. Background and Preliminaries

In this paper we will reduce several problems regarding PCP-nets to problems on Bayesian Networks. Thus, in the following section, in addition to providing the main notions on CP-nets, voting theory, and CP-net aggregation, we also discuss Bayesian Networks.

2.1 Bayesian Networks

Bayesian networks (BNs) are a compact framework for representing and reasoning with uncertain knowledge (Pearl, 1988). A BN is a directed graph where each node corresponds to a random variable; the set of nodes is denoted by V ; a set of directed edges connects pairs of nodes, that is, if there is an edge from node X to node Y , X is said to be a *parent* of Y ; the graph is a directed acyclic graph (DAG); each node X_i has a conditional probability distribution $\mathbb{P}(X_i|Pa(X_i))$ that quantifies the effect of the parents on the node X_i . If the nodes are discrete variables, each X_i has a *conditional*

probability table (CPT) that contains the conditional probability distribution, $\mathbb{P}(X_i|Pa(X_i))$. Each CPT row must therefore have probabilities that sum to 1.

Inference in a BN corresponds to calculating $\mathbb{P}(X|E)$ where both X and E are sets of variables of the BN, or to finding the most probable assignment for X given E . A given assignment to the variables in E is called *evidence*. There are three standard inference tasks in BNs:

- *belief updating*, which is finding the probability of an assignment of a variable or set of variables, possibly given evidence;
- finding the *most probable explanation (MPE)* means finding the most probable assignment for all variables that are not given as evidence; and
- finding the *maximum a-posteriori (MAP)* means finding the most probable assignment for some subset of variables (possibly given evidence).

These inference tasks are computationally hard in general. However, they can be solved in polynomial time if we impose restrictions on the topology of the BNs such as bounding the *induced width*, see the work of D’Ambrosio (1999) or Dechter (1999) for details. We recall that the *width of a graph* is the maximum width among all nodes in the graph, where the *width of a node X* in a graph (given an ordering of the nodes) is the number of nodes preceding X in the ordering that are connected to X . The **induced width** of a graph is the width of its corresponding induced graph (given an ordering of the variables) that is obtained by recursively connecting earlier neighbors of each node following the given ordering (D’Ambrosio, 1999).

Given an ordering of the variables of a BN, these tractable algorithms have a number of steps linear in the number of variables, and each step is exponential in the number of variables preceding the current one in the ordering and connected to it in the BN graph. The largest of these numbers is the induced width of the graph of the BN. Different variable orderings give steps with different complexity. Finding a good variable ordering is a difficult problem. It is important to observe that if we assume the induced width is bounded (and we know the optimal ordering), the overall algorithm is polynomial. If we consider the case of trees or polytrees, then the induced width is bounded.

2.2 CP-nets

CP-nets are a graphical model for compactly representing conditional and qualitative preference relations (Boutilier et al., 2004). They exploit conditional preferential independence by decomposing an agent’s preferences via the *ceteris paribus* (cp) assumption (all other things being equal). CP-nets bear some similarity to Bayesian networks (see 2.1). Both use directed graphs where each node stands for a single feature (also called variable) from a set of features $F = \{X_1, \dots, X_n\}$ with finite domains D_1, \dots, D_n . For each feature X_i , each user specifies a set of *parent* features $Pa(X_i)$ that can affect her preferences over the values of X_i . This defines a *dependency graph* in which each node X_i has $Pa(X_i)$ as its immediate predecessors. Given this structural information, the user explicitly specifies her preference over the values of X_i for *each complete assignment* on $Pa(X_i)$. This preference is a total or partial order over the domain of X_i (Boutilier et al., 2004). More formally:

Definition 1. A CP-net over a set of features $F = \{X_1, \dots, X_n\}$ with finite domains is a directed graph G over X_1, \dots, X_n . The nodes of the graph G are annotated with conditional preference tables $CPT(X_i)$ for each $X_i \in F$. Each conditional preference table $CPT(X_i)$ associates a total

order $>_i^u$ over X_i 's domain to each complete assignment u of X_i 's parents $Pa(X_i)$. A single row of a CPT is also called cp-statement. The total number of cp-statement in the CP-net is also called the size of the CP-net.

Note that the number of complete assignments over a set of variables (or features) is exponential in the size of the set. However, for any particular instance, there is an implicit constant k that specifies the maximum number of parent features $|Pa(X)|$, that any feature may have. In the following we make no assumptions about the value of k other than that it is fixed for a particular problem instance. With this restriction, and an implicit bound on $|D_X|$ (the domain of the variable X), we can and do treat the size of the conditional preference representation for any X as a constant.

An *acyclic* CP-net is one in which the dependency graph is acyclic, and in this paper we focus on acyclic CP-nets. However, a CP-net need not be acyclic.

Example 1. Consider a CP-net whose features are X_1, X_2, X_3 , and X_4 , with binary domains containing x and \bar{x} if X is the name of the feature, and with the preference statements as follows: $x_1 > \bar{x}_1, x_2 > \bar{x}_2, x_1x_2 : x_3 > \bar{x}_3, \bar{x}_1\bar{x}_2 : x_3 > \bar{x}_3, x_1\bar{x}_2 : \bar{x}_3 > x_3, \bar{x}_1x_2 : \bar{x}_3 > x_3, x_3 : x_4 > \bar{x}_4, \bar{x}_3 : \bar{x}_4 > x_4$. Here, statement $x_1 > \bar{x}_1$ represents the unconditional preference for $X_1 = x_1$ over $X_1 = \bar{x}_1$, while statement $x_3 : x_4 > \bar{x}_4$ states that $X_4 = x_4$ is preferred to $X_4 = \bar{x}_4$, given that $X_3 = x_3$.

The semantics of CP-nets are based on the notion of a *worsening flip*. A worsening flip is a change in the value of a variable to a value which is less preferred by the cp-statement for that variable. We say that one outcome α is *more preferred* than another outcome β (written $\alpha > \beta$) if and only if there is a chain of worsening flips from α to β . This definition induces a partial order over the outcomes. A CP-net thus represents a partial order over the set of possible complete assignments, defined as the transitive closure of the relation induced by direct comparisons through cp-statements under the ceteris paribus assumption.

Given a set of outcomes, an outcome o is optimal if and only if there does not exist any other outcome o' such that $o' > o$, formally given in the following definition.

Definition 2. Given a CP-net inducing a preorder $>$ over the set of outcomes, an optimal outcome is an outcome that is maximal for $>$ in the set of outcomes.

There could exist more than one optimal outcome for the same CP-net, but if the CP-net has an acyclic dependency graph the optimal outcome is unique.

Goldsmith, Lang, Truszczynski, and Wilson (2008) prove that finding an optimal outcome of a general CP-net is PSPACE-complete. However, in acyclic CP-nets, the unique optimal outcome can be found in linear time by sweeping through the CP-net (Boutilier et al., 2004), assigning the most preferred values in the preference tables. We sweep through the CP-net, following the arrows in the dependency graph and assigning at each step the most preferred value in the preference table. Each step in the sweep forward procedure is linear in the number of assignments to parents of the current feature, and there are as many steps as features. This algorithm takes polynomial time in the size of the CP-net description and, if we assume that the number of parents is bounded, is polynomial in the number of nodes.

Two outcomes o and o' can be in one of three possible relations with respect to a CP-net \mathcal{C} :

- o dominates o' : $\mathcal{C} \models o > o'$

- o' dominates o : $\mathcal{C} \models o' > o$
- o and o' are incomparable: $\mathcal{C} \models o \not> o' \wedge \mathcal{C} \models o' \not> o$ written also as $\mathcal{C} \models o \bowtie o'$

where $\mathcal{C} \models o > o'$ means that there exists a worsening path (chain of worsening flips) from o' to o .

A dominance query represents preferential comparison between outcomes.

Definition 3. Given a CP-net \mathcal{C} and a pair of outcomes o and o' , a dominance query returns True if $\mathcal{C} \models o > o'$, False otherwise.

Determining if one outcome is more preferred than another according to the outcome ordering is NP-hard even for acyclic CP-nets (Domshlak & Brafman, 2002; Goldsmith et al., 2008). However, Boutilier et al. (2004) prove that evaluating a dominance query for binary-valued tree structured CP-nets has complexity $O(n^2)$, where n is the number of features. This complexity result is improved by Bigot et al. (2013) that prove that dominance for tree-shaped CP-nets takes $O(n)$ time. Boutilier et al. (2004) prove that dominance is polynomial also for polytrees but NP-complete for acyclic CP-nets that are directed path singly connected, and NP-complete for CP-nets with a bounded number of paths between two nodes. Goldsmith et al. (2008) prove that dominance for CP-nets with a general structure is PSPACE-complete.

2.3 Voting with CP-nets

Voting is a widely studied subject, both in economics (Arrow, Sen, & Suzumura, 2002) and, more recently, in computer science (Brandt, Conitzer, Endriss, Lang, & Procaccia, 2016). In the standard setting, a set of voters express their preferences over a set of candidate alternatives, and a *voting rule* or *social choice rule* maps the set of voters' preferences to one or more elements in the set of candidates (the *winning alternatives*). In many AI settings, however, the set of candidates have a combinatorial structure, with features and feature domains, and preferences over the candidates needs to be modeled via compact preference formalisms.

In this paper we assume that individual preferences are modelled with CP-nets defined on the same set of features. Candidates are all possible outcomes, that is, the complete assignments of the set of features over their domains. To aggregate preferences modelled this way, classical voting rules are not suitable, since they may generate the so-called "multiple election paradoxes" (Brams, Kilgour, & Zwicker, 1998) or may have a high computational complexity to compute the winner (Lang & Xia, 2009). Voting with CP-nets have been already studied extensively. Sequential voting rules have been proposed in this context (Lang, Pini, Rossi, Venable, & Walsh, 2007; Lang & Xia, 2009): First, an ordering of the features is chosen, which should be compatible with the dependency graphs of all the CP-nets, and then voting is applied to one feature at a time in the chosen order.

Let us now introduce the formal setting of voting with CP-nets, together with a number of normative properties for voting rules in this setting, that will be useful to give a first assessment. Let $P = (C_1, \dots, C_m)$ be a *profile* of CP-nets, that is, a vector of m CP-nets each corresponding to an individual in the set $\{1, \dots, m\}$. We assume that each C_i is defined over the same set of variables X_1, \dots, X_n , and that each variable is binary. An *anonymous* profile of CP-nets, denoted with $\bar{P} = ((C_1, freq_1), \dots, (C_k, freq_k))$, specifies for k different CP-nets C_1, \dots, C_k , all defined on the same set of features X_1, \dots, X_n , their relative frequency of appearance in a population of n voters. Given a profile of CP-nets $P = (C_1, \dots, C_m)$, its anonymous version \bar{P} can be easily constructed by setting $freq_i = \frac{|\{j | C_j = C_i\}|}{n}$. In what follows we will indicate with \bar{P} a generic anonymous CP-profile, or the anonymous profile associated to a profile P . In most of the paper we will make use

of anonymous CP-profiles. We will often assume the following property for profiles (Lang & Xia, 2009). We state it here for anonymous profiles:

Definition 4. A profile $\bar{P} = ((C_1, \text{freq}_1), \dots, (C_k, \text{freq}_k))$ is said to be \mathcal{O} -legal if all the dependency graphs of the CP-nets C_i share a topological ordering of the variables.

Finally, a voting rule for CP-nets r is a function that maps each anonymous profile \bar{P} into a single alternative $r(\bar{P})$. Recall that in our case, the candidate alternatives are all possible assignments to the variables. Voting rules are typically assessed by means of desirable properties, often named axioms, that allow us to classify classes of rules. In this paper we consider the following ones:

- **Permutation-neutrality:** The voting rule is not biased towards any of the variables. That is, for any anonymous CP-profile \bar{P} and any permutation $\sigma : \{1, \dots, n\} \rightarrow \{1, \dots, n\}$ on variable names, $r(\sigma(\bar{P})) = r(\bar{P})$.¹
- **Weak Pareto (or unanimity):** If the winner in profile \bar{P} is $r(\bar{P})$, there is no candidate that is preferred by all agents over $r(\bar{P})$. That is, for any profile $\bar{P} = ((C_1, \text{freq}_1), \dots, (C_k, \text{freq}_k))$, if we denote with $>_i$ the preference order associated to C_i , then there is no outcome o such that $o >_i r(\bar{P})$, for all $i \in \{1, \dots, k\}$.
- **Consistency:** If two profiles P_1 and P_2 have the same winner, then the same winner will result by merging the two profiles. That is, given two profiles \bar{P}_1 and \bar{P}_2 over the same candidates, with no voters in common, and such that $r(\bar{P}_1) = r(\bar{P}_2)$, we have $r(\bar{P}_1 \cup \bar{P}_2) = r(\bar{P}_1) = r(\bar{P}_2)$, where $\bar{P}_1 \cup \bar{P}_2$ is the anonymous CP-profiles obtained by concatenating the two profiles, and adapting the relative frequencies accordingly.
- **Participation:** It is always in the interest of a voter to participate in the voting process. Formally, for any profile \bar{P} and any CP-net C_j , $r(\bar{P} \cup C_j)$ is not worse than $r(\bar{P})$ in the ordering $>_j$ associated with C_j .
- **Homogeneity:** If we take several copies of a profile \bar{P} , the result remains the same as taking just \bar{P} . For any profile \bar{P} and any $s \in \mathbb{N}$, we have $r(\bar{P}) = r(\bar{P} \cup \dots \cup \bar{P})$ where \bar{P} is concatenated s times.

The properties we enumerated are rather natural and simple requirements for voting with CP-nets (a complete and detailed axiomatic analysis is outside the scope of this paper). Note that, although all of the above properties seem very natural to have, and it could be hard to think of why a voting rule should not possess them, there are indeed voting rules that do not satisfy some of these properties individually or collections of them (Arrow et al., 2002).

3. Probabilistic CP-nets (PCP-nets)

In this section we introduce a generalization of traditional CP-nets with probabilities on individual cp-statements (Cornelio et al., 2013; Bigot et al., 2013). We assume that all the cp-statements are independent of each other since this allows us to use efficient algorithms and techniques from

1. With a slight abuse of notation, we denote with $\sigma(\bar{P})$ the anonymous CP-profile obtained from \bar{P} by permuting the names of the variables in each CP-net C_i of \bar{P} according to σ .

Bayesian Networks. Moreover in many scenarios this assumption is very reasonable: for example, when data comes from several different individuals, such as in a multi-agent scenario (see Section 6), it is reasonable to treat the incoming preferences as independent. However, there exist also many scenarios where it makes sense to remove this independence assumption (e.g., often people who prefer fish over meat also prefer wine over beer) and we plan to study them in future work.

Definition 5. A *Probabilistic CP-net* (PCP-net) has a feature set $F = \{X_1, \dots, X_n\}$, where each feature has domain D_1, \dots, D_n . For each feature X_i , there is a set of parent features $\text{Pa}(X_i)$. This defines a dependency graph in which each node X_i has directed edges from all features in $\text{Pa}(X_i)$. Given this structural information, for each feature X_i , there is a probability distribution over the set of all preference orderings (total orderings over D_i) for each complete assignment on $\text{Pa}(X_i)$. Given a feature X in a PCP-net, its PCP-table is the table associating each combination of the values of the parent features of X with a probability distribution over the set of total orderings over the domain of X .

Note that PCP-nets are a strict generalization of CP-nets. When a PCP-net is restricted to probabilities in $\{0, 1\}$ we recover the definition of CP-nets (Boutilier et al., 2004).

In all the examples and experimental settings in this paper, we assume the domain of each variable to be binary and the induced width k of the dependency graph to be bounded by a constant. However, the definition of PCP-nets does not require these restrictions. Moreover, in what follows, we use a lexicographic tie-breaking rule when needed, to ensure a unique top element (since generally PCP-nets may have several ones). The proofs for all theorems and lemmas are in the Appendix.

Definition 6. Given a PCP-net \mathcal{Q} , a CP-net is said to be induced by \mathcal{Q} if it has the same variables, with the same domains, same dependency graph as \mathcal{Q} and the CP-table of each variable X_i is generated by choosing a specific ordering (with non-zero probability) for each combination of the values of the parent features $\text{Pa}(X_i)$.

Example 2. Consider the PCP-net in Figure 1(a) with two features, X and Y , with domains $D_X = \{x, \bar{x}\}$ and $D_Y = \{y, \bar{y}\}$. We can observe: the graph with the two nodes X and Y representing (through the edge between X and Y) a preferential dependency between the two features; the preference table under the node X showing for each ordering of the values of X the corresponding probability; and the preference table under the node Y showing for each assignment of X the probability of observing a given ordering over values of Y . Figure 1(b) describes a CP-net that has been induced from the PCP-net of Figure 1(a).

It is important to notice that by generating the induced CP-tables we obtain redundant information when we select rows that are equal for each assignment to the parent variables. For example, considering a binary PCP-net with two variables X and Y and an edge from X to Y , we can have an induced CP-net with the following CP-table for the variable Y : $x : y > \bar{y}$ and $\bar{x} : y > \bar{y}$. This redundant information can be interpreted as *vacuous edges*: edges that are not actually dependencies. This means that removing these type of edges, does not change the preorder induced over the set of outcomes. Eliminating this redundant information (all the vacuous edges) allows us to reduce an (induced) CP-net to its *normal form*.

Definition 7. A CP-net is in normal form if it doesn't have any vacuous edge. This means that for each node X there does not exist any node $Y \in \text{Pa}(X)$ such that the rows of the CP-table of X ,

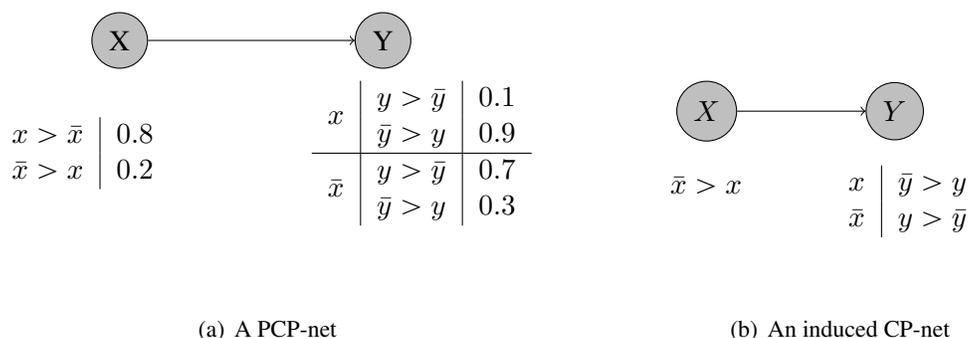


Figure 1: A PCP-net and one induced CP-net.

in correspondence to the parents assignments $u_i y_1, u_i y_2, \dots, u_i y_l$, have the same ordering for the values of the variable X , for each u_i in the set $\{u_1, \dots, u_m\} = \text{Dom}(Y_1) \times \dots \times \text{Dom}(Y_k)$ where $\text{Dom}(Y) = \{y_1, \dots, y_l\}$ and $\text{Pa}(X) = \{Y, Y_1, \dots, Y_k\}$.

It is important to notice that transforming a CP-net not in normal form to another CP-net that is in normal form is always possible. This transformation preserves the same induced preorder over the set of outcomes. Moreover the process is invariant to the order in which vacuous edges are removed to transform a given a CP-net not in normal form to its normal form.

Example 3. This example shows the difference between a CP-net not in normal form and its normal form. The right part of Figure 2 shows the normal form of the CP-net in the left part of Figure 2.

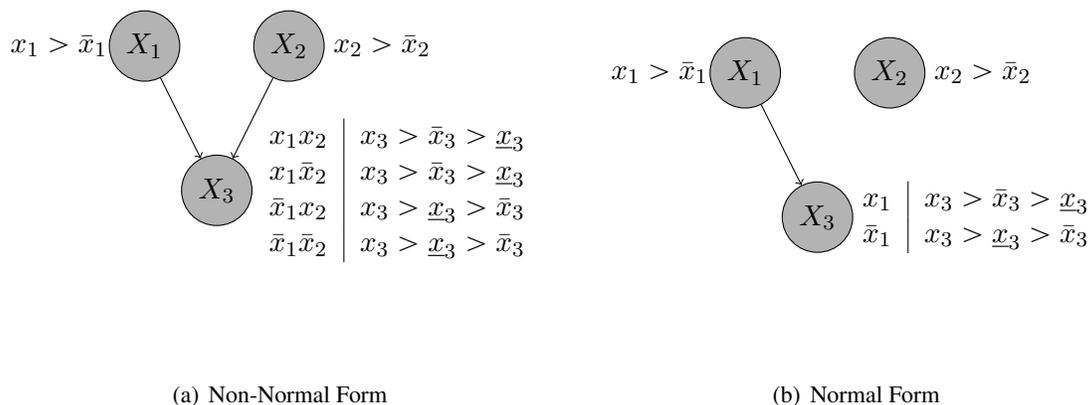


Figure 2: A CP-net and its associated CP-net in normal form. Note that this example uses ternary domains (x, \bar{x} , and \underline{x})

If we transform an induced CP-net in normal form, we observe that, after the transformation, the edges are a subset of the edges in the PCP-net. Thus, CP-nets induced by the same PCP-net may have different dependency graphs, if considered in their normal form.

Each induced CP-net has an associated probability, obtained from the PCP-net by taking the product of the probabilities of the orderings chosen in the CP-net.

Definition 8 (Probability of an Induced CP-net). *Given a PCP-net \mathcal{Q} and an induced CP-net C , let \bar{q} be the set of all rows of the CPTs that define C . We define the probability of C , $\text{fp}_C(\bar{q})$, to be the product of the probabilities of the $q_i \in \bar{q}$.*

Hence, a PCP-net induces a probability distribution over the set of all induced CP-nets.

Example 4. *Consider again the PCP-net in Figure 3(a) and the induced CP-net in Figure 3(b). The probability associated with this CP-net is defined by: $\text{fp}_C(\bar{q}) = [(1 - q_1) \cdot (1 - q_{2_1}) \cdot q_{2_2}]$.*

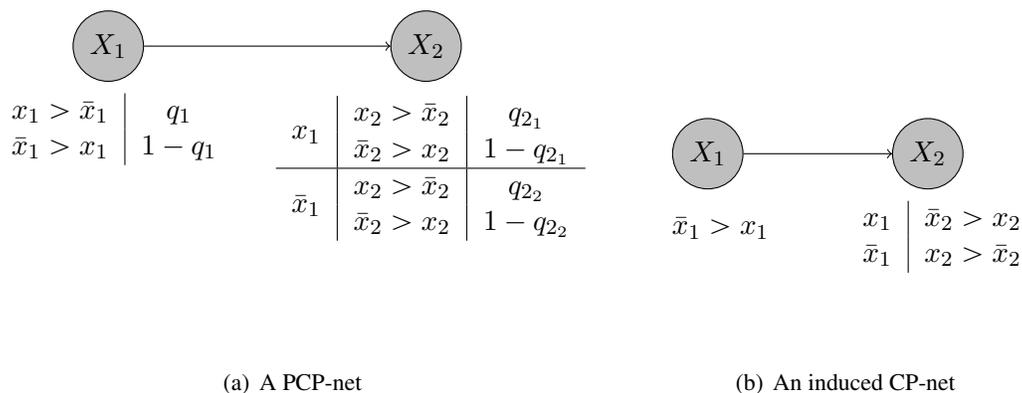


Figure 3: A PCP-net where the probability of the cp-statements are given as variables and one induced CP-net.

3.1 Probability over the Edges

In this section we assume that all the CP-nets are in normal form. As shown in the previous section, a PCP-net induces a collection of CP-nets, each one with a possibly different dependency graph. Each edge of the PCP-net dependency graph may or may not appear in an induced CP-net. Thus each edge, which represents whether or not some feature is dependent on another, is associated with a probability of existing.

This probability is important as the modification of the CP-nets graph through the addition or deletion of edges can have significant impacts on the corresponding partial order over the set of outcomes described by the CP-net. As Boutilier et al. (2004) show, parent variables are more important than their child variables in the ceteris paribus semantics, since violating the preference for a parent variable is less preferred than violating the preference for any of its children.

Furthermore the probability of an edge has an additional interpretation depending on the semantics of the context in which we are using the PCP-nets. For example, if we consider a PCP-net as the representation of the preferences in a multi-agent system as we have proposed, the edge probability can be interpreted as a measure of the agreement of the agents in the profile on the dependency between two features, with the assumption that the rows are indeed independent from each other in the population. Hence, in the multi-agent context, the semantics of the PCP-net edges is useful

as it allows for a quick, interpretable understanding of the degree of agreement between the agents about a preferential dependency. This ability can be crucial in many scenarios, for example, when performing a market survey and using PCP-nets to represent customer preferences, the probability over the edges allows the surveyor to prioritize some features over others, if the majority of the customers find them more important.

Definition 9. *Given a PCP-net \mathcal{Q} and an edge e in the dependency graph of \mathcal{Q} , we define the probability of the existence of the edge e as the sum of the probabilities of those induced CP-nets in normal form that have e in their dependency graph:*

$$\mathbb{P}(e) = \sum_{i \in I \text{ s.t. } e \in C_i} \mathbb{P}(C_i)$$

where I is the index set of induced CP-nets C_i .

This definition implies that the probability of non-existence of an edge e , thus the probability of having an induced CP-net without the edge e , is equal to $1 - \mathbb{P}(e)$.

This probability can be derived using the PCP-table of the child node. We consider an edge e , from a feature X to a feature Y . The set of Y 's parent nodes are the features in the set $Pa(Y)$ with cardinality $k = |Pa(Y)|$. We define \mathcal{E} as:

$$\mathcal{E} = \prod_{F \in (Pa(Y) \setminus X)} D_F$$

where D_F is the domain for a variable F .

In the following theorem we show how to compute the probability of an edge as the complement of the probability of its nonexistence. This last probability is computed by summing the probability of having the same preference order for the second variable conditioned to the different values of the parent node (defined by the edge), considering each instantiation of the other possible parent nodes. The proofs for all theorems and lemmas can be found in the Appendix.

Theorem 1. *Given a PCP-net \mathcal{Q} with features with binary domains, and given $\Upsilon = \{\sigma : \mathcal{E} \rightarrow \{y > \bar{y}, \bar{y} > y\}\}$, the set of all possible functions σ that associate to each element $v \in \mathcal{E}$ an element of the set $\{y > \bar{y}, \bar{y} > y\}$ (the set of all the possible total orders of the domain of Y), then the probability of an edge $e = (X, Y)$ can be computed by the following conditional probabilities:*

$$\mathbb{P}(e) = 1 - \sum_{\sigma \in \Upsilon} \left(\prod_{v \in \mathcal{E}} \mathbb{P}[\sigma(v)|vx] \mathbb{P}[\sigma(v)|v\bar{x}] \right).$$

The previous theorem can be easily generalized to PCP-net with features with non-binary domains as follows:

Theorem 2. *The probability of the edge $e = (X, Y)$ can be computed as follows:*

$$\mathbb{P}(e) = 1 - \sum_{\sigma \in \Upsilon} \left(\prod_{v \in \mathcal{E}} \mathbb{P}[\sigma(v)|vx_1] \mathbb{P}[\sigma(v)|vx_2] \cdots \mathbb{P}[\sigma(v)|vx_d] \right)$$

where $D_X = \{x_1, x_2, \dots, x_d\}$, $\Theta(Y)$ is the set of all the possible total orders of the domain of Y and $\Upsilon = \{\sigma : \mathcal{E} \rightarrow \Theta(Y)\}$ is the set of all possible functions σ that associate to each element $v \in \mathcal{E}$ a total order over the domain of Y (an element of the set $\Theta(Y)$).

4. Optimality

Given a PCP-net we study mainly two optimality tasks: finding the most probable induced CP-net and finding the most probable optimal outcome.

Definition 10. *Given a PCP-net \mathcal{Q} , the most-probable induced CP-net C^* is the induced CP-net with highest probability, considering the probability distribution defined by \mathcal{Q} over the set of induced CP-nets (see Definition 8):*

$$C^* = \arg \max_{C_i} \mathbb{P}(C_i)$$

Another relevant optimality task is to find the *optimal outcome of the most probable induced CP-net*, that can be computed using known techniques for optimality over CP-nets (e.g., the sweep forward procedure mentioned in Section 2.2).

Definition 11. *Given a PCP-net \mathcal{Q} , the most probable optimal outcome o^* is the outcome that occurs with the greatest probability as the optimal one in the set of induced CP-nets:*

$$o^* = \arg \max_{o_j} \sum_{C_i \in \mathcal{C}_{o_j}} \mathbb{P}(C_i)$$

where \mathcal{C}_{o_j} is the set of induced CP-nets that have o_j as optimal outcome.

These two reasoning tasks have slightly different semantics and may be of use to different groups in the preference reasoning community. The most probable induced CP-net is analogous to the CP-net that has the highest probability to map onto a user in a profile. Whereas, the most probable optimal outcome would be what a recommendation engine should suggest to maximize the probability of being optimal. One can be seen as an aggregated model, that still retains usefulness for prediction and sampling, while the other is an aggregated outcome, that maximizes the probability of being optimal.

In this section we do not make any general restriction on the PCP-net structure beside acyclicity. Thus, unless otherwise specified, we just assume an acyclic structure of the dependency graph.

4.1 The Most Probable Induced CP-net

We reduce the problem of finding the most probable induced CP-net, that is, the induced CP-net with highest probability according to Definition 8, to that of finding an assignment with maximal joint probability of an appropriately defined BN that we call the *general network* or *G-net*. Making this explicit connection to BNs is important for several reasons: it formalizes the connection between PCP-nets and BNs, which is of independent mathematical interest; by establishing this connection several of our proofs become easier and it is simpler to write the equations for finding the probability of a CP-net induced by a PCP-net; and finally our PCP-net formalism (and that of Cornelio et al. (2013)) assumes that within a CP-table all statements are independent and all variables are independent as well.

Definition 12. *Given a PCP-net \mathcal{Q} , we define the Bayesian network called the general network, or G-net(\mathcal{Q}), associated with \mathcal{Q} , as follows:*

- *For each independent feature A of the PCP-net, the G-net has a corresponding variable A . The domain of A in G-net(\mathcal{Q}) corresponds to the set of all possible total orderings over the*

domain of the corresponding feature A in \mathcal{Q} . The probability distribution over these orderings is given by the PCP-table of A .

- For each dependent feature B of \mathcal{Q} , $G\text{-net}(\mathcal{Q})$ has as many variables as there are combinations of value assignments to the parents. The domain of the B variables in the $G\text{-net}(\mathcal{Q})$ corresponds to the set of all possible total orderings over the domain of the feature B in \mathcal{Q} . They are independent, since we consider all the cp-statement as independent.

Example 5. Consider the PCP-net with two features, \mathcal{Q} , from Example 2 and Figure 1. The corresponding G-net is shown in Figure 4. The variables have the following domains: $D_X = \{x > \bar{x}, \bar{x} > x\}$, $D_{Y_x} = D_{Y_{\bar{x}}} = \{y > \bar{y}, \bar{y} > y\}$.

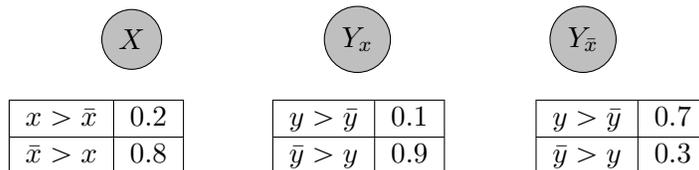


Figure 4: The G-net associated with the PCP-net in Example 2 and Figure 1.

By definition of G-net, given a PCP-net \mathcal{Q} and the corresponding G-net N , there is a one-to-one correspondence between the assignments of N and the induced CP-nets of \mathcal{Q} . This is because an assignment to a G-net N corresponds to a choice of a specific total order (over the domain of the corresponding feature in \mathcal{Q}) for each variable and for each parent assignment. This defines a specific total order in the PCP-tables of each variable and for each parent assignment, characterizing an induced CP-net.

Theorem 3. Given a PCP-net \mathcal{Q} and its set of induced CP-nets, then:

- The probability of realizing one of its induced CP-nets is the joint probability of the corresponding assignment in the G-net for \mathcal{Q} ;
- The probability distribution defined in Definition 8 over the set of induced CP-nets is a well defined discrete probability distribution (i.e., the probabilities over the domain elements sum to one and the probability of each element belongs to the interval $[0, 1]$).
- The most probable induced CP-net corresponds to the CP-net associated (using the one-to-one correspondence between induced CP-nets and assignments of the corresponding G-net described above) to the assignment of the G-net for \mathcal{Q} , with maximal joint probability.

Because the G-net has a separable dependency graph (there are no edges and all the nodes are independent), to compute the assignment with maximal joint probability takes $O(\tilde{n})$ time, where \tilde{n} is the number of the nodes of the G-net. If k is the maximal number of parents nodes for each node (assuming a fixed constant as maximum value for k) and d the maximal cardinality of the domain of the nodes of the PCP-net, then $\tilde{n} = n \cdot d^k$ (n is the number of nodes in the PCP-net). We therefore

have the computational complexity $O(nd^k)$ (since the G-net is separable) that is polynomial in the size of the description (nodes and PCP-tables) of the PCP-net.

The most probable induced CP-net could also be found by selecting the most probable ordering in the PCP-net CP-tables (with the same computational complexity as in G-nets). However, G-nets are useful for the more general case where there are dependencies between cp-statements. Extending PCP-nets to remove the independence assumption is an interesting direction for future work and establishing the relationship with BNs will help with this process.

The *G-net* concept (and the *Opt-net* we introduce in Definition 13) that we introduce here has been used to reason about optimally in probabilistic TCP-nets (Ahmed, 2020; Ahmed & Mouhoub, 2018b).

4.2 PCP-nets vs Induced CP-nets

Given a discrete probability distribution \vec{P} over a collection of CP-nets \mathcal{C} , all with the same features and domains, there may be no PCP-net \mathcal{Q} such that the set \mathcal{C} of CP-nets are all and only the CP-nets induced by \mathcal{Q} .

Theorem 4. *Given a discrete probability distribution \vec{P} over a set of CP-nets \mathcal{C} (whether or not they have the same dependency graph), there may exist no PCP-net inducing \vec{P} over the set \mathcal{C} .*

Theorem 5. *Given a probability distribution over a set of CP-nets, each of which contains the same feature variables, if this distribution is generated by a PCP-net \mathcal{Q} , then there exists and we can define a set of non-linear equations that provides as its unique solution the PCP-net \mathcal{Q} inducing them.*

Theorem 4 and 5 give us interesting clues about the practicality of eliciting and aggregating CP-nets. They indicate that, in some instances, we can derive a PCP-net for a population by eliciting CP-nets from individuals. However, solving the system of equation generated by Theorem 5 is likely a computationally hard problem since in general systems of non-linear equations are not even decidable; there may be a direct algorithm to solve this problem (given the fact that the system of equation constrains the solutions in the interval $[0, 1]$), but we leave this for future work.

4.3 The Most Probable Optimal Outcome

The most probable optimal outcome is the outcome that occurs with the greatest probability as the optimal one in the set of induced CP-nets. The probability that an outcome is optimal corresponds to the sum of the probabilities of the CP-nets that have that outcome as optimal.

To reason about this task, we first observe that the most probable optimal outcome may not be the optimal outcome of the most probable CP-net.

Example 6. *Let us consider a PCP-net with only one feature X with domain $D_X = \{x_1, x_2, x_3\}$ and the following PCP-table:*

- $x_1 > x_2 > x_3$ with probability 0.3
- $x_1 > x_3 > x_2$ with probability 0.3
- $x_3 > x_2 > x_1$ with probability 0.4

The most probable CP-net is the one corresponding to the third ordering and it has the optimal outcome x_3 . The other CP-nets have x_1 as optimal, so $\mathbb{P}(x_1) = 0.6$ and $\mathbb{P}(x_3) = 0.4$. The most probable optimal outcome is therefore x_1 but the optimal outcome of the most probable CP-net is x_3 .

Therefore, to find the most probable optimal outcome, we cannot use the most probable induced CP-net (by the G-net procedure described) and then find its optimal outcome. Rather, we make use of another Bayesian network, that we call the *Opt-net*.

In general, given a PCP-net \mathcal{Q} , the *optimal network* (*Opt-net*) for \mathcal{Q} is a BN with the same dependency graph as \mathcal{Q} . Thus, the Opt-net has a variable for each of the PCP-net's features. The domains of the variables in the Opt-net are the values of the corresponding features that are ranked first in at least one ordering with non-zero probability. The conditional probability tables of the Opt-net are obtained from the corresponding PCP-tables as follows: for each assignment of the parent variables, we consider the corresponding probability distribution over the values of the dependent variable defined in the PCP-table. The probability of a value for the dependent variable is the sum of the probabilities of the orderings that have that particular value as most preferred according to that distribution. More formally:

Definition 13. Given a PCP-net \mathcal{Q} , its Opt-net is a Bayesian network N such that:

- for each node $X^{\mathcal{Q}}$ in \mathcal{Q} , N has a node X^N ;
- the domain $Dom(X^N) \subseteq Dom(X^{\mathcal{Q}})$ and consists of the values of $X^{\mathcal{Q}}$ that are ranked first in at least one ordering with non-zero probability in the CP-table of $X^{\mathcal{Q}}$;
- for each edge $(X^{\mathcal{Q}}, Y^{\mathcal{Q}})$ in \mathcal{Q} , N has an edge (X^N, Y^N) ;
- the probability distribution over $Dom(X^N)$ is defined as:

$$\mathbb{P}(w|u_i) = \sum_{j \in \mathcal{J}_w} p_j^i, \quad \forall w \in Dom(X^N)$$

where:

- u_i is an assignment of the variables in $Pa(X^N)$;
- $\mathcal{J}_w = \{j \mid o_j[1] = w\}$, where o_j is a total order over $Dom(X^N)$ and $o_j[1]$ is the top element in the ordering o_j ;
- p_j^i is the probability of the ordering o_j given the assignment u_i in the PCP-net \mathcal{Q} .

Example 7. Consider the PCP-net \mathcal{Q} with three features A , B and C with domains $D_A = \{a_1, a_2\}$, $D_B = \{b_1, b_2\}$ and $D_C = \{c_1, c_2, c_3\}$ shown in Figure 5.

The Opt-net has the same dependency graph as \mathcal{Q} , with three variables A , B and C with domains: $D_A = \{a_1, a_2\}$, $D_B = \{b_1, b_2\}$ and $D_C = \{c_1, c_2\}$, and two edges AC and BC . The domain of variable C in the Opt-net does not contain value c_3 because it never appears as most preferred in any ordering. Therefore, the Opt-net has a table for entry a_1b_2 where c_1 appears with probability 0.2 and c_2 appears with probability 0.8. The Opt-net is shown in Figure 5.

Theorem 6. Given a PCP-net \mathcal{Q} and its Opt-net:

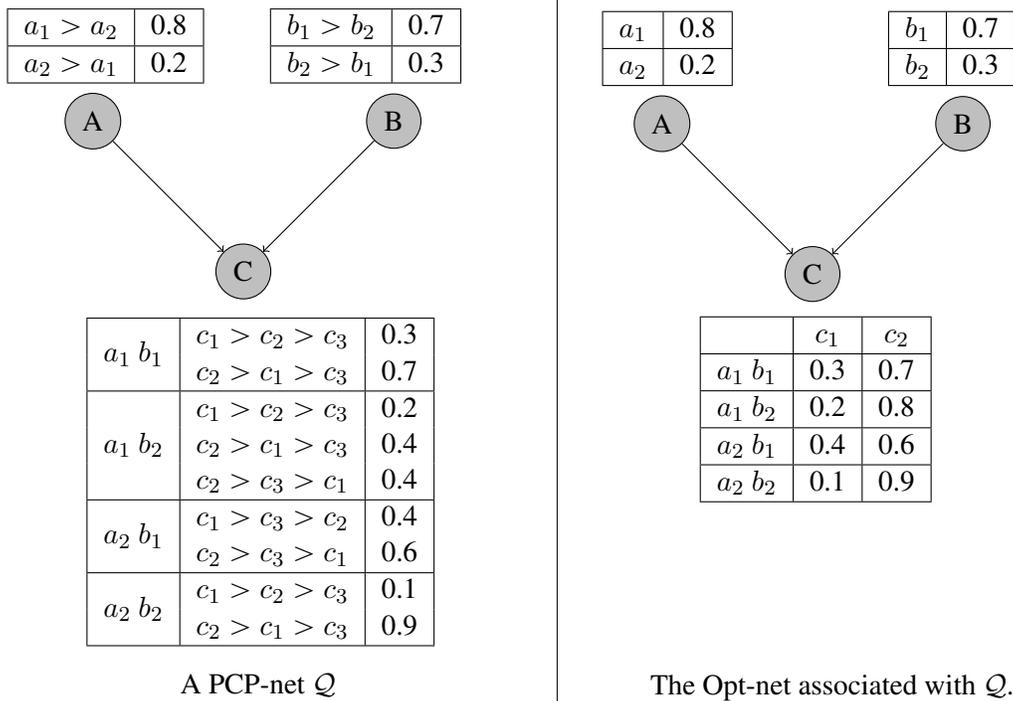


Figure 5: A PCP-net (left) and its associated Opt-net (right). We have omitted the rows of the cp-tables with zero probabilities for simplicity.

- *There is a one-to-one correspondence between the assignments (with non-zero probability) of the Opt-net and the outcomes that are optimal in at least one induced CP-net of \mathcal{Q} .*
- *Given a PCP-net \mathcal{Q} , the probability that an outcome is optimal is the joint probability of the corresponding assignment in the Opt-net (we consider only the assignments of the Opt-net with positive probability). If no such corresponding assignment exists, then the probability of being optimal is 0.*
- *The most probable optimal outcome for a PCP-net \mathcal{Q} is obtained by computing the assignment that maximizes the joint probability of the Opt-net.*

Since the problem of maximum assignment in BNs is NP-hard, the complexity of computing the most probable optimal outcome for a generic PCP-net is not harder than NP-hard. Moreover, if the structure of the PCP-net dependency is a poly-tree with bounded induced width, computing the most probable optimal outcome becomes polynomial in the PCP-net description (see Section 2.1). In the case of binary PCP-net with tree structure, Bigot et al. (2013) show that this task can be solved in linear time by dynamic programming. Notice that the Opt-net can solve the problem with the same computational complexity under these restrictions, however the Opt-net works for any PCP-net.

4.4 Optimality Under Partial Observations

A PCP-net models a distribution over a collection of CP-nets: the set of induced CP-nets. The results/definitions introduced above for the two optimality tasks (most probable induced cp-net and most probable optimal outcome) can be easily extended to deal with the presence of some evidence, that is, observation of partial outcomes. This is true also for the dominance task that we will introduce in the next section (Section 5).

A partial observation for some subset of features of a PCP-net is an instantiation of such features to a value in their domain. Given a PCP-net \mathcal{Q} and an instantiation of some variable $X = x_i$, we can compute the result of either optimality task on \mathcal{Q} given the evidence, by performing the standard optimality tasks on another PCP-net \mathcal{Q}' (a sub-PCP-net of \mathcal{Q}). We obtain \mathcal{Q}' from \mathcal{Q} by removing X and its outgoing edges, and modifying the cp-tables of all the features depending on X (i.e., the children nodes) to include only the rows that correspond to the value observed for X (i.e., with the assignment $X = x_i$).

Example 8. *Consider the PCP-net \mathcal{Q} with three features A , B and C with domains $D_A = \{a_1, a_2\}$, $D_B = \{b_1, b_2\}$ and $D_C = \{c_1, c_2, c_3\}$ shown in Figure 6 (left). Consider the observation $A = a_2$. We construct the PCP-net \mathcal{Q}' that has only the two features B and C with the same domains as in \mathcal{Q} , and only one edge, BC , as shown in Figure 6 (right).*

5. Dominance in PCP-nets

In this section we restrict to acyclic dependency graphs, moreover in Section 5.2, Section 5.3, Section 5.4 and Section 5.5 we further restrict to binary-valued PCP-nets that have a polytree structure.

Dominance queries are another relevant task to perform on a given PCP-net. In the PCP-net context this means computing the probability that one outcome dominates another one. In a multi-agent scenario it corresponds to understanding whether one outcome is preferred by most agents.

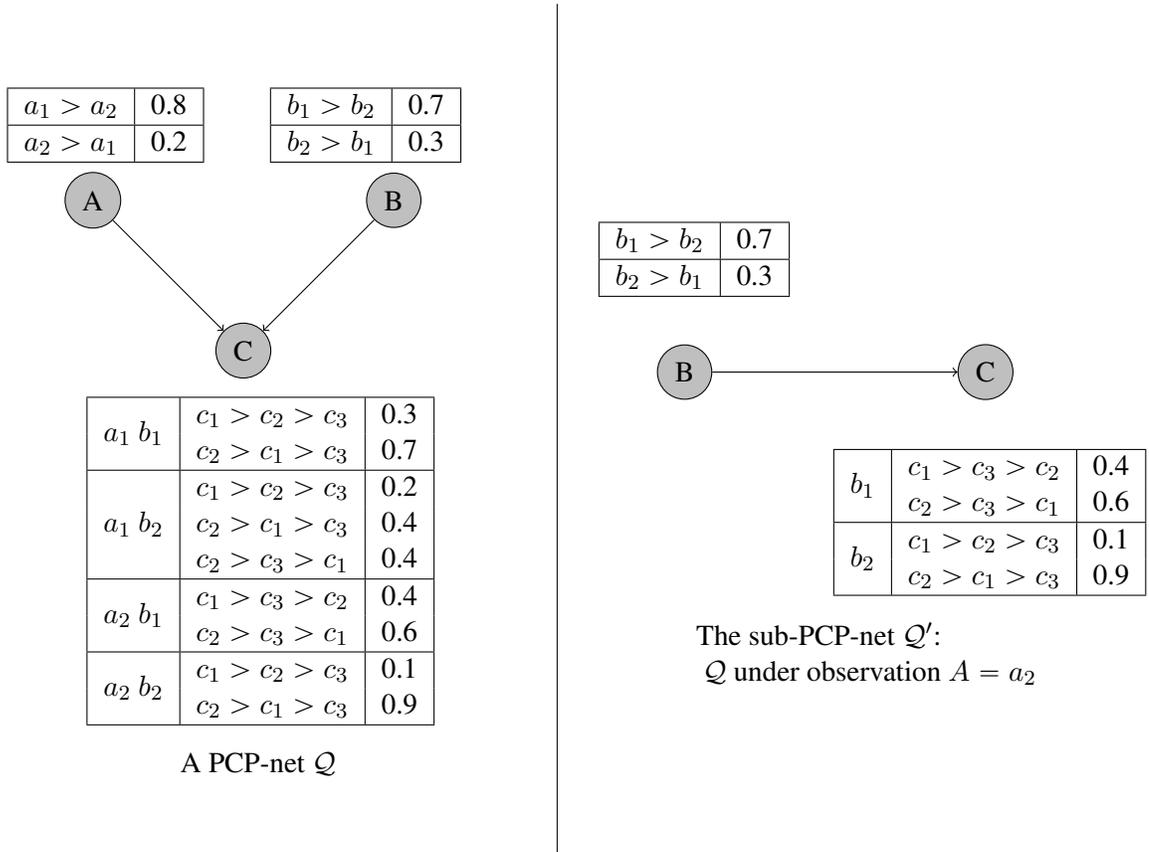


Figure 6: A PCP-net (left) and its associated sub-PCP-net under observation $A = a_2$ (right). We have omitted the rows of the cp-tables with zero probabilities for simplicity.

Definition 14 (Dominance for PCP-nets). *Given a PCP-net Q and a pair of outcomes o and o' , a dominance query returns the probability that $Q \models o > o'$. That is:*

$$\mathbb{P}_Q(o > o') = \sum_{\substack{C \text{ induced by } Q \\ \text{s.t. } C \models o > o'}} \mathbb{P}(C).$$

Since any CP-net C can be interpreted as a PCP-net with probability values in $\{0, 1\}$, a dominance query for C returns either 0 or 1, where 0 corresponds to $C \not\models o > o'$ and 1 corresponds to $C \models o > o'$.

5.1 Computational Cost of Dominance

A CP-net is a particular PCP-net where all the probabilities in the PCP-tables are 0 or 1. Thus a CP-net can be seen as a subcase of PCP-net. It was shown by Boutilier et al. (2004) that evaluating a dominance query for binary-valued CP-nets is quadratic in the number of variables for tree structured CP-nets. Later Bigot et al. (2013) provided an algorithm for dominance testing in tree structured CP-nets with computational complexity $O(n)$ where n is the number of features.

Moreover, dominance testing is polynomial (in the size of the CP-net description) for poly-trees (Boutilier et al., 2004; Domshlak & Brafman, 2002) and NP-complete for directed-path singly connected CP-nets (Boutilier et al., 2004), even if the in-degree of the nodes is bounded (Domshlak & Brafman, 2002), and for CP-nets with a bounded number of paths between each pair of nodes (Boutilier et al., 2004; Domshlak & Brafman, 2002). Finally, Goldsmith et al. (2008) proved that dominance is PSPACE-complete for general CP-nets .

The difficult cases remain of course difficult for PCP-nets since CP-nets are a restricted form of PCP-nets. Bigot et al. (2013) prove that dominance for PCP-nets is #P-hard, even if the graph is acyclic, the longest path has length 3 and each node has at most one outgoing edge and at most 4 parents. They also give an algorithm to obtain dominance for a binary-valued tree structured PCP-net that takes $O(2^{2h^2}n)$ time, where n is the number of features and h is the number of variables that have a different value in o and o' .

Computing the probability of dominance for separable PCP-nets (with no edges in their dependency graph) is polynomial (see Section 5.2), but computing the exact value of the probability of dominance for a more structured PCP-net is hard. In the next sections we detail algorithms to compute both a lower bound (Section 5.2) and an upper bound (Section 5.3) in polynomial time for PCP-nets if the dependency graph is a polytree (a polytree is a directed graph whose underlying undirected graph is a tree).

5.2 Lower Bound

In this section we consider only binary-valued PCP-nets that have a polytree structure. This category includes directed trees.

Assuming the polytree structure (thus acyclicity), we can define an order O over n features X_i such that $\forall i \in \{1, \dots, n\}, Pa(X_i) \subseteq \{X_1, \dots, X_{i-1}\}$.

In what follows, given an outcome $o = o_1 o_2 \dots o_n$, $o \upharpoonright_{X_i}$ will indicate o_i and $o \upharpoonright_S$ will indicate $o \upharpoonright_{s_1} \dots o \upharpoonright_{s_m}$ where $S = \{s_1, \dots, s_m\} \subseteq \{X_1, \dots, X_n\}$ and S is such that $\forall s_i \in S [Pa(s_i) \subset S]$. Given a CP-net C and two outcomes o and o' , when we use the notation $o \upharpoonright_S >_S o' \upharpoonright_S$, $o \upharpoonright_S <_S o' \upharpoonright_S$ or $o \upharpoonright_S \bowtie_S o' \upharpoonright_S$ we mean that the comparison is considered in the sub-CP-net C' that has the features in S and the arcs in C that connect them, with the corresponding CP-tables (to simplify the heavy notation, in what follow we will indicate $>_S$, $<_S$, and \bowtie_S as $>$, $<$, and \bowtie since the restriction is clear from the context).

Definition 15. Given a PCP-net \mathcal{Q} with n variables X_i and given two outcomes o and o' , let Diff be the set of feature indices that have different values on o and o' :

$$\text{Diff} = \{i \mid o \upharpoonright_{X_i} \neq o' \upharpoonright_{X_i}\}.$$

The lower bound for dominance for $o > o'$ in \mathcal{Q} is defined as follows:

$$\mathbb{P}_{\mathcal{Q}}^L(o > o') = \prod_{i \in \text{Diff}} (1 - \bar{\rho}_i) \quad (1)$$

where $\forall i \in \text{Diff}$, $\bar{\rho}_i$ is defined as:

$$\bar{\rho}_i = \begin{cases} \prod_{u \in U_i^{o, o'}} \mathbb{P}(o' \upharpoonright_{X_i} > o \upharpoonright_{X_i} | u) & \text{if } Pa(X_i) \neq \emptyset \\ \mathbb{P}(o' \upharpoonright_{X_i} > o \upharpoonright_{X_i}) & \text{if } Pa(X_i) = \emptyset \end{cases}$$

where u is an assignment of the parents of X_i and $U_i^{o,o'}$ is the set of all assignments to $\text{Pa}(X_i)$ such that, given $Y \in \text{Pa}(X_i)$, if $o \upharpoonright_Y = o' \upharpoonright_Y$ then $u \upharpoonright_Y = o \upharpoonright_Y = o' \upharpoonright_Y$ (otherwise, if $o \upharpoonright_Y \neq o' \upharpoonright_Y$ then $u \upharpoonright_Y \in \{y, \bar{y}\}$).

We require Definition 16 and the following lemmas in the proof of Theorem 8. In the first lemma we consider CP-nets and we show that if $o > o'$ then the same dominance relation holds on all prefixes of equal length of the two outcomes.

Lemma 1. *Given an acyclic CP-net C with n nodes and two outcomes $o = o_1 o_2 \cdots o_n$ and $o' = o'_1 o'_2 \cdots o'_n$ such that $o > o'$ and given i an index in $\{1, \dots, n\}$ such that $o_i \neq o'_i$ and $o_j = o'_j \forall j < i$ then*

$$o \upharpoonright_{X_1 \dots X_j} > o' \upharpoonright_{X_1 \dots X_j} \quad \forall j \geq i.$$

In Lemma 2 and Corollary 6.1 we prove that if o and o' have two incomparable prefixes (of equal length and involving the same variables), then, o and o' are incomparable (denoted with $o \bowtie o'$).

Lemma 2. *Given an acyclic CP-net C with n nodes and two outcomes o and o' , if there exists an index $i \in \{1, \dots, n\}$ such that $o \upharpoonright_{X_1 \dots X_i} \bowtie o' \upharpoonright_{X_1 \dots X_i}$ then we have that $o \upharpoonright_{X_1 \dots X_j} \bowtie o' \upharpoonright_{X_1 \dots X_j}$ for each index j between i and n ($i \leq j \leq n$).*

Corollary 6.1 (of Lemma 2). *Given an acyclic CP-net C with n nodes and two outcomes o and o' such that there exists an index $i \in \{1, \dots, n\}$ such that $o \upharpoonright_{X_1 \dots X_i} \bowtie o' \upharpoonright_{X_1 \dots X_i}$, then $o \bowtie o'$.*

In the following lemma we prove that, given two incomparable outcomes, there is a prefix length threshold (denoted as index i in Definition 16), such that all prefixes of shorter length are ordered and all those of equal or greater length are incomparable.

Lemma 3. *Given an acyclic CP-net C with n nodes and two outcomes o and o' , then $o \bowtie o'$ if and only if there exists index $i \in \{1, \dots, n\}$ such that $o \upharpoonright_{X_1 \dots X_j} \bowtie o' \upharpoonright_{X_1 \dots X_j} \forall j < i$ and $o \upharpoonright_{X_1 \dots X_l} \bowtie o' \upharpoonright_{X_1 \dots X_l} \forall l \geq i$.*

Definition 16. *Given two outcomes o and o' such that $o \bowtie o'$, we say that they are incomparable for the index i ($o \bowtie_i o'$) if i satisfies Lemma 3.*

In the following lemma we connect a topological property of the dependency graph of a CP-net to the incomparability of certain pairs of outcomes.

Lemma 4. *Given an acyclic CP-net C with n nodes formed by two connected components C_1 and C_2 , and given two different outcomes o and o' such that $[(o \upharpoonright_{C_1} > o' \upharpoonright_{C_1} \text{ and } o \upharpoonright_{C_2} < o' \upharpoonright_{C_2}) \text{ or } (o \upharpoonright_{C_1} < o' \upharpoonright_{C_1} \text{ and } o \upharpoonright_{C_2} > o' \upharpoonright_{C_2})]$ then $o \bowtie o'$.*

Theorem 7. *Given a polytree-structured CP-net C with n variables and two outcomes o and o' then, if for all X_i in Diff (see Definition 15), there exists an assignment $u_i \in U_i^{o,o'}$ (see Definition 15) such that $u_i : o \upharpoonright_{X_i} > o' \upharpoonright_{X_i}$ then $o > o'$.*

We are now in the position of proving the main result.

Theorem 8. *Given a PCP-net \mathcal{Q} and two outcomes o and o' , $\mathbb{P}_{\mathcal{Q}}^L(o > o')$ is a lower bound for $\mathbb{P}_{\mathcal{Q}}(o > o')$.*

Observe that the lower bound is a product of at most n factors. The computation of each factor is $O(2^k)$ multiplications, where k is the maximal number of parents. Thus, the formula can be computed in $O(n2^k)$ time.

Example 9. Consider a PCP-net \mathcal{G} as defined in Figure 7 and outcomes $o = x_1x_2\bar{x}_3x_4x_5$ and $o' = x_1\bar{x}_2x_3x_4\bar{x}_5$. As an example, computing the dominance value is given by $\mathbb{P}_{\mathcal{G}}(o > o') = 0.10867$, the lower bound is: $\mathbb{P}_{\mathcal{G}}^L(o > o') = [1 - \mathbb{P}(\bar{x}_2 > x_2)] \cdot [1 - \mathbb{P}(x_3 > \bar{x}_3|x_1)] \cdot [1 - \mathbb{P}(\bar{x}_5 > x_5|x_4)] = 0.4 \cdot 0.3 \cdot 0.9 = 0.108$. The distance between the lower bound and the true value is 0.00067, which means that the lower bound approximation gives an error of 0.067% on the correct value.

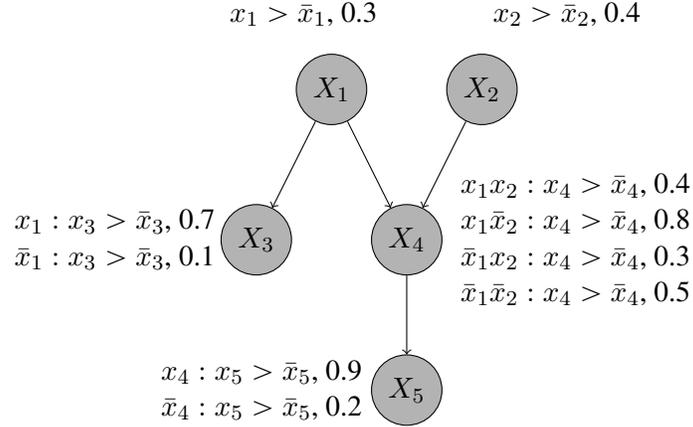


Figure 7: A PCP-net. We show only $\mathbb{P}(x_i > \bar{x}_i)$, since $\mathbb{P}(\bar{x}_i > x_i) = 1.0 - \mathbb{P}(x_i > \bar{x}_i)$.

If we apply the formula for the lower bound to separable PCP-nets, the resulting probability is the same as the actual value of dominance and can be computed in $O(n)$ time.

Theorem 9. Given a separable PCP-net \mathcal{Q} over n features X_1, \dots, X_n with binary domains $\mathcal{D}_i = \{x_i, \bar{x}_i\} \forall i \in \{1, \dots, n\}$ and given two outcomes o and o' then

$$\mathbb{P}(o > o') = \left(\prod_{i \in \text{Diff}^+(o, o')} p_i \right) \cdot \left(\prod_{j \in \text{Diff}^-(o, o')} (1 - p_j) \right)$$

where $p_i = \mathbb{P}(x_i > \bar{x}_i)$ in the PCP-net (and thus $(1 - p_i) = \mathbb{P}(\bar{x}_i > x_i)$) and where Diff^+ and Diff^- are a partition of Diff . Diff^+ is the set of indexes of variables such that $o \upharpoonright_{X_i} = x_i$ and $o' \upharpoonright_{X_i} = \bar{x}_i$ and Diff^- is the set of indexes of variables such that $o \upharpoonright_{X_i} = \bar{x}_i$ and $o' \upharpoonright_{X_i} = x_i$.

Now we can prove the following theorem.

Theorem 10. Given a PCP-net \mathcal{Q} with a separable dependency graph and two outcomes o and o' , then $\mathbb{P}_{\mathcal{Q}}(o > o') = \mathbb{P}_{\mathcal{Q}}^L(o > o')$.

5.3 Upper Bound

We now move our attention to computing an upper bound to the probability that $o > o'$ in a PCP-net \mathcal{Q} . The computation of the upper bound, which we denote with $\mathbb{P}_{\mathcal{Q}}^U(o > o')$, turns out to be more straightforward than the one for the lower bound and it is detailed in Theorem 11.

We say that a feature X in a PCP-net has a *fixed ancestor* over two outcomes o and o' if X is independent (has no parents in the PCP-net graph) or all the variables $Y \in Anc(X)$ (ancestors of X) have the same value in o and o' : $o \upharpoonright_Y = o' \upharpoonright_Y \forall Y \in Anc(X)$. We denote this set of all fixed-ancestor variables as FA. As before, let $Diff$ be the set of the indexes of features that have different values on o and o' .

Definition 17. Given a PCP-net \mathcal{Q} with n variables X_i and given two outcomes o and o' , the dominance upper bound for $o > o'$ in \mathcal{Q} is:

$$\mathbb{P}_{\mathcal{Q}}^U(o > o') = \prod_{j \in (Diff \cap FA)} \mathbb{P}(o \upharpoonright_{X_j} > o' \upharpoonright_{X_j} | u_j)$$

where u_j is the assignment of the parents: $u_j = o \upharpoonright_{Pa(X_j)} = o' \upharpoonright_{Pa(X_j)}$.

The formula above is a product of at most n factors, whose computation takes $O(k)$, hence $O(nk)$ in total (which becomes $O(n)$, since k is bounded).

Theorem 11. Given a PCP-net \mathcal{Q} and two outcomes o and o' , $\mathbb{P}_{\mathcal{Q}}^U(o > o')$ is an upper bound for $\mathbb{P}_{\mathcal{Q}}(o > o')$.

Example 10. Consider a PCP-net \mathcal{G} and two outcomes o and o' as in Example 9. $Diff \cap FA = \{2, 3\}$ as they have different values on o and o' and fixed ancestors. Variable X_5 has different values in o and o' and its ancestor X_2 has different values in o and o' . Thus the upper bound is: $\mathbb{P}_{\mathcal{G}}^U(o > o') = \mathbb{P}(x_2 > \bar{x}_2) \cdot \mathbb{P}(\bar{x}_3 > x_3 | x_1) = 0.4 \cdot 0.3 = 0.12$. We recall that the true probability of dominance is $\mathbb{P}_{\mathcal{G}}(o > o') = 0.10867$ and the lower bound is $\mathbb{P}_{\mathcal{G}}^L(o > o') = 0.108$. Thus, the interval size between the two approximations is 0.012.

5.4 Experimental Evaluation: Dominance

Computing the lower bound and the upper bound gives us an interval in which the true value of the probability of dominance lies. In this section we experimentally explore the quality of our bounds by looking at the size of the interval while varying the number of features and, fixing the number of features, varying the maximal number k of parents that a feature can have.

For all our experiments we randomly generate collections of CP-nets, and PCP-nets. Generating PCP-nets that are independent and identically distributed (i.i.d.) is non-trivial as shown by Allen, Goldsmith, Justice, Mattei, and Raines (2016) and Allen, Goldsmith, Justice, Mattei, and Raines (2017) for CP-nets, a special case of PCP-nets, and to generate the PCP-nets i.i.d. is an interesting question for future work. Therefore we use an approximation method for random generation of a CP-nets:

- *Generating DAG structured CP-nets:* We assume a fixed ordering, X_1, \dots, X_n , of nodes, to conform to \mathcal{O} -legality, and a maximum in-degree, k . We begin by generating the CP-net dependency graph. For each node X_i , we first randomly choose its in-degree (uniformly) between 0 and $\min\{k, i-1\}$. Next, we randomly add all the parents, choosing them from the set of variables $\{X_1, \dots, X_{i-1}\}$ (without replacement, since we cannot add the same parent twice). When the graph is constructed, we fill in the CPTs at random: given a variable X and a parents' assignment u , we fill the corresponding CPT row with an ordering uniformly sampled from the set $\{x > \bar{x}, \bar{x} > x\}$.

$n =$	1	2	3	4	5	6	7
$Pa : n - 1$	0.0000	0.0000	0.0010	0.0020	0.0047	0.0060	0.0047
$Pa : n/2$	0.0000	0.0000	0.0010	0.0016	0.0020	0.0046	0.0040
$Pa : n/4$	0.0000	0.0000	0.0000	0.0008	0.0018	0.0007	0.0014

Table 1: Distance between lower bound and the true probability of dominance.

- *Generating poly-tree structured CP-nets and PCP-nets:* We generate poly-tree structured CP-nets as we did for DAG structured ones, but we generate parents differently. For each node X_i we add parents, one by one, using rejection sampling on X_1, \dots, X_{i-1} , rejecting those that add cycles in the underlying undirected graph, until we reach the in-degree or exhaust the set of possible parents. The CPTs are filled using the same methodology used for DAG structured CP-nets.

We use a similar approximation method for generation PCP-nets:

- *Generating DAG structured PCP-nets:* We generate the PCP-net graphs as we did the CP-net graphs, and then assign probabilities to CPT rows at random: given a variable X , for each parent assignment u we generate a value p uniformly in $[0, 1]$ for the statement $u : x > \bar{x}$, then we assign to $u : \bar{x} > x$ the value $1 - p$.
- *Generating poly-tree structured PCP-nets:* We generate the dependency graph of poly-tree structured PCP-nets as we did for poly-tree structured CP-nets. The CPTs are filled using the same methodology used for DAG structured PCP-nets.

To generate a profile of m CP-nets we use the method described above independently. In all our experiments ties are broken lexicographically, that is, we select the winning element according to the fixed lexicographical order over the elements names.

Figure 8 shows the size of the interval as a function of the number of features, n , and of the number of parents per feature, k . In this experiment we vary $n \in [0, 30]$ and fix the maximum k to $n - 1$, $n/2$ and $n/4$. We compute the mean of the dominance approximation interval over 100 PCP-nets for each set of parameters. For each PCP-net, we take the mean over 100 outcome pairs.²

Turning back to Figure 8, we observe that the mean interval size is small, since it is between 0.1 and 0.2 over a possible maximal interval of size 1. The number of features reaches a plateau around 10 features. We conjecture that this is due to the high number of incompatibilities in the PCP-net. We also note that as the number of parents in the PCP-net grows, the size of the approximation interval increases.

Table 1 and Figure 9 show the distance between our lower bound and the true value of the dominance probability (computed using the formula in Definition 14) as we vary the number of features. In this experiment we vary the number of features $n \in [0, 7]$ and we fix the maximum k to $n - 1$, $n/2$ and $n/4$. We compute the mean over 50 PCP-nets for each value of n where, for each PCP-net we use the mean over 20 outcome pairs. Hence, each point is computed from 1,000 computational trials. We note that the true value of the probability of dominance is very close to

2. We have not included aggregate statistics about run times as the distribution was highly variable. Some instances took seconds, others took days.

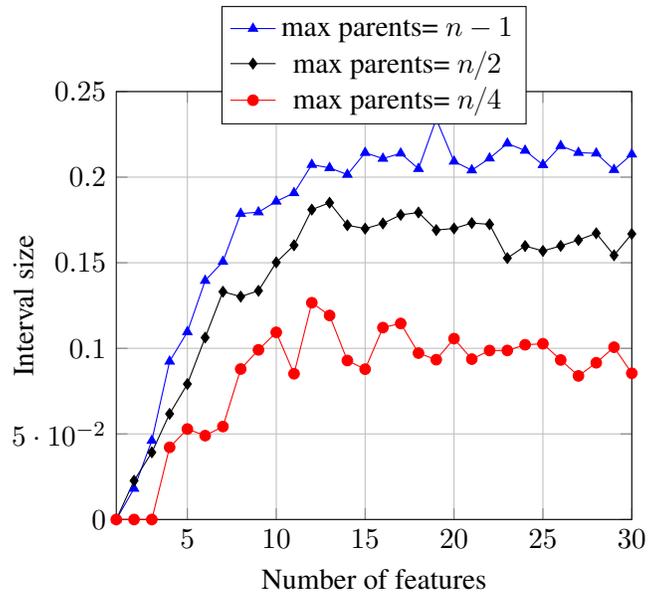


Figure 8: Interval size vs. number of features.

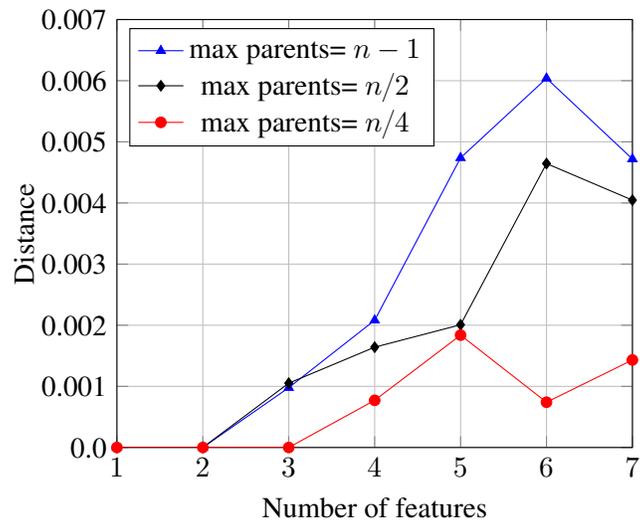


Figure 9: Distance between true probability of dominance and lower bound vs. number of features.

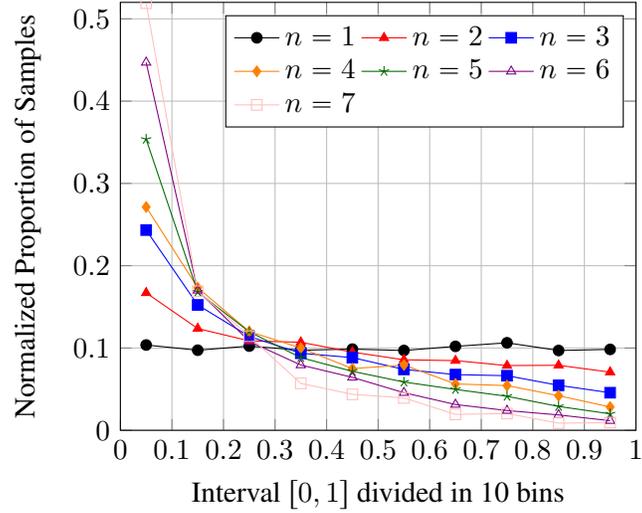


Figure 10: Distribution of the true probability of dominance in the interval $[0, 1]$.

the lower bound since the error of the approximation is never greater than 0.6% with 7 features, although it grows as a function of the number of nodes and parents.

To give more context to the significance of the results shown in Figure 9 we need to understand the distribution of the true value of the probability of dominance to ensure that it is not always very small, and hence trivial to approximate from below. Indeed, the true probability of dominance can occur anywhere in the range $[0, 0.5]$ for $n \leq 7$, see Figure 10. This implies that our lower bound has good accuracy across the possible range of values.

More precisely, in Figure 10 we perform a frequency study of the distribution of the true probability of dominance from Definition 14. We divided the interval $[0, 1]$ in 10 bins and computed the normalized proportion of samples of the real probability of dominance in each bin. We observe that, for a small number of nodes, the distribution is uniform and as the number of features increases, the true value of dominance becomes more frequent in the first bins.

5.5 Dominance as a Decision Problem

In some settings it could be enough to get a yes/no answer from a dominance query, rather than the exact probability value. To do this, it is enough to set a threshold to transform the $[0, 1]$ interval into two values. For example, suppose a choice between two products must be made, based on a profile of agents' CP-nets. In this case, instead of perform a dominance query on each CP-net, we can simply performing a dominance query on the aggregated PCP-net representing the profile (see Section 6 for details on aggregation methods).

Definition 18 (DD for PCP-nets). *Given a PCP-net \mathcal{Q} and a pair of outcomes o and o' , the Decisional Dominance (DD) over o and o' for \mathcal{Q} is defined as follows:*

- if $\mathbb{P}_{\mathcal{Q}}(o > o') > \mathbb{P}_{\mathcal{Q}}(o' > o)$ then $DD_{\mathcal{Q}}(o, o') = True$
- if $\mathbb{P}_{\mathcal{Q}}(o > o') < \mathbb{P}_{\mathcal{Q}}(o' > o)$ then $DD_{\mathcal{Q}}(o, o') = False$
- if $\mathbb{P}_{\mathcal{Q}}(o > o') = \mathbb{P}_{\mathcal{Q}}(o' > o)$ then $DD_{\mathcal{Q}}(o, o') = random\{True, False\}$.

Since dominance is computationally difficult also in CP-nets, this notion of decisional dominance for PCP-nets is computationally difficult as well. We define the following two notions of approximate decisional dominance. One is just standard dominance on the most probable induced CP-nets, while the other one is the lower bound approximation to PCP-net dominance that we described in the previous section.

Definition 19 (MP-DD for PCP-nets). *Given a PCP-net \mathcal{Q} and a pair of outcomes o and o' , and given the most probable CP-net \mathcal{C} induced by \mathcal{Q} , the most probable induced CP-net based decisional dominance for PCP-nets (MP-DD) is defined as follows:*

- if $o \succ_{\mathcal{C}} o'$ then $MP-DD_{\mathcal{Q}}(o, o') = True$
- if $o \prec_{\mathcal{C}} o'$ then $MP-DD_{\mathcal{Q}}(o, o') = False$
- if $o \bowtie_{\mathcal{C}} o'$ then $MP-DD_{\mathcal{Q}}(o, o') = random\{True, False\}$

Definition 20 (LB-DD for PCP-nets). *Given a PCP-net \mathcal{Q} and a pair of outcomes o and o' , the Lower Bound based decisional dominance for PCP-nets (LB-DD) is defined as follows:*

- if $\mathbb{P}_{\mathcal{Q}}^L(o > o') > \mathbb{P}_{\mathcal{Q}}^L(o' > o)$ then $LB-DD_{\mathcal{Q}}(o, o') = True$
- if $\mathbb{P}_{\mathcal{Q}}^L(o > o') < \mathbb{P}_{\mathcal{Q}}^L(o' > o)$ then $LB-DD_{\mathcal{Q}}(o, o') = False$
- if $\mathbb{P}_{\mathcal{Q}}^L(o > o') = \mathbb{P}_{\mathcal{Q}}^L(o' > o)$ then $LB-DD_{\mathcal{Q}}(o, o') = random\{True, False\}$.

Figure 11 shows the frequency of correct answers of MP-DD and LB-DD (in $[0, 1]$), varying the number of nodes. In this experiment we vary the number of nodes $n \in [0, 7]$ and we fix the parents ≤ 2 . We compute the mean over 50 PCP-nets for each value of n and for each PCP-net we take the mean over 20 outcomes pairs, obtaining a mean over 1000 iterations for each value of n . It is easy to see that the lower bound based decisional dominance is very accurate and is much more accurate than MP-DD. Also, LB-DD is polynomially computable, while MP-dominance is not. Moreover, we analyzed if this result changes if we consider only affirmative DD answers or only negative ones. As first result of this study we noticed that the frequencies of affirmative and negative answers is uniform (50% each). Furthermore, we obtained that the affirmative and negative cases are identical and thus correspond to the general case depicted in Figure 11.

6. PCP-nets in a Multiagent Context

Since a PCP-net can be interpreted as a probability distribution over a collection of CP-nets, it is natural to investigate the use of PCP-nets as structures that compactly represent the aggregation of the preferences given by a collection of agents via CP-nets. We have covered the necessary background and notations for discussing multi-agent aggregation of CP-nets and PCP-nets in Section 2.3.

CP-nets have been used in the multiagent systems literature to represent the joint preferences of collections of agents (Rossi et al., 2004; Lukasiewicz & Malizia, 2016; Haret, Novaro, & Grandi, 2018) and as a preference formalism for aggregation using voting techniques (Xia et al., 2011; Li et al., 2010, 2011; Xia et al., 2008; Maran et al., 2013; Mattei et al., 2013). In a real-world scenario, the preference profile of a multiagent system may be composed of hundreds or thousands of individual CP-nets. If the system needed to find out a collective decision of all the agents then

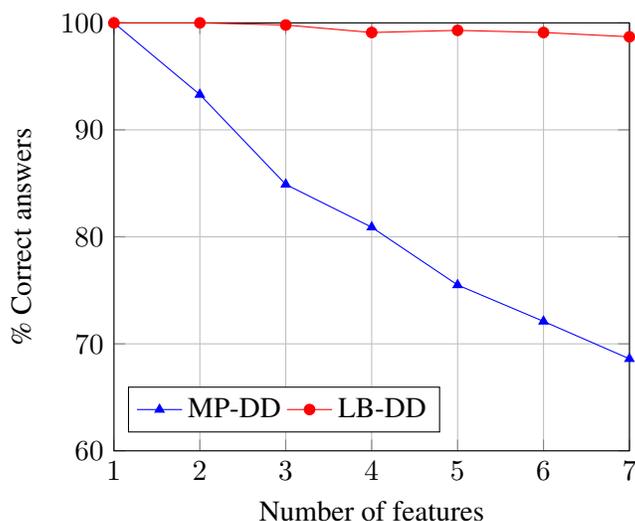


Figure 11: Percentage of right answers of MP-DD and LB-DD vs. number of features.

it may be easier to communicate between agents a single, compact structure instead of the entire profile. Moreover, imposing some reasonable assumption, such as \mathcal{O} -legality (or \mathcal{O} -boundedness) (Lang & Xia, 2009), the aggregation methods can be performed in polynomial time in the size of the profile. In addition, by using a summary or aggregation of the entire preference profile, as has been done in some recommender systems (Adomavicius & Kwon, 2015; Beliakov et al., 2015), we can perform tasks such as deciding optimality and dominance without having to query all the agents again. The probability distribution over the CP-nets can be interpreted as the frequency of each CP-net in the collection as different agents may provide the same CP-net, and the probability values in the PCP-net can be seen as a way to reconcile the possibly different preferences stated in the CP-nets.

In the following sections we restrict to the case in which we have \mathcal{O} -legal profile of acyclic CP-nets with binary features or acyclic PCP-nets with binary features, unless otherwise specified.

6.1 Aggregation Methods

We consider an \mathcal{O} -legal anonymous profile (that is, a collection) of CP-nets over features X_1, \dots, X_n , each with a binary domain. Recall that the *relative frequency* of a CP-net C_i , written as $freq_i$, is the fraction of times C_i appears in the voter population, and that a profile is denoted as $\bar{P} = ((C_1, freq_1), \dots, (C_m, freq_m))$, with $\sum_{i=1}^m freq_i = 1$.

As noted earlier in Theorem 4, given a profile \bar{P} of CP-nets, there may be no PCP-net which induces exactly the same distribution over the CP-nets in \bar{P} . Therefore, we need to define aggregation methods that make sense even when there is no PCP-net that exactly recovers the input profile of CP-nets. The first method we propose (called the Proportion (PR) aggregation method) generates a PCP-net by taking the union of the dependency graphs of the given CP-nets and by determining the probabilities in the PCP-tables from the frequencies of the CP-nets in the profile.

Definition 21 (PR aggregation method). *Given a profile of CP-nets $\bar{P} = (C_i, freq_i)$, the Proportion (PR) aggregation method defines a PCP-net whose dependency graph is the union of the graphs of*

the CP-nets in the profile. Given a variable X and an assignment u to its parents, the probabilities in the PCP-tables are defined as follows:

$$\mathbb{P}(x > \bar{x}|u) = \sum_{C_i: x > \bar{x}|u} \text{freq}_i$$

(where the set $\{C_i : x > \bar{x}|u\}$ denotes the set of CP-nets in the profile that have the CP-row $x > \bar{x}|u$) that is, the probability of the ordering $x > \bar{x}$ for variable X , given assignment u of $\text{Pa}(X)$, is the sum of probabilities of the CP-nets that have that particular ordering over the domain of X , given u . Similarly, $\mathbb{P}(\bar{x} > x|u) = 1 - \mathbb{P}(x > \bar{x}|u)$.

The second method (called the Least Square (LS) aggregation method) minimizes the mean squared error between the probability distribution induced by the PCP-net over the CP-nets given in the input and the relative frequency observed in the input.

Definition 22 (LS aggregation method). Let $\bar{P} = (C_i, \text{freq}_i)$ be a profile of m CP-nets. The Least Square (LS) aggregation method defines a PCP-net whose underlying graph is the union of the graphs of the CP-nets and the probabilities q_j^i in the PCP-tables are those that solve the following problem:

$$\underset{\bar{q} \in [0,1]^r}{\text{argmin}} \sum_{i=1}^m [\text{fp}_{C_i}(\bar{q}) - \text{freq}_i]^2$$

where \bar{q} is the vector of q_j^i ordered lexicographically with i as the first variable, q_j^i is the probability variable of the j -th row of the PCP-table of the variable X_i in the PCP-net, r is the number of PCP-table-rows in the whole PCP-net, C_i are the m CP-nets observed in the profile P and $\text{fp}_{C_i}(\bar{q})$ is the function of probability of the CP-net C_i introduced in Definition 8.

The LS method objective function is a sum of a linear number of components (one for each CP-net in the initial profile), however, evaluating fp_{C_i} may require exponential time as the PCP-net resulting from a generic profile may have an exponential number of cp-statements. The quadratic programming formulation of LS is the following:

$$\underset{\bar{q}}{\text{argmin}} \sum_{i=1}^m [\text{fp}_{C_i}(\bar{q}) - \text{freq}_i]^2$$

$$\forall i : q_i \in [0, 1]$$

We can ensure that the union graph of an \mathcal{O} -legal profile has bounded induced width by assuming the following \mathcal{O} -boundedness condition: for each feature j there are sets $PP(X_j) \subseteq \{X_1, \dots, X_n\}$ of possible parents such that (1) $|PP(X_j)| < k$ for all j and a given constant k , and (2) for all individuals i , $\text{Pa}_i(X_j) \subseteq PP(X_j)$.

Example 11. Given the CP-nets profile in Figure 12, the LS method generates the following:

$$\underset{\bar{q} \in [0,1]^3}{\text{argmin}} [(1 - q_{1_1})(1 - q_{2_1})q_{2_2} - 0.5]^2 + [(1 - q_{1_1})(1 - q_{2_1})(1 - q_{2_2}) - 0.4]^2 + [q_{1_1}q_{2_1}(1 - q_{2_2}) - 0.1]^2.$$

The result corresponds to the PCP-net in Figure 13.

The aggregated PCP-net using PR method is instead shown in Figure 14.

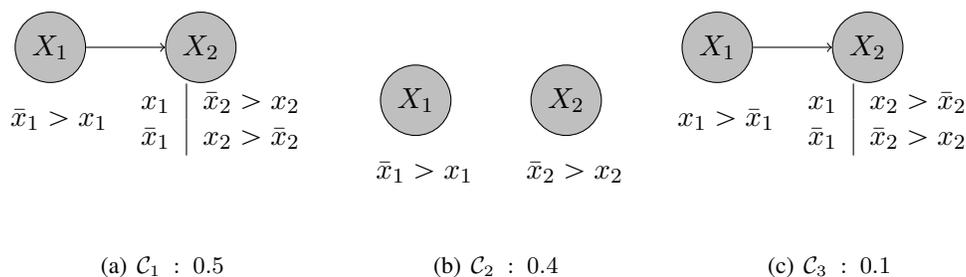


Figure 12: A profile of CP-net with their frequencies.

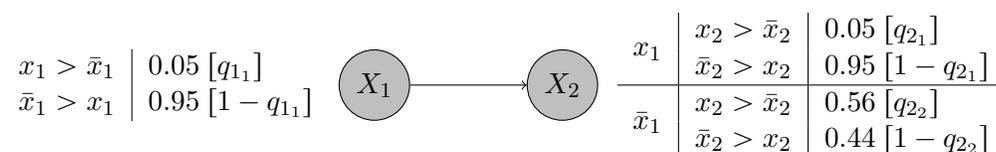


Figure 13: An aggregated PCP-net using LS.

It is interesting to observe that adding a pair $(C_i, 0)$ to a profile of CP-nets will change the LS result. Since LS is the result of a linear program, including $(C_i, 0)$ is actually adding constraints to this linear program, explicitly telling the aggregation procedure that a certain CP-net cannot appear in the output distribution (of induced CP-nets). This can cause problems as the most probable induced CP-net is the one that best summarizes the preferences of the agents. This CP-net should be an aggregation of the preferences of the entire profile, but observe that this CP-net might not appear in the original profile. To ensure that only a CP-net in the original profile is selected, constraints could be added that set all other CP-nets to 0, but this is an exponential program and would not be feasible.

Computing a PCP-net using either *PR* or *LS* may require exponential time as the PCP-net resulting from a generic profile may have an exponential number of cp-statements. However, *PR* also becomes a polynomial method if we have the \mathcal{O} -boundedness condition. As *LS* formalized a notion of “distance to the best” we thought it was a fitting complement to *PR*’s “item by item majority best” method. We wanted two slightly different methods to compare and contrast.

6.2 Optimality in Aggregated CP-nets

Let \mathcal{P} be the set of all anonymous CP-net profiles \bar{P} of m voters over a set of alternatives X . Recall that a CP-voting rule $r : \mathcal{P} \rightarrow X$ is a function that maps each profile P to an alternative $r(P) \in X$. In what follows we assume that CP-voting rules use lexicographic tie-breaking to return a unique winner. The choice and implementation of the tie-breaking rule can have significant impacts on the properties of voting rules and, as is standard in the social choice literature for initial work in an area, we focus on the worst case here, that is, an ordered tie-breaking rule that is known to all agents (for

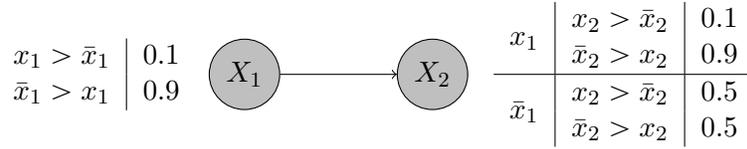


Figure 14: An aggregated PCP-net using PR.

more details see the work of Mattei, Narodytska, and Walsh (2014); Obraztsova and Elkind (2011); Aziz, Gaspers, Mattei, Narodytska, and Walsh (2013), and Obraztsova, Elkind, and Hazon (2011)).

We define four voting rules by combining the two aggregation methods PR and LS presented in Definition 21 and 22 with the two possible ways of extracting an optimal outcome from a PCP-net:

- PR_O : PR and most probable optimal outcome;
- PR_I : PR and optimal outcome of most probable induced CP-net;
- LS_O : LS and most probable optimal outcome;
- LS_I : LS and optimal outcome of most probable induced CP-net.

Computing PR_I is polynomial (if the \mathcal{O} -boundedness condition is met) and computing PR_O is guaranteed to be polynomial only if the graph of the resulting PCP-net has bounded induced width. This result follows from the fact that computing the most probable outcome and the optimal outcome of the most probable induced CP-net are polynomial in the PCP-net description, if it has bounded width. This property can be obtained via the \mathcal{O} -boundedness condition, which also ensures a polynomial complexity to the computation of the PR method.

The four methods may produce different outcomes:

Lemma 5. *There exists a profile of CP-nets \bar{P} such that*

$$\{PR_O(\bar{P}), PR_I(\bar{P})\} \cap \{LS_O(\bar{P}), LS_I(\bar{P})\} = \emptyset;$$

and there exists a profile of CP-nets \bar{P}' such that:

$$PR_O(\bar{P}') \neq PR_I(\bar{P}');$$

and there exists a profile of CP-nets \bar{P}'' such that:

$$LS_O(\bar{P}'') \neq LS_I(\bar{P}'').$$

It is interesting to observe that PR_I returns the same result as the *sequential voting rule with majority* (Lang & Xia, 2009), which consists of applying the majority rule “locally” on each feature of the given CP-nets in the order given by O .

Theorem 12. *Given a profile of CP-nets, PR_I produces the same result as sequential voting with majority.*

Other notions of optimal outcome may be defined. For example, the outcome that maximizes the average number of outcomes worse than it in the induced CP-nets. However, this kind of optimality (and other similar ones) is computationally hard since it exploits the full partial order defined over the outcomes.

6.3 Axiomatic Properties

A first way to compare the four voting rules is by checking if they satisfy the axiomatic properties discussed in Section 2.3. Our results are summarized in Table 2, where empty cells correspond to open problems.

It is easy to see that neutrality holds for all four voting rules. Other results follow from previous work: we can infer that PR_I satisfies consistency and homogeneity, since it coincides with the sequential voting procedure on \mathcal{O} -legal profiles defined and studied by Lang and Xia (2009). For the other results, we provide the following theorem:

Theorem 13. *The following holds:*

- (i) PR_O , LS_O , and LS_I satisfy homogeneity;
- (ii) PR_I and PR_O satisfy participation.
- (iii) PR_I satisfies consensus and PR_O satisfies consensus over a single feature.

In conclusion, the aggregation method PR , generating voting rules PR_O and PR_I , satisfies a good number of desirable axiomatic properties. For LS instead there are some open problems, thus we do not know if it is better or worse than PR from the analysis of its properties. This provides further motivation to the experimental section that follows.

Property	PR_O	PR_I	LS_O	LS_I
Neutrality	✓	✓	✓	✓
Homogeneity	✓	✓	✓	✓
Consistency		✓		
Participation	✓	✓		
Consensus	✓ ³	✓		

Table 2: Axiomatic properties of PR_O , PR_I , LS_O and LS_I .

6.4 Experimental Evaluation: Optimality

In this section we describe our experiments to evaluate the quality of the outcomes returned by the four voting rules. We compare the results of these four voting rules with a baseline we call $PLUR$, which outputs the result of the plurality voting rule applied to the initial profile of CP-nets. $PLUR$ takes the optimal outcome of each CP-net and returns the outcome which is optimal for the largest number of CP-nets. We then compare these five voting rules using a scoring function that is computed using dominance queries on the input profile of CP-nets.

Let F and G be two CP-voting rules, and let T be a set of \mathcal{O} -legal CP-profiles. For ease of presentation, this section considers non-anonymous profiles $P = (C_1, \dots, C_n)$. Given a profile P , we first define three parameters:

- $d_P^>(F, G) = |\{C \in P : F(P) >_C G(P)\}|$,

3. With some restrictions (see Theorem 13).

- $d_P^<(F, G) = |\{C \in P : G(P) >_C F(P)\}|$ and
- $d_P^\times(F, G) = |\{C \in P : F(P) \bowtie_C G(P)\}|$.

Note that $d_P^> + d_P^< + d_P^\times = |P|$. We then define the function $Dom_P(F, G)$ as:

$$Dom_P(F, G) = \begin{cases} 1 & \text{if } d_P^>(F, G) > \max\{d_P^<(F, G), d_P^\times(F, G)\} \\ -1 & \text{if } d_P^<(F, G) > \max\{d_P^>(F, G), d_P^\times(F, G)\} \\ 0 & \text{otherwise.} \end{cases}$$

This function captures the dominance relationship between the results of applying two voting rules F and G . To declare dominance, we require that the number of CP-nets where dominance holds is larger than each of the number of CP-nets (we recall that ties are broken lexicographically) where we have incomparability, and the number of CP-nets where we have dominance in the other direction.

Based on function Dom_P , we define the following notion of pairwise score:

$$PairScore_T(F, G) = \frac{\sum_{P \in T} Dom_P(F, G)}{|T|}.$$

Note that this score belongs to the interval $[-1, 1]$. We are now ready to define the scoring function, inspired by the Copeland voting rule, which gives a score for each voting rule, compared to all others via the *PairScore* score:

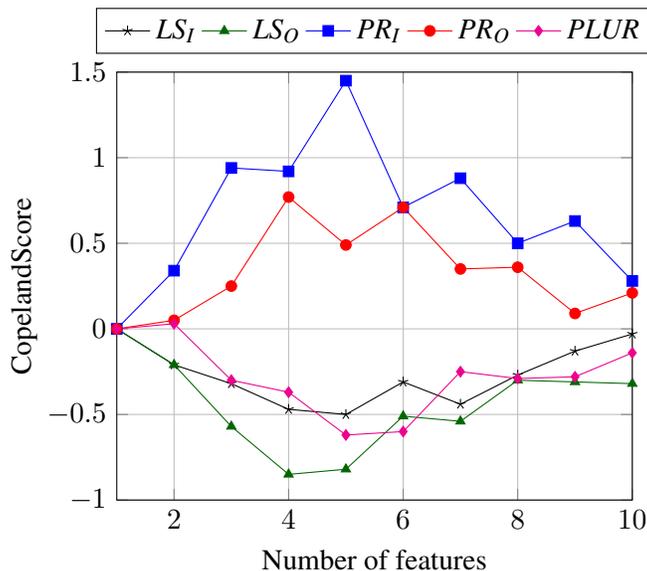
$$CopelandScore_T(F) = \sum_{G \in V \setminus \{F\}} PairScore_T(F, G)$$

where $V = \{PR_O, PR_I, LS_O, LS_I, PLUR\}$. Observe that this score belongs to the interval $[-4, 4]$.

While the choice of scores can have a dramatic effect on the outcome of the experiments, we feel that these three scores are reasonable and give interesting insight into the results. Informally, the *PairScore* method only focuses on the dominance relationship while *CopelandScore* focuses on the decision of the overall aggregation. Comparing *PLUR* to *PairScore* gives us an idea of the power of the dominance relation versus just looking at the top element. *CopelandScore* is designed to compare the different methods on how they do on the overall aggregation. We choose Copeland as the basis for this score as Copeland is qualitative so it is more suitable for qualitative preference formalisms like CP-nets and PCP-nets than other voting rules like Borda that are quantitative. In addition, to evaluate the winner we compute dominance relationships between the pairs of outcomes, hence we want to focus on pairs and Copeland is a popular pairwise voting rule.

Results. To generate \mathcal{O} -legal profiles of CP-nets, and PCP-nets, we follow the same procedure described in Section 5.4. An interesting direction for future work is to generate the profiles of CP-nets using one or more statistical distributions, as discussed by Allen et al. (2017) and often done in experiments for voting rules (Mattei & Walsh, 2017, 2013). However this would first require being able to generate \mathcal{O} -legal CP-nets i.i.d., which is a standing open problem. We show results for some specific parameter values. Similar results were observed also for other values.

In both the experiments shown in Figure 15 and Figure 16, we compute the mean *CopelandScore* over 100 \mathcal{O} -legal profiles of CP-nets and we consider two parameters: the number of features and the number of individuals in the profile. In the experiment in Figure 15 we investigate the quality

Figure 15: *CopelandScore* vs. the number of features.

of all the voting rules, varying the number of features and fixing the number of individuals in the profile. In Figure 16 we analyze the quality of the voting rules, varying the number of individuals in the profile and fixing the number of features.

In the first set of experiments (Figure 15) the profiles have 20 individual CP-nets and the number of features varies from 1 to 10, and each has at most 2 parents. According to the *CopelandScore*, the best voting rule is PR_I , and we observe that PR is consistently better than LS using either the O or I method. PR_I is consistently better than PR_O and LS_I is also better than LS_O . Hence, the most probable optimal outcome (O) is worse than the optimal outcome of the most probable induced CP-net (I) in both the PR and the LS method.

The dominance of the I methods over the O methods is interesting and consistent across the experiments. Looking specifically at PR_I , we think the result is due to the fact that PR_I selects the most representative agent in the profile, that is, the one that best represents all the preferences of all the agents, and then computes the optimal outcome of this agent. While, in contrast, PR_O is selecting the outcome that most frequently appears in the profile as optimal. As each of the agent can have a different optimal outcome, this may spread out the voting and lead us toward fewer “compromise” candidates.

The variance in Figure 15 could be explained by the fact that, with more features, the number of pairs of outcomes that are incomparable increases. There is also a noticeable difference of behavior of the PR method between even and odd numbers of features (and even and odd numbers of individuals in Figure 16). Our conjecture is that an odd number of features (resp. individuals) leads to more decisiveness in the voting rules because of the presence of a smaller number of ties.

In the second set of experiments (Figure 16) the profiles have $n = 3$ and at most $k = 1$ parent per feature, and the number of individual CP-nets varies in $[1, 30]$. We observed that the number of CP-nets in the profile does not significantly influence the *CopelandScore* of the voting rule. We therefore conclude that, in our experimental setting, the PR method is the best one, since it better satisfies the preferences expressed by the CP-nets, according to the scores we defined.

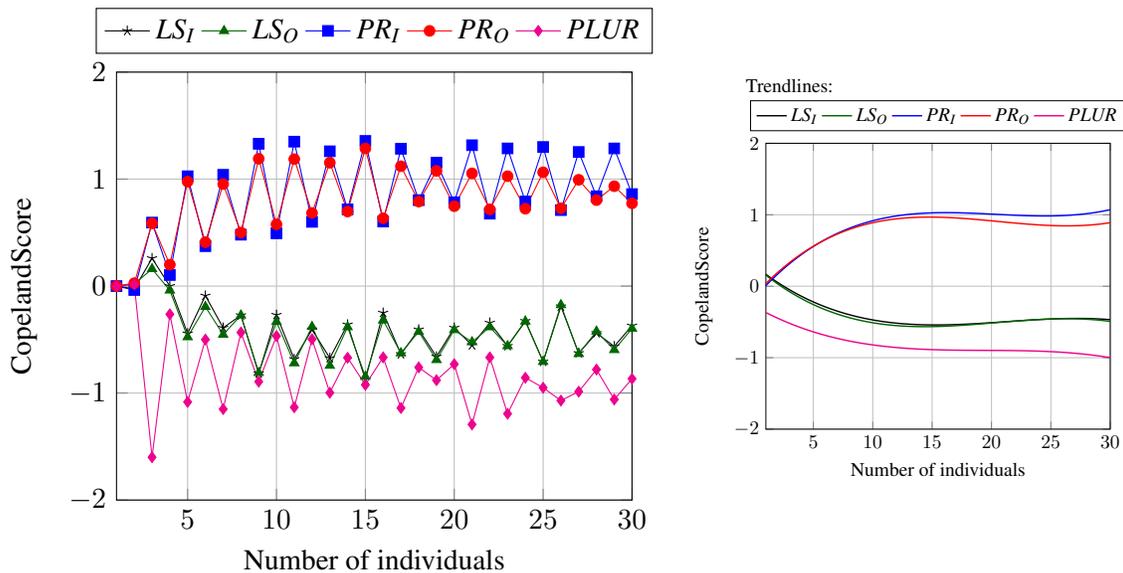


Figure 16: *CopelandScore* vs. the number of individuals from 1 to 30 (and trendlines).

Intuitively, we explain the better performance of the *PR* method with its “majoritarian” nature, that is, it looks at the set of CP-nets and select the cp-statement present in the majority of input CP-nets. Instead, the *LS* method finds a mid-point between the cp-statements in all the CP-nets. If we look at our outcome metrics, *I* is the optimal outcome of the most probable induced CP-net and *O* the most probable optimal outcome of the resulting PCP-net, and both of these are “majoritarian” metrics since they are attempting to pick an outcome with highest probability. Hence these outcome metrics are closer to the philosophy of *PR* as aggregation method, rather than *LS*, which is attempting to minimize the distance to the CP-nets in the input set.

6.5 Dominance on PR

Given that *PR* is the best method for optimality in PCP-nets for multi-agent scenarios, we now analyze both exact and approximate dominance in this setting.

It turns out that *PR* is also accurate in terms of dominance. To show this, we compare the result of dominance tests on random pairs of outcomes in CP-net profiles and their associated PCP-net. Given two outcomes o and o' , the dominance value in the initial profile is the probability mass of the CP-nets that entail $o > o'$. In Table 3, we report the difference between the two values of dominance. Note that this values are in the interval $[0, 1]$ where 1 is the worst value (maximum difference). We can see that the two values of dominance had maximum difference 0.047 when varying the number of features and the number of CP-nets in the profile. Thus, the approximation induced by a PCP-net generated with the *PR* method, given a profile of CP-nets, is accurate both for optimality and for dominance.

n features =	1	2	3	4	5	6
difference	0.0	0.023	0.028	0.037	0.045	0.047

Table 3: Difference between the value of dominance in the initial profile of CP-nets and in the PCP-net aggregated with PR.

7. Conclusions and Future Work

We have defined and shown how to reason with PCP-nets, a generalized version of CP-nets, that can model probabilistic conditional preferences. PCP-nets can be seen as a way to extend CP-nets by including probability information similarly to Bayesian Nets, thus modeling preference and probabilities in one unified structure. We have studied how to reason with PCP-nets in terms of both optimality and dominance. Since dominance is computationally hard, we also provided a tractable approximation which is very accurate when the dependency graph of the PCP-net has a polytree structure. We have also studied the use of PCP-nets as a compact representation structure for expressing the aggregated preferences of a set of agents. Starting from a collection of individual CP-nets, we introduced and evaluated two aggregation methods to obtain a PCP-net, comparing them both theoretically and experimentally. By generating a compact representation of a set of CP-net preferences via a single PCP-net, we are able to perform both optimality and (exact or approximate) dominance reasoning on the given collection of CP-nets.

There are several lines for future work regarding PCP-nets. The first one concerns the possibility of relaxing the independence assumption that is fundamental to CP-nets. In both CP-nets and PCP-nets, the dependencies are among features, and they appear in the dependency graph, but the cp-statements are considered to be independent from each other (Bigot et al., 2013; Cornelio et al., 2013). This means that the preferences of a user over a feature, given an assignment to the parent nodes, are independent from her preferences over another feature (or over the same feature but with another assignment of the parent nodes). This independence assumption appears clearly in the structure of the G-net (see Section 3). The G-net structure expresses the relation between the cp-statements and, in our formulation, has a separable graph (that is, without edges), while the graph of a PCP-net has edges between the features. Thus, in PCP-nets, we are considering all the cp-statements independent. In many scenarios it makes sense to stick to this independence assumption: when we are modelling noise in choices, hidden external factors, or when data comes from several different individuals, such as in a multi-agent scenario, and we want to treat the incoming preferences as independent. However, there are also other scenarios where it makes sense to remove this independence assumption. By doing this, the G-net will model a non-trivial relation between the cp-statements, since it will admit edges between its nodes, representing the connection between cp-statements. For example, we may observe that there is a strong correlation among the probability distributions of the preferences over the same variable (for different values of its parents), or among the probability distributions associated to preferences defined over different variables (e.g., often people who prefer fish over meat also prefer wine over beer). Moreover we want expand the profiles of CP-nets that we consider to include different generation methods and correlations.

The second line of possible future work concerns learning PCP-nets starting from examples defined by pairs of outcomes. We already have initial results on learning PCP-nets with only in-

dependent features, but since a separable structure is not always compatible with the input data, we intend to define methods to also learn non-separable PCP-nets, which are more expressive. We believe that using pairs of outcomes (rather than optimal outcomes, as done by Bigot et al. (2014)) as training data makes the collection of training data much easier, since it is easier for people to compare two outcomes rather than to recognise if an outcome is optimal (Allen et al., 2015). This claim is also supported by an extensive literature on learning from pairs (Chevaleyre et al., 2011; Lang & Mengin, 2009). A possible approach to this learning problem is to follow a methodology similar to the one in the work of Bigot et al. (2014) (where however PCP-nets are learnt from optimal outcomes), in which first we learn the structure of the PCP-net and then, once the structure is fixed, we learn the probabilities in the PCP-tables.

To ensure good computational complexity properties of the various reasoning tasks in (P)CP-nets, we also plan to combine the PCP-net framework with machine learning approaches (Li, 2011), with the intent to learn the dominance relation. Finally, another interesting direction for future work would be performing more empirical experiments comparing the similarity, for example, overlap of the structure, of a learned PCP-net to a CP-net profile. Another experiment could compare the results from the LS and PR (as well as others) approaches starting from a noisy, “ground-truth” PCP-net. However, doing this would require also understanding how to efficiently and statistically accurately sample from a PCP-net (Allen et al., 2017).

Acknowledgments

Judy Goldsmith’s work was partially supported by NSF grant IIS-1649152. The work of Cristina Cornelio and Francesca Rossi was partially supported by the University of Padova project “Incorporating patients’ preferences in kidney transplant decision protocols”. The work of Cristina Cornelio was partially done while at IBM Research. Nicholas Mattei’s work was partially supported by NICTA, funded by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Centre of Excellence Program. The work of Nicholas Mattei was also partially done while at IBM Research. We would also like to thank the anonymous referees whose careful feedback made this a better paper than it had been.

Portions of this paper were included in three conference publications: the first introduced the notion of PCP-nets at the Australasian Joint Conference on Artificial Intelligence (Cornelio et al., 2013), the second introduced the idea of using PCP-nets in a multiagent context at the International Workshop on Computational Social Choice (Cornelio et al., 2014), and the third more clearly formalized this multiagent view while adding experiments and tighter bounds at the International Conference on Autonomous Agents and Multiagent Systems (Cornelio et al., 2015).

References

- Adomavicius, G., & Kwon, Y. (2015). Multi-criteria recommender systems. In Ricci, F., Rokach, L., & Shapira, B. (Eds.), *Recommender Systems Handbook*, pp. 847–880. Springer.
- Ahmed, S., & Mouhoub, M. (2018a). Representation and reasoning with probabilistic tcp-nets. *Computer and Information Science*, 11, 9.
- Ahmed, S., & Mouhoub, M. (2018b). Representation and reasoning with probabilistic tcp-nets.. *Comput. Inf. Sci.*, 11(4), 9–28.

- Ahmed, S. U. (2020). *Conditional Preference Networks: Constraints, Genuine Decision, and Aggregation*. Ph.D. thesis, Faculty of Graduate Studies and Research, University of Regina.
- Allen, T. E., Goldsmith, J., Justice, H. E., Mattei, N., & Raines, K. (2017). Uniform random generation and dominance testing for cp-nets. *Journal of Artificial Intelligence Research*, 59, 771–813.
- Allen, T., Chen, M., Goldsmith, J., Mattei, N., Popova, A., Regenwetter, M., Rossi, F., & Zwilling, C. (2015). Beyond theory and data in preference modeling: Bringing humans into the loop. In *Proceedings of the 4th International Conference on Algorithmic Decision Theory (ADT)*.
- Allen, T., Goldsmith, J., Justice, H., Mattei, N., & Raines, K. (2016). Generating CP-nets uniformly at random. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*.
- Amor, N. B., Dubois, D., Gouider, H., & Prade, H. (2016). Graphical models for preference representation: An overview. In *Proceedings of the 10th International Scalable Uncertainty Management (SUM 2016)*, pp. 96–111.
- Arrow, K. J., Sen, A. K., & Suzumura, K. (2002). *Handbook of Social Choice and Welfare*. North-Holland.
- Aziz, H., Gaspers, S., Mattei, N., Narodytska, N., & Walsh, T. (2013). Ties matter: Complexity of manipulation when tie-breaking with a random vote. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence (AAAI)*, pp. 74–80.
- Beliakov, G., Calvo, T., & James, S. (2015). Aggregation of preferences in recommender systems. In Ricci, F., Rokach, L., & Shapira, B. (Eds.), *Recommender Systems Handbook*, pp. 777–808. Springer.
- Bigot, D., Fargier, H., Mengin, J., & Zanuttini, B. (2013). Probabilistic conditional preference networks. In *Proc. of the 29th International Conference on Uncertainty in Artificial Intelligence (UAI)*.
- Bigot, D., Mengin, J., & Zanuttini, B. (2014). Learning probabilistic cp-nets from observations of optimal items. In *7th Starting AI Researcher Symposium (STAIRS 2014)*, pp. 81–90.
- Bistarelli, S., Montanari, U., & Rossi, F. (1997). Semiring-based constraint satisfaction and optimization. *Journal of the ACM (JACM)*, 44(2), 201–236.
- Boutilier, C., Bacchus, F., & Brafman, R. I. (2001). UCP-networks: A directed graphical representation of conditional utilities. In *Proceedings of the 17th International Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 56–64.
- Boutilier, C., Brafman, R., Domshlak, C., Hoos, H., & Poole, D. (2004). CP-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements. *Journal of Artificial Intelligence Research (JAIR)*, 21, 135–191.
- Brafman, R., & Domshlak, C. (2002). Introducing variable importance tradeoffs into cp-nets. In *UAI 02*, pp. pages 69–76 (cited on pages 26, 40).
- Brafman, R., & Domshlak, C. (2009). Preference handling—an introductory tutorial. *AI Magazine*, 30(1), 58.
- Brams, S., Kilgour, D., & Zwicker, W. (1998). The paradox of multiple elections. In *Soc Choice Welfare*.

- Brandt, F., Conitzer, V., Endriss, U., Lang, J., & Procaccia, A. D. (Eds.). (2016). *Handbook of Computational Social Choice*. Cambridge University Press.
- Chevaleyre, Y., Endriss, U., Lang, J., & Maudet, N. (2008). Preference handling in combinatorial domains: From AI to social choice. *AI Magazine*, 29(4), 37.
- Chevaleyre, Y., Koriche, F., Lang, J., Mengin, J., & Zanuttini, B. (2011). *Preference Learning*, chap. Learning Ordinal Preferences on Multiattribute Domains: The Case of CP-nets. Springer Berlin Heidelberg.
- Cornelio, C., Goldsmith, J., Mattei, N., Rossi, F., & Venable, K. (2013). Updates and uncertainty in CP-nets. In *Proceedings of the 26th Australasian Joint Conference on Artificial Intelligence (AUSAI)*.
- Cornelio, C., Grandi, U., Goldsmith, J., Mattei, N., Rossi, F., & Venable, K. (2014). Voting with CP-nets using a probabilistic preference structure. In *Proceedings of the 5th International Workshop on Computational Social Choice (COMSOC)*.
- Cornelio, C., Grandi, U., Goldsmith, J., Mattei, N., Rossi, F., & Venable, K. (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)*.
- D’Ambrosio, B. (1999). Inference in Bayesian networks. *AI Magazine*, 20(2), 21.
- Dechter, R. (1999). Bucket elimination: A unifying framework for reasoning. *Artificial Intelligence*, 113(1-2), 41–85.
- Domshlak, C., & Brafman, R. (2002). CP-nets: Reasoning and consistency testing. In *Proc. 8th International Conference on Principles of Knowledge Representation and Reasoning (KR)*.
- Faltings, B., Torrens, M., & Pu, P. (2004). Solution generation with qualitative models of preferences. *Computational Intelligence*, 20(2), 246–263.
- Fidha, S. E., & Amor, N. B. (2015). Probabilistic weighted CP-nets. In *Intelligent Decision Technologies: Proceedings of the 7th KES International Conference on Intelligent Decision Technologies (KES-IDT 2015)*, pp. 159–169.
- Fürnkranz, J., & Hüllermeier, E. (2010). *Preference Learning: An Introduction*. Springer.
- Goldsmith, J., Lang, J., Truszczynski, M., & Wilson, N. (2008). The computational complexity of dominance and consistency in CP-nets. *Journal of Artificial Intelligence Research (JAIR)*, 33(1), 403–432.
- Gonzales, C., & Perny, P. (2004). GAI networks for utility elicitation.. In *Proc. of the 10th International Conference on Principles of Knowledge Representation and Reasoning (KR)*.
- Haret, A., Novaro, A., & Grandi, U. (2018). Preference aggregation with incomplete cp-nets. In *Proceedings of the 16th International Conference on Principles of Knowledge Representation and Reasoning (KR)*.
- Lang, J., Pini, M., Rossi, F., Venable, K., & Walsh, T. (2007). Winner determination in sequential majority voting. In *Proc. of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*.
- Lang, J., & Xia, L. (2009). Sequential composition of voting rules in multi-issue domains. *Mathematical Social Sciences*, 57(3), 304–324.

- Lang, J., & Mengin, J. (2009). The complexity of learning separable ceteris paribus preferences. In *Proceedings of 21st International Joint Conference on Artificial Intelligence (IJCAI)*.
- Li, H. (2011). A short introduction to learning to rank. *IEICE Transactions*, 94-D(10), 1854–1862.
- Li, M., Vo, Q., & Kowalczyk, R. (2010). An efficient procedure for collective decision-making with CP-nets. In *Proc. of the 19th European Conference on Artificial Intelligence (ECAI)*.
- Li, M., Vo, Q., & Kowalczyk, R. (2011). Majority-rule-based preference aggregation on multi-attribute domains with CP-nets. In *Proc. of the 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*.
- Loreggia, A., Mattei, N., Rossi, F., & Venable, K. B. (2018a). On the distance between CP-nets. In *Proceedings of the 17th International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*.
- Loreggia, A., Mattei, N., Rossi, F., & Venable, K. B. (2018b). Value alignment via tractable preference distance. In Yampolskiy, R. V. (Ed.), *Artificial Intelligence Safety and Security*. CRC Press.
- Loreggia, A., Mattei, N., Rossi, F., & Venable, K. B. (2019). Cpmetric: Deep siamese networks for metric learning on structured preferences. In Seghrouchni, A. E. F., & Sarne, D. (Eds.), *IJCAI 2019 International Workshops, Revised Selected Best Papers*, Vol. 12158 of *Lecture Notes in Computer Science*, pp. 217–234. Springer.
- Lukasiewicz, T., Martinez, M., & Simari, G. (2014). Probabilistic preference logic networks.. In *Proceedings of the 21st European Conference on Artificial Intelligence (ECAI 2014)*, pp. 561–566.
- Lukasiewicz, T., & Malizia, E. (2016). On the Complexity of mCP-nets. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI-2016)*.
- Maran, A., Maudet, N., Pini, M., Rossi, F., & Venable, K. (2013). A framework for aggregating influenced CP-nets and its resistance to bribery. In *Proc. of the 27th AAAI Conference on Artificial Intelligence (AAAI)*.
- Marden, J. I. (1995). *Analyzing and Modeling Rank Data*. CRC Press.
- Mattei, N., Narodytska, N., & Walsh, T. (2014). How hard is it to control an election by breaking ties?. In *Proceedings of the 21st European Conference on Artificial Intelligence (ECAI)*, pp. 1067–1068.
- Mattei, N., Pini, M. S., Rossi, F., & Venable, K. B. (2012). Bribery in voting over combinatorial domains is easy. In *Proceedings of the 11th International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*.
- Mattei, N., Pini, M. S., Rossi, F., & Venable, K. B. (2013). Bribery in voting with CP-nets. *Annals of Mathematics and Artificial Intelligence*, 68(1–3), 135–160.
- Mattei, N., & Walsh, T. (2013). PrefLib: A library for preferences, [HTTP://WWW.PREFLIB.ORG](http://www.preflib.org). In *Proceedings of the 3rd International Conference on Algorithmic Decision Theory (ADT)*.
- Mattei, N., & Walsh, T. (2017). A PREFLIB.ORG Retrospective: Lessons Learned and New Directions. In Endriss, U. (Ed.), *Trends in Computational Social Choice*, chap. 15, pp. 289–309. AI Access Foundation.

- Obraztsova, S., & Elkind, E. (2011). On the complexity of voting manipulation under randomized tie-breaking. In *Proceedings of 22nd International Joint Conference on Artificial Intelligence IJCAI*, pp. 319–324.
- Obraztsova, S., Elkind, E., & Hazon, N. (2011). Ties matter: Complexity of voting manipulation revisited. In *Proceedings of the 10th International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pp. 71–78.
- Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.
- Pigozzi, G., Tsoukias, A., & Viappiani, P. (2015). Preferences in artificial intelligence. *Annals of Mathematics and Artificial Intelligence*, 77(3), 361–401.
- Price, R., & Messinger, P. R. (2005). Optimal recommendation sets: Covering uncertainty over user preferences. In *Proc. 20th AAAI Conference on Artificial Intelligence*, pp. 541–548.
- Pu, P., Faltings, B., Chen, L., Zhang, J., & Viappiani, P. (2011). Usability guidelines for product recommenders based on example critiquing research. In Ricci, F., Rokach, L., Shapira, B., & Kantor, P. B. (Eds.), *Recommender Systems Handbook*, pp. 511–545. Springer.
- Regenwetter, M., Dana, J., & Davis-Stober, C. (2011). Transitivity of preferences. *Psychological Review*, 118(1), 42.
- Regenwetter, M., Dana, J., & Davis-Stober, C. P. (2010). Testing transitivity of preferences on two-alternative forced choice data. *Frontiers in psychology*, 1, 148.
- Regenwetter, M., & Davis-Stober, C. P. (2012). Behavioral variability of choices versus structural inconsistency of preferences. *Psychological Review*, 119(2), 408.
- Rossi, F., & Mattei, N. (2019). Building ethically bounded AI. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*.
- Rossi, F., Venable, K., & Walsh, T. (2004). mCP-nets: representing and reasoning with preferences of multiple agents. In *Proc. of the 19th AAAI Conference on Artificial Intelligence (AAAI)*.
- Rossi, F., Venable, K., & Walsh, T. (2011). *A Short Introduction to Preferences: Between Artificial Intelligence and Social Choice*. Morgan & Claypool Publishers.
- Roth, A., & Kagel, J. (1995). *The Handbook of Experimental Economics*, Vol. 1. Princeton University Press Princeton.
- Santhanam, G. R., Basu, S., & Honavar, V. (2010). Dominance testing via model checking.. In *Proc. of the 24th AAAI Conference on Artificial Intelligence (AAAI)*.
- Sikdar, S., Adali, S., & Xia, L. (2017). Optimal decision making with CP-nets and PCP-nets. In *Proceedings of the 16th International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*.
- Wang, H., Zhang, J., Sun, W., Song, H., Guo, G., & Zhou, X. (2012). WCP-nets: A weighted extension to cp-nets for web service selection. In *Service-Oriented Computing: 10th International Conference (ICSOC 2012)*, pp. 298–312.
- Xia, L., Conitzer, V., & Lang, J. (2008). Voting on multiattribute domains with cyclic preferential dependencies.. In *Proc. of the 23rd AAAI Conference on Artificial Intelligence (AAAI)*.

Xia, L., Conitzer, V., & Lang, J. (2011). Hypercubewise preference aggregation in multi-issue domains. In *Proc. of the 22nd International Joint Conference on Artificial Intelligence (IJCAI)*.

Appendix A. Proofs

Theorem 1. *Given a PCP-net Q with features with binary domains, the probability of an edge $e = (X, Y)$ can be computed as follows:*

$$\mathbb{P}(e) = 1 - \sum_{\sigma \in \Upsilon} \left(\prod_{v \in \mathcal{E}} \mathbb{P}[\sigma(v)|vx] \mathbb{P}[\sigma(v)|v\bar{x}] \right)$$

where $\Upsilon = \{\sigma : \mathcal{E} \rightarrow \{y > \bar{y}, \bar{y} > y\}\}$ is the set of all possible functions σ that associate to each element $v \in \mathcal{E}$ an element of the set $\{y > \bar{y}, \bar{y} > y\}$ (the set of all the possible total orders of the domain of Y).

Proof. We equivalently prove that the probability of nonexistence of the edge $e = (X, Y)$ is:

$$\sum_{\sigma \in \Upsilon} \left(\prod_{v \in \mathcal{E}} \mathbb{P}[\sigma(v)|vx] \mathbb{P}[\sigma(v)|v\bar{x}] \right).$$

To prove the formula, we have to reason about the particular configurations of the PCP-table of Y that induce CP-nets that do not have the edge e .

A node X is not a parent of Y exactly when (by definition of a parent) for each configuration v of the other parents, the preference over Y is the same for vx and for $v\bar{x}$. Given σ , a mapping from each element $v \in \mathcal{E}$ to a total order over the domain of Y , this happens with probability

$$\mathbb{P}[\sigma(v)|vx] \mathbb{P}[\sigma(v)|v\bar{x}]$$

for each configuration v . Since all the rules are independent (per definition of a PCP-net), this happens with probability:

$$\prod_{v \in \mathcal{E}} \mathbb{P}[\sigma(v)|vx] \mathbb{P}[\sigma(v)|v\bar{x}]$$

for each σ . Since different σ 's concern different CP-nets, this happens with probability

$$\sum_{\sigma \in \Upsilon} \left(\prod_{v \in \mathcal{E}} \mathbb{P}[\sigma(v)|vx] \mathbb{P}[\sigma(v)|v\bar{x}] \right)$$

overall. □

Theorem 2. *The probability of the edge $e = (X, Y)$ can be computed as follows:*

$$\mathbb{P}(e) = 1 - \sum_{\sigma \in \Upsilon} \left(\prod_{v \in \mathcal{E}} \mathbb{P}[\sigma(v)|vx_1] \mathbb{P}[\sigma(v)|vx_2] \cdots \mathbb{P}[\sigma(v)|vx_d] \right)$$

where $D_X = \{x_1, x_2, \dots, x_d\}$, $\Theta(Y)$ is the set of all the possible total orders of the domain of Y and $\Upsilon = \{\sigma : \mathcal{E} \rightarrow \Theta(Y)\}$ is the set of all possible functions σ that associate to each element $v \in \mathcal{E}$ a total order over the domain of Y (an element of the set $\Theta(Y)$).

Proof. The proof follows step by step the proof of the binary case. □

Theorem 3. *Given a PCP-net \mathcal{Q} and its set of induced CP-nets, then:*

1. *The probability of realizing one of its induced CP-nets is the joint probability of the corresponding assignment in the G-net for \mathcal{Q} ;*
2. *The probability distribution defined in Definition 8 over the set of induced CP-nets is a well defined discrete probability distribution (i.e., the probabilities over the domain elements sum to one and the probability of each element belongs to the interval $[0, 1]$).*
3. *The most probable induced CP-nets correspond to the CP-nets associated (using the one-to-one correspondence between induced CP-nets and assignment of the corresponding G-nets described above) to the assignment of the G-net for \mathcal{Q} , with maximal joint probability.*

Proof.

1. The first statement of this theorem follows from the fact that we take exactly the product of the chosen rows in the PCP-tables.
2. The probability defined in the first statement is computed as a product of non-negative factors, thus it is non-negative. Moreover, the sum of the probabilities of all the CP-nets in the set of the induced CP-nets is equal to 1, because there's a 1-1 correspondence between the assignments of the G-net with positive (not zero) probability and the induced CP-nets, and the sum of the probabilities of all the assignments of a BN is equal to 1.
3. This statement follows directly from the first one. □

Theorem 4. *Given a probability distribution over a set of CP-nets (even if they have the same dependency graph), there may exist no PCP-net inducing it.*

Proof. Consider the following two CP-nets over two variables X_1 and X_2 :

- $(C_1, 0.5)$: $(x_1 > \bar{x}_1)$ and $(x_2 > \bar{x}_2)$
- $(C_2, 0.5)$: $(\bar{x}_1 > x_1)$ and $(\bar{x}_2 > x_2)$.

The PCP-net representing such a collection of CP-nets must satisfy the following system of equations, where $\bar{q} = (q_1, q_2)$, q_1 is the probability of $x_1 > \bar{x}_1$, q_2 is the probability of $x_2 > \bar{x}_2$:

$$\begin{cases} fp_{C_1}(\bar{q}) = q_1 q_2 \\ fp_{C_2}(\bar{q}) = (1 - q_1)(1 - q_2) \end{cases} \Rightarrow \begin{cases} fp_{C_1}(\bar{q}) = 0.5 \\ fp_{C_2}(\bar{q}) = 0.5. \end{cases}$$

This system has no solution for $\bar{q} \in [0, 1]^2$ (it has only complex solutions). □

Theorem 5. *Given a probability distribution over a set of CP-nets defined over the same features, if this distribution is generated by a PCP-net \mathcal{Q} , then we can write a system of non-linear equations that provides a unique solution that uniquely recovers the PCP-net \mathcal{Q} .*

Proof. It is equivalent to prove that, if the system generated by a probability distribution over a set of CP-nets has a solution, it is unique.

We consider a PCP-net and then, to compute the probabilities, we consider the G-net associated. We suppose that the G-net has n variables X_1, X_2, \dots, X_n with domains $D_1 = \{a_j^1\}$: for $j = 1$ to

$k_1\}, \dots, D_n = \{a_j^n: \text{for } j = 1 \text{ to } k_n\}$ (the values of a_j^i can be either an order, or 0 or 1). We define p_j^i as the probability $\mathbb{P}(X_i = a_j^i)$.

Our system of equations (by definition) contains the numerical values of all the joint probabilities of every assignment to the n variables.

We suppose now that there are two distinct solutions p_j^{i*} and $p_j^{i\diamond}$ ($\forall i = 1, \dots, n$ and $\forall j = 1, \dots, k_i$).

We know that p_j^i is equal to:

$$\sum_{t_1=1}^{k_1} \dots \sum_{t_{i-1}=1}^{k_{i-1}} \sum_{t_{i+1}=1}^{k_{i+1}} \dots \sum_{t_n=1}^{k_n} \mathbb{P}(X_1 = a_{t_1}^1, \dots, X_{i-1} = a_{t_{i-1}}^{i-1}, X_i = a_j^i, X_{i+1} = a_{t_{i+1}}^{i+1}, \dots, X_n = a_{t_n}^n).$$

But all the numerical values of the joint probabilities are known (for all assignments of the n variables), so all the probabilities:

$$\mathbb{P}(X_1 = a_{t_1}^1, \dots, X_{i-1} = a_{t_{i-1}}^{i-1}, X_i = a_j^i, X_{i+1} = a_{t_{i+1}}^{i+1}, \dots, X_n = a_{t_n}^n)$$

are known as well.

Let us consider the value of p_j^i , which we call it b_j^i . Now since both p_j^{i*} and $p_j^{i\diamond}$ are equal to b_j^i , they coincide.

This allows us to conclude that there is a unique solution. \square

Theorem 6. *Given a PCP-net \mathcal{Q} and its Opt-net:*

1. *There is a one-to-one correspondence between the assignments (with non-zero probability) of the Opt-net and the outcomes that are optimal in at least one induced CP-net of \mathcal{Q} .*
2. *Given a PCP-net \mathcal{Q} , the probability that an outcome is optimal is the joint probability of the corresponding assignment in the Opt-net (we consider only the assignments of the Opt-net with positive probability). If no such corresponding assignment exists, then the probability of being optimal is 0.*
3. *To find the most probable optimal outcome for a PCP-net \mathcal{Q} , it is sufficient to compute the assignment that maximizes the joint probability on the Opt-net.*

Proof.

1. True by construction.
2. By construction, the set of assignments of the Opt-net is a subset of those of \mathcal{Q} . By the definition of the Opt-net, if an assignment of \mathcal{Q} is not an assignment of the Opt-net then it cannot be optimal in any induced CP-net.

Let us now focus only on the subset of assignments of \mathcal{Q} that have a corresponding assignment in the Opt-net. Let $x = (x_1, x_2, \dots, x_n)$ be one of these assignments. We denote by $P_{opt}(x) = \mathbb{P}(X_1 = x_1, \dots, X_n = x_n)$ the joint probability of x in the Opt-net. We recall that the probability that x is optimal in the PCP-net is the sum of the probabilities of the induced CP-nets that have assignment x as optimal. We call this probability $P_{cp}(x)$. Thus we want to prove that $P_{opt}(x) = P_{cp}(x)$.

Let us consider A_x , the set of induced CP-nets that have x as their optimal assignment. Thus we have:

$$P_{cp}(x) = \sum_{\mathcal{C} \in A_x} \mathbb{P}(\mathcal{C}) .$$

To compute the optimal value for an induced CP-net \mathcal{C} , the sweep-forward procedure does not consider all the rows of the CP-tables, but only a subset of them. For example in the CP-net with two variables Y and Z with the cp-tables: $y_1 > y_2$, $y_1 : z_1 > z_2 > z_3$ and $y_2 : z_2 > z_3 > z_1$, the sweep-forward procedure will consider only the first two and not $y_2 : z_2 > z_3 > z_1$. So, we can split the CP-tables of \mathcal{C} into two parts, one affecting the choice of the optimal outcome (denoted with \mathcal{C}_+) and one not used by the sweep-forward procedure (denoted with \mathcal{C}_-). In the example above we will have $\mathcal{C}_+ = \{y_1 > y_2, y_1 : z_1 > z_2 > z_3\}$ and $\mathcal{C}_- = \{y_2 : z_2 > z_3 > z_1\}$.

Since each row in the CP-tables of an induced CP-net is associated with a probability (the probability of that row in the PCP-net) and contributes to define the probability of the induced CP-net, we can divide the induced CP-net probability in two independent parts: one is the product of the rows that are used in the sweep-forward procedure (the product of the probabilities of the rows belonging to \mathcal{C}_+ , that we will denote as $\mathbb{P}(\mathcal{C}_+)$) and one as the product of the rows that are not used in the sweep-forward procedure (the product of the probabilities of the rows belonging to \mathcal{C}_- , that we will denote as $\mathbb{P}(\mathcal{C}_-)$):

$$\mathbb{P}(\mathcal{C}) = \mathbb{P}(\mathcal{C}_+)\mathbb{P}(\mathcal{C}_-) .$$

Thus we have that:

$$P_{cp}(x) = \sum_{\mathcal{C} \in A_x} \mathbb{P}(\mathcal{C}) = \sum_{\mathcal{C} \in A_x} \mathbb{P}(\mathcal{C}_+)\mathbb{P}(\mathcal{C}_-) .$$

We observe now that the optimal outcome x can be produced in many different ways, as there can be many different orderings that produce the same result (all these orderings belonging to \mathcal{C}_+ and not to \mathcal{C}_-). For example, considering an independent variable Y , the orderings $y_1 > y_2 > y_3$ and $y_1 > y_3 > y_2$ both produce the optimal value y_1 , but in different ways. For this reason we consider a partition $(A_{x,1}, \dots, A_{x,t})$ of the set A_x (the set of induced CP-nets that produce the same optimal outcome x) such that each induced CP-net belonging to the same equivalence class produces the same optimal outcome in the same way. This means that given two CP-nets \mathcal{D} and \mathcal{E} that belong to the same equivalence class $A_{x,i}$, they have the same rows that actively affect the choice of the optimal value and differ only in the remaining part, namely $\mathcal{D}_+ = \mathcal{E}_+$ and $\mathcal{D}_- \neq \mathcal{E}_-$.

We now define as \mathcal{C}_+^i the part that is equal for all the members of the equivalence class $A_{x,i}$. That means:

$$\mathcal{C}_+^i = \bigcap_{\mathcal{C} \in A_{x,i}} \mathcal{C}_+ = \mathcal{C}_+ \quad \forall \mathcal{C} \in A_{x,i} .$$

Then we can rewrite the probability $P_{cp}(x)$ as:

$$P_{cp}(x) = \sum_{i=1}^t \sum_{\mathcal{C} \in A_{x,i}} [\mathbb{P}(\mathcal{C}_+)\mathbb{P}(\mathcal{C}_-)] = \sum_{i=1}^t \sum_{\mathcal{C} \in A_{x,i}} [\mathbb{P}(\mathcal{C}_+^i)\mathbb{P}(\mathcal{C}_-)] = \sum_{i=1}^t [\mathbb{P}(\mathcal{C}_+^i) \cdot \sum_{\mathcal{C} \in A_{x,i}} \mathbb{P}(\mathcal{C}_-)] .$$

We observe now that:

$$\sum_{C \in A_{x,i}} \mathbb{P}(C_-) = 1 \quad \forall i = 1, \dots, t$$

since we are summing the probabilities of all possible cases (the entire distribution) in the part C_- (the part not influencing the sweep-forward procedure). Thus, the probability becomes

$$P_{cp}(x) = \sum_{i=1}^t \mathbb{P}(C_+^i).$$

We notice now that the joint probability in the Opt-net $\mathbb{P}(X_1 = x_1, \dots, X_n = x_n)$ corresponds exactly to $\sum_{i=1}^t \mathbb{P}(C_+^i)$ by contraction. We recall that we built the rows of the probability tables for the variables X_1, \dots, X_n by summing the probabilities of the orderings that have the same head (first ranked value) in the PCP-net \mathcal{Q} . This is the same as summing the probabilities $\mathbb{P}(C_+^i)$ over the subset $A_{x,i}$.

Thus we proved that $P_{cp}(x) = P_{opt}(x)$.

3. This statement is an immediate consequence of the previous two. \square

Lemma 1. *Given an acyclic CP-net C with n nodes and two outcomes $o = o_1 o_2 \dots o_n$ and $o' = o'_1 o'_2 \dots o'_n$ such that $o > o'$ and given i an index in $\{1, \dots, n\}$ such that $o_i \neq o'_i$ and $o_j = o'_j \forall j < i$ then*

$$o \upharpoonright_{X_1 \dots X_j} > o' \upharpoonright_{X_1 \dots X_j} \quad \forall j \geq i.$$

Proof. Because of $o > o'$, there exists a worsening path from o to o' : $P = o, o^1, o^2, \dots, o'$. We prove that $P \upharpoonright_{X_1, \dots, X_j} = o \upharpoonright_{X_1, \dots, X_j}, o^1 \upharpoonright_{X_1, \dots, X_j}, \dots, o' \upharpoonright_{X_1, \dots, X_j}$ is a worsening path from $o \upharpoonright_{X_1, \dots, X_j}$ to $o' \upharpoonright_{X_1, \dots, X_j}$ (considering only non-repeated outcomes in the projected space $\{X_1, \dots, X_j\}$, thus removing from $P \upharpoonright_{X_1, \dots, X_j}$ the repetition steps in which $o^i \upharpoonright_{X_1, \dots, X_j} = o^{i+1} \upharpoonright_{X_1, \dots, X_j}$). We have to prove that each step of $P \upharpoonright_{X_1, \dots, X_j}$ is a worsening flip. Considering a flip of the value of variable X_l , the fact that it is a worsening step does not depend on the values of X_l 's children or of the values of the nodes that are independent of X_l . All the nodes X_{j+1}, \dots, X_n are either children or nodes (conditionally) independent of X_l . Thus, removing X_l does not change the preferences for the outcomes restricted to the variables X_1, \dots, X_j . Thus, each worsening flip $o^i > o^{i+1}$ in the path P , it is also a worsening flip in the path $P \upharpoonright_{X_1, \dots, X_j}$ (unless $o^i \upharpoonright_{X_1, \dots, X_j} = o^{i+1} \upharpoonright_{X_1, \dots, X_j}$). We proved that each step in the path $P \upharpoonright_{X_1, \dots, X_j}$ obtained from P and projecting on the variables X_1, \dots, X_j without repetitions is a worsening path from $o \upharpoonright_{X_1 \dots X_j}$ to $o' \upharpoonright_{X_1 \dots X_j}$. \square

Lemma 2. *Given an acyclic CP-net C with n nodes and two outcomes o and o' , if there exists an index $i \in \{1, \dots, n\}$ such that $o \upharpoonright_{X_1 \dots X_i} \bowtie o' \upharpoonright_{X_1 \dots X_i}$ then we have that $o \upharpoonright_{X_1 \dots X_j} \bowtie o' \upharpoonright_{X_1 \dots X_j}$ for each index j between i and n ($i \leq j \leq n$).*

Proof. We prove the lemma by contradiction. Suppose that $o \upharpoonright_{X_1 \dots X_j} \bowtie o' \upharpoonright_{X_1 \dots X_j}$, then either $o \upharpoonright_{X_1 \dots X_j} > o' \upharpoonright_{X_1 \dots X_j}$ or $o \upharpoonright_{X_1 \dots X_j} < o' \upharpoonright_{X_1 \dots X_j}$. We suppose $o \upharpoonright_{X_1 \dots X_j} > o' \upharpoonright_{X_1 \dots X_j}$ (the other case is analogous). Since $o \upharpoonright_{X_1 \dots X_j} > o' \upharpoonright_{X_1 \dots X_j}$ there exists a worsening path P from $o \upharpoonright_{X_1 \dots X_j}$ to $o' \upharpoonright_{X_1 \dots X_j}$ and $P = o \upharpoonright_{X_1 \dots X_j}, o_1 \upharpoonright_{X_1 \dots X_j}, \dots, o' \upharpoonright_{X_1 \dots X_j}$. From Lemma 1 we have that $\forall k \in$

$\{1, \dots, j\} o \upharpoonright_{X_1, \dots, X_k} \geq o \upharpoonright_{X_1, \dots, X_k}$ because the path $P \upharpoonright_{X_1, \dots, X_k} = o \upharpoonright_{X_1, \dots, X_k}, o_1 \upharpoonright_{X_1, \dots, X_k}, \dots, o' \upharpoonright_{X_1, \dots, X_k}$ is a worsening path from $o \upharpoonright_{X_1, \dots, X_k}$ to $o' \upharpoonright_{X_1, \dots, X_k}$ in the sub-CP-net of the nodes X_1, \dots, X_k , in particular for $k = i$. Thus, $o \upharpoonright_{X_1 \dots X_i} > o' \upharpoonright_{X_1 \dots X_i}$. That is a contradiction. \square

Lemma 3. *Given an acyclic CP-net C with n nodes and two outcomes o and o' , then $o \bowtie o'$ if and only if there exists index $i \in \{1, \dots, n\}$ such that $o \upharpoonright_{X_1 \dots X_j} \bowtie o' \upharpoonright_{X_1 \dots X_j} \forall j < i$ and $o \upharpoonright_{X_1 \dots X_i} \bowtie o' \upharpoonright_{X_1 \dots X_i} \forall l \geq i$.*

Proof. \Leftarrow Consequence of Lemma 2.

\Rightarrow We suppose $o \bowtie o'$ and prove that exists index i such that

$$o \upharpoonright_{X_1 \dots X_j} \bowtie o' \upharpoonright_{X_1 \dots X_j} \quad \forall j < i \quad \text{and}$$

$$o \upharpoonright_{X_1 \dots X_l} \bowtie o' \upharpoonright_{X_1 \dots X_l} \quad \forall l \geq i$$

by induction on the number of variables.

Base case: The base case for induction considers two variables A and B (in the case of 1 variable, $i = 1$). In the case in which there is an edge between them, there are no incomparable outcomes. Thus we consider only the cases in which there is no edge between the two variables. We have four cases for the CP-tables:

- $a > \bar{a}$ and $b > \bar{b}$, which implies that $\bar{a}\bar{b} \bowtie \bar{a}b$
- $a > \bar{a}$ and $\bar{b} > b$, which implies that $\bar{a}\bar{b} \bowtie ab$
- $\bar{a} > a$ and $b > \bar{b}$, which implies that $\bar{a}\bar{b} \bowtie ab$
- $\bar{a} > a$ and $\bar{b} > b$, which implies that $\bar{a}\bar{b} \bowtie \bar{a}b$.

In each we obtain $i = 2$.

Induction step: Supposing the lemma holds for n variables, we prove it for $n + 1$.

We consider a CP-net with $n + 1$ variables in which $o \bowtie o'$. We consider $o \upharpoonright_{X_1, \dots, X_n}$ and $o' \upharpoonright_{X_1, \dots, X_n}$. We can have three cases:

1. $o \upharpoonright_{X_1, \dots, X_n} > o' \upharpoonright_{X_1, \dots, X_n}$
2. $o \upharpoonright_{X_1, \dots, X_n} < o' \upharpoonright_{X_1, \dots, X_n}$
3. $o \upharpoonright_{X_1, \dots, X_n} \bowtie o' \upharpoonright_{X_1, \dots, X_n}$

In Case 3 we can apply the induction hypothesis. Cases 1 and 2 are symmetric so we will just prove Case 1. We observe that $o \upharpoonright_{X_1, \dots, X_n} \neq o' \upharpoonright_{X_1, \dots, X_n}$, because otherwise the two outcome o and o' are a flipping pair, which are never incomparable.

We consider the case in which $o \bowtie o'$ and $o \upharpoonright_{X_1, \dots, X_n} > o' \upharpoonright_{X_1, \dots, X_n}$. We have to prove that $\forall j < n$ $o \upharpoonright_{X_1, \dots, X_j} > o' \upharpoonright_{X_1, \dots, X_j}$. We prove it by contradiction.

- Suppose that $\exists j < n$ such that $o \upharpoonright_{X_1, \dots, X_j} \bowtie o' \upharpoonright_{X_1, \dots, X_j}$. We apply Lemma 2 and we obtain that $o \upharpoonright_{X_1, \dots, X_n} \bowtie o' \upharpoonright_{X_1, \dots, X_n}$ is also a contradiction.

- Suppose that $\exists j < n$ such that $o \upharpoonright_{X_1, \dots, X_j} < o' \upharpoonright_{X_1, \dots, X_j}$. Thus, there is a worsening path P from $o \upharpoonright_{X_1, \dots, X_n}$ to $o' \upharpoonright_{X_1, \dots, X_n}$. Considering P restricted to the variables X_1, \dots, X_j we obtain \bar{P} that is a worsening path from $o \upharpoonright_{X_1, \dots, X_j}$ to $o' \upharpoonright_{X_1, \dots, X_j}$. That is a contradiction. \square

Lemma 4. *Given an acyclic CP-net C with n nodes formed by two connected components⁴ C_1 and C_2 , and given two different outcomes o and o' such that $[(o \upharpoonright_{C_1} > o' \upharpoonright_{C_1} \text{ and } o \upharpoonright_{C_2} < o' \upharpoonright_{C_2}) \text{ or } (o \upharpoonright_{C_1} < o' \upharpoonright_{C_1} \text{ and } o \upharpoonright_{C_2} > o' \upharpoonright_{C_2})]$ then $o \bowtie o'$.*

Proof. We prove only the case in which $o \upharpoonright_{C_1} > o' \upharpoonright_{C_1}$ and $o \upharpoonright_{C_2} < o' \upharpoonright_{C_2}$, because the other one is symmetric.

We suppose by contradiction that $o \bowtie o'$. Then we can have either $o > o'$ or $o < o'$. We suppose $o > o'$ (the other case is analogous). Since $o > o'$ there exists a worsening path P from o to o' , $P = o, o_1, o_2, \dots, o'$. Each step o_i of the path correspond to a worsening flip of a variable X_j that belongs either to C_1 or to C_2 . Thus, the two sub-paths P_1 and P_2 such that:

$$P_1 = P \upharpoonright_{C_1} = o \upharpoonright_{C_1}, o_1 \upharpoonright_{C_1}, o_2 \upharpoonright_{C_1}, \dots, o' \upharpoonright_{C_1}$$

$$P_2 = P \upharpoonright_{C_2} = o \upharpoonright_{C_2}, o_1 \upharpoonright_{C_2}, o_2 \upharpoonright_{C_2}, \dots, o' \upharpoonright_{C_2}$$

are paths (with repetition) such that P_1 is a worsening path from $o \upharpoonright_{C_1}$ to $o' \upharpoonright_{C_1}$ in CP-net C_1 and P_2 is a worsening path from $o \upharpoonright_{C_2}$ to $o' \upharpoonright_{C_2}$ in CP-net C_2 . We obtain that $o \upharpoonright_{C_1} > o' \upharpoonright_{C_1}$ and $o \upharpoonright_{C_2} > o' \upharpoonright_{C_2}$. That is a contradiction. \square

Theorem 7. *Given a polytree-structured CP-net \mathcal{C} with n variables and two outcomes o and o' , if for all X_i in Diff (see Definition 15) there exists an assignment $u_i \in U_i^{o, o'}$ (see Definition 15) such that $u_i : o \upharpoonright_{X_i} > o' \upharpoonright_{X_i}$ then $o > o'$.*

Proof. Let's order the variables such that for each variable X_i then $X_j \in \text{Pa}(X_i)$ implies that $j < i$. Such an order always exists since the CP-net is polytree-shaped. Moreover, it is possible that there exists more than one. We will take a random order ω between those.

We prove by contradiction that $o > o'$. We have to consider two cases: $o < o'$ and $o \bowtie o'$:

- Let's suppose $o < o'$.

Considering the first variable $X_{\tilde{i}}$ in Diff following the order ω , from Lemma 1 and $o < o'$ we have that $o \upharpoonright_{X_1 \dots X_j} < o' \upharpoonright_{X_1 \dots X_j} \forall j \geq \tilde{i}$. In particular $o \upharpoonright_{X_1 \dots X_{\tilde{i}}} < o' \upharpoonright_{X_1 \dots X_{\tilde{i}}}$.

We notice that $U_{\tilde{i}}^{o, o'}$ has only one element: $U_{\tilde{i}}^{o, o'} = \{o \upharpoonright_{\text{Pa}(X_{\tilde{i}})} = o' \upharpoonright_{\text{Pa}(X_{\tilde{i}})}\}$. Thus, for all the assignments in $U_{\tilde{i}}^{o, o'}$ (only one in this case: $u_{\tilde{i}}$) we have that $u_{\tilde{i}} : o \upharpoonright_{X_{\tilde{i}}} > o' \upharpoonright_{X_{\tilde{i}}}$. This is in contradiction with our hypothesis, since we proved that there exists a variable X_i in Diff such that for all the assignments $u_i \in U_i^{o, o'}$ the following does not hold: $u_i : o \upharpoonright_{X_i} > o' \upharpoonright_{X_i}$.

- Let's now suppose $o \bowtie o'$.

By Lemma 3 we have that there exists an index $\tilde{i} \in \{1, \dots, n\}$ such that

$$o \upharpoonright_{X_1 \dots X_j} \bowtie o' \upharpoonright_{X_1 \dots X_j} \forall j < \tilde{i} \text{ and } o \upharpoonright_{X_1 \dots X_l} \bowtie o' \upharpoonright_{X_1 \dots X_l} \forall l \geq \tilde{i}.$$

4. Connected components are subgraphs in which any two vertices are connected to each other by an undirected paths; there are no edges connecting nodes belonging to different connected components.

Let's consider the case in which \bowtie is equal to $>$.

We do not consider the other case (with \bowtie equal to $<$) because this would lead to an immediate contradiction.⁵

So, given that \bowtie is equal to $>$, we can suppose that $o \upharpoonright_{X_1 \dots X_{\tilde{i}-1}} > o' \upharpoonright_{X_1 \dots X_{\tilde{i}-1}}$.⁶

We will prove now that also $o \upharpoonright_{X_1 \dots X_{\tilde{i}}} > o' \upharpoonright_{X_1 \dots X_{\tilde{i}}}$, which is a contradiction to our hypothesis of $o \upharpoonright_{X_1 \dots X_{\tilde{i}}} \bowtie o' \upharpoonright_{X_1 \dots X_{\tilde{i}}}$.

Let's consider two cases assuming that $X_{\tilde{i}} \in \text{Diff}$:⁷ the case in which $X_{\tilde{i}}$ is an independent node and the case in which it is a dependent node.

- $X_{\tilde{i}}$ is an **independent node**.

Let's consider a worsening path P from $o \upharpoonright_{X_1, \dots, X_{\tilde{i}-1}}$ to $o' \upharpoonright_{X_1, \dots, X_{\tilde{i}-1}}$:

$$P = o \upharpoonright_{X_1, \dots, X_{\tilde{i}-1}} \rightarrow O_1 \rightarrow O_2 \rightarrow \dots \rightarrow o' \upharpoonright_{X_1, \dots, X_{\tilde{i}-1}}.$$

Then it is easy to find a worsening path from $o \upharpoonright_{X_1, \dots, X_{\tilde{i}}}$ to $o' \upharpoonright_{X_1, \dots, X_{\tilde{i}}}$ given the assumption that for all X_i in Diff there exists an assignment $u_i \in U_i^{o, o'}$ such that $u_i : o \upharpoonright_{X_i} > o' \upharpoonright_{X_i}$. In this case $X_{\tilde{i}} \in \text{Diff}$. and $U_{\tilde{i}}^{o, o'} = \emptyset$ and thus $o \upharpoonright_{X_{\tilde{i}}} > o' \upharpoonright_{X_{\tilde{i}}}$. Thus one possible worsening path would be:

$$\begin{aligned} \tilde{P} &= o \upharpoonright_{X_1, \dots, X_{\tilde{i}-1}} o \upharpoonright_{X_{\tilde{i}}} \rightarrow O_1 o \upharpoonright_{X_{\tilde{i}}} \rightarrow O_2 o \upharpoonright_{X_{\tilde{i}}} \rightarrow \dots \\ &\dots \rightarrow o' \upharpoonright_{X_1, \dots, X_{\tilde{i}-1}} o \upharpoonright_{X_{\tilde{i}}} \rightarrow o' \upharpoonright_{X_1, \dots, X_{\tilde{i}-1}} o' \upharpoonright_{X_{\tilde{i}}}. \end{aligned}$$

leading to the contradiction that $o \upharpoonright_{X_1 \dots X_{\tilde{i}}} > o' \upharpoonright_{X_1 \dots X_{\tilde{i}}}$.

- Now we consider the case in which $X_{\tilde{i}}$ is **not an independent node**.

Because \mathcal{C} is polytree-structured, the sub-CP-net $\tilde{\mathcal{C}}$ of the nodes $X_1, \dots, X_{\tilde{i}-1}$ is formed by $|Pa(X_{\tilde{i}})|$ connected components,⁸ each one containing exactly one parent of $X_{\tilde{i}}$.

Without lack of generality, we can assume that we picked the order ω that has all the siblings of $X_{\tilde{i}}$ appearing after \tilde{i} in the order.

In this case, for each $u \in U_{\tilde{i}}^{o, o'}$ we can have a worsening path P from $o \upharpoonright_{X_1, \dots, X_{\tilde{i}-1}}$ to $o' \upharpoonright_{X_1, \dots, X_{\tilde{i}-1}}$ that contains u . This is because the parents belong to different connected components and we can permute, in the worsening path, the order in which we make the worsening flips of the variables in the single connected components (as a consequence of Lemma 4 and the fact that all the siblings of $X_{\tilde{i}}$ appearing after \tilde{i} in the order ω).

5. For the first variable in Diff ($X_{\tilde{i}}$, following the order ω) we have that there exists an assignment $u_i \in U_i^{o, o'}$ such that $u_i : o \upharpoonright_{X_i} > o' \upharpoonright_{X_i}$. Since $X_{\tilde{i}}$ is the first variable to change in the order then $u_i = o \upharpoonright_{Pa(X_{\tilde{i}})} = o' \upharpoonright_{Pa(X_{\tilde{i}})}$. Moreover since $X_{\tilde{i}}$ is the first variable to change in the order, this implies that also $o \upharpoonright_{X_1 \dots X_{\tilde{i}}} > o' \upharpoonright_{X_1 \dots X_{\tilde{i}}}$. This is in contradiction with $o \upharpoonright_{X_1 \dots X_{\tilde{i}-1}} < o' \upharpoonright_{X_1 \dots X_{\tilde{i}-1}}$ by the application of Lemma 1.

6. For Lemma 1 we have also that $o \upharpoonright_{X_1 \dots X_j} \geq o' \upharpoonright_{X_1 \dots X_j} \forall j < \tilde{i}$. Thus, we obtain that: $o \upharpoonright_{X_1 \dots X_j} > o' \upharpoonright_{X_1 \dots X_j} \forall j < \tilde{i}$ and $o \upharpoonright_{X_1 \dots X_l} \bowtie o' \upharpoonright_{X_1 \dots X_l} \forall l \geq \tilde{i}$.

7. We make this assumption because the other case would lead to an immediate contradiction: $X_{\tilde{i}} \notin \text{Diff}$ implies that $o \upharpoonright_{X_{\tilde{i}}} = o' \upharpoonright_{X_{\tilde{i}}}$ and thus since $o \upharpoonright_{X_1 \dots X_{\tilde{i}-1}} > o' \upharpoonright_{X_1 \dots X_{\tilde{i}-1}}$ also $o \upharpoonright_{X_1 \dots X_{\tilde{i}}} > o' \upharpoonright_{X_1 \dots X_{\tilde{i}}}$.

8. We recall that connected components are subgraphs in which any two vertices are connected to each other by an undirected paths; there are no edges connecting nodes belonging to different connected components.

Thus for each $u \in U_{\tilde{i}}^{o, o'}$ there exists a worsening path P containing u :

$$P = o \upharpoonright_{X_1, \dots, X_{\tilde{i}-1}} \rightarrow o_1 \rightarrow o_2 \rightarrow \dots \rightarrow o_l \rightarrow \dots \rightarrow o' \upharpoonright_{X_1, \dots, X_{\tilde{i}-1}}.$$

Now consider $u_{\tilde{i}}$ such that $u_{\tilde{i}} : o \upharpoonright_{X_{\tilde{i}}} > o' \upharpoonright_{X_{\tilde{i}}}$, which exists by hypothesis.

Let's suppose that o_l in the worsening path P contains $u_{\tilde{i}}$: $o_l \upharpoonright_{Pa(X_{\tilde{i}})} = u_{\tilde{i}}$.

Therefore, we can create a worsening path from $o \upharpoonright_{X_1, \dots, X_{\tilde{i}}}$ to $o' \upharpoonright_{X_1, \dots, X_{\tilde{i}}}$ as follows:

$$\begin{aligned} \tilde{P} &= o \upharpoonright_{X_1, \dots, X_{\tilde{i}-1}} o \upharpoonright_{X_{\tilde{i}}} \rightarrow o_1 o \upharpoonright_{X_{\tilde{i}}} \rightarrow o_2 o \upharpoonright_{X_{\tilde{i}}} \rightarrow \dots \\ &\dots \rightarrow o_l o \upharpoonright_{X_{\tilde{i}}} \rightarrow o_l o' \upharpoonright_{X_{\tilde{i}}} \rightarrow \dots \rightarrow o' \upharpoonright_{X_1, \dots, X_{\tilde{i}-1}} o' \upharpoonright_{X_{\tilde{i}}} \end{aligned}$$

leading to the contradiction that $o \upharpoonright_{X_1 \dots X_{\tilde{i}}} > o' \upharpoonright_{X_1 \dots X_{\tilde{i}}}$.

Thus we can conclude that $o > o'$. □

Theorem 8. *Given a PCP-net \mathcal{Q} and two outcomes o and o' , $\mathbb{P}_{\mathcal{Q}}^L(o > o')$ is a lower bound for $\mathbb{P}_{\mathcal{Q}}(o > o')$.*

Proof.

Intuitively the lower bound value is just the probability that a randomly generated induced CP-net (from the PCP-net \mathcal{Q}) satisfies the condition expressed in Theorem 7. Let's now prove it formally.

We can rewrite the lower-bound formula (Equation 1) as:

$$\mathbb{P}_{\mathcal{Q}}^L(o > o') = (1 - \bar{\rho}_0)(1 - \bar{\rho}_1) \cdots (1 - \bar{\rho}_m) \quad (2)$$

where $j = 1, \dots, m$ are indices that belongs to $Diff$ (set of m indices ($m \leq n$) of variables that change value from o to o') and are ordered as in ω (an ordering of the PCP-net variables such that for each variable X_i , if $X_j \in Pa(X_i)$ then $j < i$).

Equation 2 is equivalent to:

$$\begin{aligned} \mathbb{P}_{\mathcal{Q}}^L(o > o') &= (1 - \bar{\rho}_0) - [(1 - \bar{\rho}_0)(\bar{\rho}_1)] - [(1 - \bar{\rho}_0)(1 - \bar{\rho}_1)(\bar{\rho}_2)] - \dots - \\ &\quad - [(1 - \bar{\rho}_0)(1 - \bar{\rho}_1) \cdots (1 - \bar{\rho}_{m-1})(\bar{\rho}_m)]. \end{aligned}$$

Let's consider the first term $1 - \bar{\rho}_0$ (we rename this term q for simplicity). We will see now that q corresponds to the following probability:

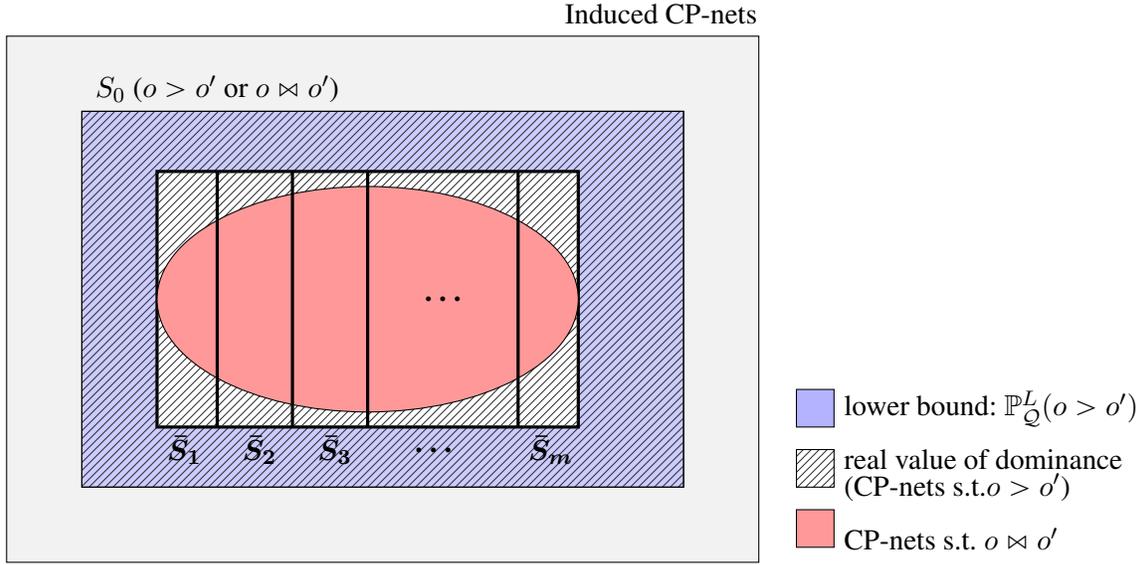
$$q = \mathbb{P}(o \upharpoonright_{X_{d(0)}} > o' \upharpoonright_{X_{d(0)}}) \quad \text{if } X_{d(0)} \text{ is independent} \quad (3)$$

$$q = \mathbb{P}(o \upharpoonright_{X_{d(0)}} > o' \upharpoonright_{X_{d(0)}} | u) \quad \text{if } X_{d(0)} \text{ is not independent} \quad (4)$$

where $d(\cdot)$ is a function such that $d : Diff \rightarrow \{1, \dots, n\}$ and $d(j)$ is the corresponding index in $\{1, \dots, n\}$ of the index $j \in Diff$ following the order ω .

This is because we notice that (from Definition 15):

- if the variable $X_{d(0)}$ is independent then: $\bar{\rho}_0 = \mathbb{P}(o' \upharpoonright_{X_{d(0)}} > o \upharpoonright_{X_{d(0)}})$, so $(1 - \bar{\rho}_0) = \mathbb{P}(o \upharpoonright_{X_{d(0)}} > o' \upharpoonright_{X_{d(0)}})$.


 Figure 17: Sets S_0 and $\bar{S}_1 \dots, \bar{S}_m$ in the proof of Theorem 8

- if $X_{d(0)}$ is dependent on a set of parents $Pa(X_{d(0)})$, then the set $U_{d(0)}^{o, o'}$ has only one element $u = o \upharpoonright_{Pa(X_{d(0)})} = o' \upharpoonright_{Pa(X_{d(0)})}$. Thus $\bar{\rho}_0 = \mathbb{P}(o' \upharpoonright_{X_{d(0)}} > o \upharpoonright_{X_{d(0)}} | u)$ and $(1 - \bar{\rho}_0) = \mathbb{P}(o \upharpoonright_{X_{d(0)}} > o' \upharpoonright_{X_{d(0)}} | u)$.

Then Formula 2 is equivalent to:

$$\mathbb{P}_{\mathcal{Q}}^L(o > o') = q - q\bar{\rho}_1 - q(1 - \bar{\rho}_1)(\bar{\rho}_2) - \dots - q(1 - \bar{\rho}_1) \dots (1 - \bar{\rho}_{m-1})(\bar{\rho}_m) \quad (5)$$

where $j = 1, \dots, m$ are indices that belong to *Diff*. We note that each term of the sum contains q . Considering $o = (x_1 x_2 \dots x_n)$ and $o' = (y_1 y_2 \dots y_n)$, the probability q is the probability in \mathcal{Q} that $x_{d(0)} > y_{d(0)}$ (or $x_{d(0)} > y_{d(0)} | u$) and corresponds to the probability that an induced CP-net has that row in their CP-tables. Let's call S_0 the set of all the induced CP-nets that have that row in their CP-tables. We can think of q as the probability of an induced CP-net belonging to the set S_0 . For simplicity, in what follows, we will refer to q as the probability of the set S_0 ⁹ (and we will denote it as $\mathbb{P}(S_0)$).

We notice that when we restrict to the set S_0 , we restrict our attention to the induced CP-nets for which either $o > o'$ or $o \bowtie o'$.¹⁰ Moreover S_0 contains all the induced CP-nets that have $o > o'$ (this follows from Lemma 1), but only some of the induced CP-nets that have $o \bowtie o'$ (since there could exist induced CP-nets with $o \bowtie o'$ that have $x_{d(0)} < y_{d(0)}$ – or $x_{d(0)} < y_{d(0)} | u$).

Definition 23. We define now two different types of sets:

- S_j : a set of induced CP-nets with probability $\mathbb{P}(S_j) = q(1 - \bar{\rho}_1) \dots (1 - \bar{\rho}_{j-1})(1 - \bar{\rho}_j)$. This set includes all the CP-nets with $x_{d(i)} > y_{d(i)}$ (or $x_{d(i)} > y_{d(i)} | u$) for all the $i \leq j$ and, for

9. The probability $\mathbb{P}(S)$ of a set S of induced CP-nets corresponds to the sum of the probabilities of the CP-nets belonging to that set.

10. It is not possible that a CP-net in S_0 has $o < o'$ because all of the CP-nets in S_0 contain the CPT row that says that $x_{d(0)} > y_{d(0)}$ (or $x_{d(0)} > y_{d(0)} | u$). This follows from Lemma 1.

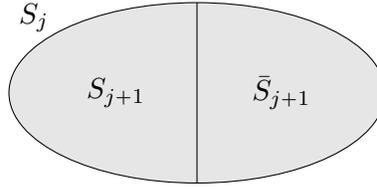


Figure 18: Partition of set S_j into S_{j+1} and \bar{S}_{j+1} in the proof of Theorem 8

the other indexes l not in Diff the set includes both the CP-net with $x_l > y_l$ (or $x_l > y_l|u$) and with the opposite $x_l < y_l$ (or $x_l < y_l|u$), in all possible combinations.

- \bar{S}_j : a set of induced CP-nets with probability $\mathbb{P}(\bar{S}_j) = (q)(1 - \bar{\rho}_1) \cdots (1 - \bar{\rho}_{j-1})(\bar{\rho}_j)$. This set includes all the CP-nets with $x_{d(i)} > y_{d(i)}$ (or $x_{d(i)} > y_{d(i)}|u$) for all the $i < j$ and $y_{d(j)} > x_{d(j)}$ (or $y_{d(j)} > x_{d(j)}|u$). For the other indexes l not in Diff or in Diff but with $d(l) > j$ the set includes both the CP-net with $x_l > y_l$ (or $x_l > y_l|u$) and with the opposite $x_l < y_l$ (or $x_l < y_l|u$), in all possible combinations.

Proof idea: The idea of this proof is to start with the set S_0 : the set that contains all the CP-nets with $o > o'$ and some CP-nets with $o \bowtie o'$. We then remove from S_0 all the induced CP-nets with $o \bowtie o'$ (since our goal is to have just the CP-nets that contribute to the dominance value $o > o'$). To do so we consider Equation 5 and we consider a correspondence between sets of induced CP-nets and the probability of these sets. Each term of Equation 5 corresponds to the probability of the set \bar{S}_j , a subset of S_0 . We will see that each set \bar{S}_j contains all the CP-nets with $o \bowtie_j o'$ but might also contain few CP-net such that $o > o'$ (so this is a lower bound). See Figure 17 for a visualization of this concept.

If we think of a generic j -term of Equation 5 as the probability of the set \bar{S}_j we can rewrite Equation 5 as:

$$\mathbb{P}_{\mathbb{Q}}^L(o > o') = \mathbb{P}(S_0) - \sum_{j=1}^m \mathbb{P}(\bar{S}_j) = \mathbb{P}(S_0 \setminus \bigcup_{j=1}^m \bar{S}_j) = \mathbb{P}(S_m).$$

where $j = 1, \dots, m$ are indices that belong to Diff . We have the following assumptions:

1. S_j and \bar{S}_j are a partition of the set S_{j-1} for each $j \in \{1, \dots, m\} = \text{Diff}$ (which implies that $S_j \cap \bar{S}_j = \emptyset \forall j \in \{1, \dots, m\}$ and that $S_j, \bar{S}_j \subset S_0 \forall j \in \{1, \dots, m\}$)
2. each set \bar{S}_j contains all the induced CP-nets that are incomparable for index $d(j)$ ($o \bowtie_{d(j)} o'$, see Definition 16), and $o \upharpoonright_{X_1, \dots, X_{d(j)-1}} > o' \upharpoonright_{X_1, \dots, X_{d(j)-1}}$

We will now prove the two conditions described above:

1. S_j and \bar{S}_j are a partition of the set S_{j-1} (Figure 18):

This is true by construction (considering the correspondence between the probability of an induced CP-net and the choice of its cp-tables) and follows from Definition 23.¹¹

11. Thus we do not remove the same CP-net more than once from the set S_0 , because we are removing partitions. In particular, since we are removing the probability of the j -term of the sum, we remove the probability of the set \bar{S}_j . The final probability that we obtain is the probability of the set S_m (as we can see from the first formulation of the formula: Equation 2).

2. \bar{S}_j contains all the induced CP-nets that have $o \bowtie o'$ for index $d(j)$ and $o \upharpoonright_{1, \dots, d(j)-1} > o' \upharpoonright_{1, \dots, d(j)-1}$:

First we note that, since \bar{S}_j are subset of S_0 , they can only contain induced CP-nets such that $o \bowtie o'$ or $o > o'$. Moreover, we know that all the induced CP-nets $\in S_0$ that have $o \bowtie o'$ satisfy the following:

- there exists an index l such that $o \bowtie_l o'$ (for Lemma 1);
- $o \upharpoonright_{1, \dots, l-1} > o' \upharpoonright_{1, \dots, l-1}$ (it cannot be $<$ because of Lemma 3).

We will now prove that \bar{S}_j contains all the induced CP-nets that have $o \bowtie_{d(j)} o'$ but we cannot exclude that it might also contain few CP-nets such that $o > o'$. We show that, given an induced CP-net C and two outcomes o and o' , such that $o \bowtie_{d(j)} o'$ then $C \in \bar{S}_j$. We know that $C \in S_0$, thus there are three options for C :

- (a) $C \in \bar{S}_l$ with $l < j$;
- (b) $C \in S_j$;
- (c) $C \in \bar{S}_j$.

We show now that:

- $C \notin S_j$: Suppose that $C \in S_j$, that is the set of the CP-nets with $o \upharpoonright_{X_{d(i)}} > o' \upharpoonright_{X_{d(i)}}$ (or $o \upharpoonright_{X_{d(i)}} > o' \upharpoonright_{X_{d(i)}|u}$) for all the $i \leq j$. In particular this is true for the index j : $o \upharpoonright_{X_{d(j)}} > o' \upharpoonright_{X_{d(j)}}$ (or $o \upharpoonright_{X_{d(j)}} > o' \upharpoonright_{X_{d(j)}|u}$). Moreover, since o and o' are incomparable for index $d(j)$, then $o \upharpoonright_{X_1, \dots, X_{d(j)-1}} \bowtie o' \upharpoonright_{X_1, \dots, X_{d(j)-1}}$. Then we know that in this case \bowtie corresponds to $>$ (see comment above). We can now apply Theorem 7 and conclude that also $o \upharpoonright_{X_1, \dots, X_{d(j)}} > o' \upharpoonright_{X_1, \dots, X_{d(j)}}$, which is a contradiction.
- $C \notin \bar{S}_l$ with $l < j$: Suppose that $C \in \bar{S}_l$ with $l < j$, this implies that o and o' are incomparable for $d(l) < d(j)$ and it is a contradiction given the assumption that $d(j)$ is the minimum index for the incomparability.

Thus we can conclude that $C \in \bar{S}_j$.

□

Theorem 9. Given a separable PCP-net \mathcal{Q} over n features X_1, \dots, X_n with binary domains $\mathcal{D}_i = \{x_i, \bar{x}_i\} \forall i \in \{1, \dots, n\}$ and given two outcomes o and o' then

$$\mathbb{P}(o > o') = \left(\prod_{i \in \text{Diff}^+(o, o')} p_i \right) \cdot \left(\prod_{j \in \text{Diff}^-(o, o')} (1 - p_j) \right)$$

where $p_i = \mathbb{P}(x_i > \bar{x}_i)$ in the PCP-net (and thus $(1 - p_i) = \mathbb{P}(\bar{x}_i > x_i)$) and where Diff^+ and Diff^- are a partition of Diff . Diff^+ is the set of indexes of variables such that $o \upharpoonright_{X_i} = x_i$ and $o' \upharpoonright_{X_i} = \bar{x}_i$ and Diff^- is the set of indexes of variables such that $o \upharpoonright_{X_i} = \bar{x}_i$ and $o' \upharpoonright_{X_i} = x_i$.

Proof. We call $\mathcal{I}(\mathcal{Q})$ the set of CP-nets induced by \mathcal{Q} , and $\mathcal{O}(C)$ the partial order defined by a CP-net C . We know that

$$\mathbb{P}(o \succ o') = \sum_{\substack{C \in \mathcal{I}(\mathcal{Q}) \text{ s.t.} \\ (o \succ o') \in \mathcal{O}(C)}} \mathbb{P}(C).$$

Now we show that:

1. if $i \in \text{Diff}^+(o, o')$ then the induced CP-nets that have $\bar{x}_i > x_i$ don't have $o \succ o'$ in their partial order over the features $(o \not\succeq o')$.
2. if $i \in \text{Diff}^-(o, o')$ then the induced CP-nets that have $x_i > \bar{x}_i$ don't have $o \succ o'$ in their partial order over the features $(o \not\succeq o')$.

We prove only the first assertion because the second is symmetric.

We suppose that $i \in \text{Diff}^+(o, o')$. We must have a chain of worsening flips from o to o' , such that $o \succ o'$.

We consider $o = (yx_iz)$ and $o' = (y'\bar{x}_iz')$ where y and y' are the assignments of the variables X_1, \dots, X_{i-1} and z and z' are the assignments of the variables X_{i+1}, \dots, X_n respectively of o and o' .

Because the features are all independent, to have a chain of worsening flips from o to o' we must have a chain of worsening flips from y to y' , from z to z' and for x_i to \bar{x}_i . But, in some induced CP-nets that have $\bar{x}_i > x_i$, the first two may be possible, but the third one is an improvement. Thus, the induced CP-nets with $\bar{x}_i > x_i$ have no chains of worsening flips from o to o' .

Thus, we have to consider only the induced CP-nets that for all the features $i \in \text{Diff}^+(o, o')$ have $x_i > \bar{x}_i$, and for all the features $i \in \text{Diff}^-(o, o')$ have $\bar{x}_i > x_i$.

$$\mathbb{P}(o \succ o') = \sum_{\substack{C \in \mathcal{I}(\mathcal{Q}) \text{ s.t.} \\ (x_i > \bar{x}_i) \forall i \in \text{Diff}^+(o, o') \\ (\bar{x}_i > x_i) \forall i \in \text{Diff}^-(o, o')}} \mathbb{P}(C).$$

We now observe that all of these induced CP-nets have $(o \succ o') \in \mathcal{O}(C)$. That is because the variables $x_j \notin (\text{Diff}^+(o, o') \cup \text{Diff}^-(o, o'))$ have the same values in o and o' , and so we can build a chain of worsening flip for the variables $x_j \in (\text{Diff}^+(o, o') \cup \text{Diff}^-(o, o'))$ by definition, while holding the other variables fixed.

We can compute the probability $\mathbb{P}(o > o')$:

$$\begin{aligned}
 \mathbb{P}(o > o') &= \sum_{\substack{C \in \mathcal{I}(\mathcal{Q}) \text{ s.t.} \\ (x_i > \bar{x}_i) \\ \forall i \in \text{Diff}^+(o, o') \\ (\bar{x}_i > x_i) \\ \forall i \in \text{Diff}^-(o, o')}} \mathbb{P}(C) \\
 &= \sum_{\substack{C \in \mathcal{I}(\mathcal{Q}) \text{ s.t.} \\ (x_i > \bar{x}_i) \\ \forall i \in \text{Diff}^+(o, o') \\ (\bar{x}_i > x_i) \\ \forall i \in \text{Diff}^-(o, o')}} \left(\prod_{j \text{ s.t. } x_j > \bar{x}_j} p_j \prod_{j \text{ s.t. } \bar{x}_j > x_j} (1 - p_j) \right) \\
 &= \left(\prod_{i \in \text{Diff}^+(o, o')} p_i \prod_{i \in \text{Diff}^-(o, o')} (1 - p_i) \right) \cdot \sum_{\substack{C \in \mathcal{I}(\mathcal{Q}) \text{ s.t.} \\ (x_i > \bar{x}_i) \\ \forall i \in \text{Diff}^+(o, o') \\ (\bar{x}_i > x_i) \\ \forall i \in \text{Diff}^-(o, o')}} \left(\prod_{\substack{j \text{ s.t. } x_j > \bar{x}_j \\ j \notin \text{Diff}^+(o, o')}} p_j \prod_{\substack{j \text{ s.t. } \bar{x}_j > x_j \\ j \notin \text{Diff}^-(o, o')}} (1 - p_j) \right).
 \end{aligned}$$

But we know that:

$$\sum_{\substack{C \in \mathcal{I}(\mathcal{Q}) \text{ s.t.} \\ (x_i > \bar{x}_i) \\ \forall i \in \text{Diff}^+(o, o') \\ (\bar{x}_i > x_i) \\ \forall i \in \text{Diff}^-(o, o')}} \left(\prod_{\substack{j \text{ s.t. } x_j > \bar{x}_j \\ j \notin \text{Diff}^+(o, o')}} p_j \prod_{\substack{j \text{ s.t. } \bar{x}_j > x_j \\ j \notin \text{Diff}^-(o, o')}} (1 - p_j) \right) = 1.$$

Thus we can conclude that:

$$\mathbb{P}(o > o') = \left(\prod_{i \in \text{Diff}^+(o, o')} p_i \right) \left(\prod_{i \in \text{Diff}^-(o, o')} (1 - p_i) \right).$$

□

Theorem 10. *Given a PCP-net \mathcal{Q} with a separable dependency graph and two outcomes o and o' , then $\mathbb{P}_{\mathcal{Q}}(o > o') = \mathbb{P}_{\mathcal{Q}}^L(o > o')$.*

Proof. We prove that the formula for the lower bound in the case of separable PCP-nets is equal to the formula described in Theorem 9. We observe that each formulation is a product of factors and each factor corresponds to a variable that has a different value in o and o' . Each variable that has a different value in o and o' has a unique factor in each formulation. We prove that the factors that correspond to the same variable coincide in the two formulations.

Suppose we have a PCP-net that has the following PCP-tables: $X_i : x_i > \bar{x}_i, q_i$ where q_i is the probability $\mathbb{P}(x_i > \bar{x}_i)$.

We consider two cases: the first one is the case of a variable X_i such that $o \upharpoonright_{X_i} = x_i$ and $o' \upharpoonright_{X_i} = \bar{x}_i$. In the formulation of dominance for separable PCP-nets (Theorem 9), we have a factor that corresponds to q_i because the variable X_i belongs to the set $\text{Diff}^+(o, o')$. In the lower bound formulation we have a factor that corresponds to:

$$(1 - \mathbb{P}(o' \upharpoonright_{X_i} > o \upharpoonright_{X_i})) = (1 - \mathbb{P}(\bar{x}_i > x_i)) = \mathbb{P}(x_i > \bar{x}_i) = q_i .$$

The second case is the case of a variable X_i such that $o \upharpoonright_{X_i} = \bar{x}_i$ and $o' \upharpoonright_{X_i} = x_i$. In the formulation of Theorem 9, we have a factor that corresponds to $(1 - q_i)$ because the variable X_i belongs to the set $\text{Diff}^-(o, o')$. In the lower bound formulation we have a factor that corresponds to:

$$(1 - \mathbb{P}(o' \upharpoonright_{X_i} > o \upharpoonright_{X_i})) = (1 - \mathbb{P}(x_i > \bar{x}_i)) = 1 - q_i .$$

Thus, the two formulations coincide. □

Theorem 11. *Given a PCP-net \mathcal{Q} and two outcomes o and o' , $\mathbb{P}_{\mathcal{Q}}^U(o > o')$ is an upper bound for $\mathbb{P}_{\mathcal{Q}}(o > o')$.*

Proof. It is enough to prove that all CP-nets induced by \mathcal{Q} that support $o > o'$ have the row $u : o \upharpoonright_{X_j} > o' \upharpoonright_{X_j}$ for all variables $X_j \in (\text{Diff} \cap \text{FA})$. We prove this sentence by contradiction: we consider an induced CP-net C that contains row $u : o' \upharpoonright_X > o \upharpoonright_X$ for $X \in (\text{Diff} \cap \text{FA})$ and we prove that $C \models (o' > o) \vee (o \bowtie o')$. We have two cases: (1) X is an independent node. Then flip $o \upharpoonright_X \rightarrow o' \upharpoonright_X$ can't be a worsening flip. That implies that all the worsening paths starting from o do not contain a flip for variable X , so o' can't be reached. (2) X is a dependent node. Thus, the set $\text{Anc}(X) = \{X_{i_1}, \dots, X_{i_k}\}$ with $k \geq 1$ and all of them have fixed value on o and o' . We call u the assignment of $\text{Pa}(X)$ in o and o' . Each worsening path from o to o' has the value u for $\text{Pa}(X)$ in each step of the path, because all the ancestors of X are fixed. This implies that no worsening path from o to o' contains other assignments for the parents of X . Thus, no worsening path starting from o contains a flip for variable X , so o' can't be reached. Thus $C \not\models (o > o')$ and so $C \models (o' > o) \vee (o \bowtie o')$. □

Lemma 5. *There exists a profile of CP-nets P such that*

$$\{\text{PR}_O(P), \text{PR}_I(P)\} \cap \{\text{LS}_O(P), \text{LS}_I(P)\} = \emptyset;$$

and there exists a profile of CP-nets P' such that:

$$\text{PR}_O(P') \neq \text{PR}_I(P');$$

and there exists a profile of CP-nets P'' such that:

$$\text{LS}_O(P'') \neq \text{LS}_I(P'').$$

Proof.

- First we prove that there exists a profile of CP-nets P such that $\{\text{PR}_O(P), \text{PR}_I(P)\} \cap \{\text{LS}_O(P), \text{LS}_I(P)\} = \emptyset$. Let us take the following profile P of four CP-nets over two variables A and B :

- \mathbf{C}_1 with probability 0.095. C_1 has an edge from X_1 to X_2 and CP-tables: $x_1 > \bar{x}_1$ and $x_1 : \bar{x}_2 > x_2, \bar{x}_1 : x_2 > \bar{x}_2$.
- \mathbf{C}_2 with probability 0.505. C_2 has an edge from X_1 to X_2 and CP-tables: $x_1 > \bar{x}_1$ and $x_1 : x_2 > \bar{x}_2, \bar{x}_1 : \bar{x}_2 > x_2$.
- \mathbf{C}_3 with probability 0.005. C_3 does not have an edge from X_1 to X_2 and has CP-tables: $\bar{x}_1 > x_1$ and $x_2 > \bar{x}_2$.
- \mathbf{C}_4 with probability 0.395. C_4 does not have an edge from X_1 to X_2 and has CP-tables: $\bar{x}_1 > x_1$ and $\bar{x}_2 > x_2$.

Proportion gives us the PCP-net $(x_1 > \bar{x}_1, 0.6)$ and $(x_1 : x_2 > \bar{x}_2, 0.51), (\bar{x}_1 : x_2 > \bar{x}_2, 0.1)$, while LS outputs $(x_1 > \bar{x}_1, 0.59)$ and $(x_1 : x_2 > \bar{x}_2, 0.29), (\bar{x}_1 : x_2 > \bar{x}_2, 0)$. Thus we obtain that $PR_O(P) = \bar{x}_1\bar{x}_2, PR_I(P) = x_1x_2, LS_O(P) = x_1\bar{x}_2$ and $LS_I(P) = x_1\bar{x}_2$.

- The fact that there exists profiles P' and P'' such that $PR_O(P') \neq PR_I(P')$ (or $LS_O(P'') \neq LS_I(P'')$) is an immediate consequence of the fact that the most probable optimal outcome and the outcome of the most probable CP-net can be different. For example, consider a PCP-net on two variables X_1 and X_2 with the PCP-tables $(x_1 > \bar{x}_1, 0.6), (x_1 : x_2 > \bar{x}_2, 0.6)$ and $(\bar{x}_1 : x_2 > \bar{x}_2, 0)$. The most probable induced CP-net has $x_1 > \bar{x}_1, x_1 : x_2 > \bar{x}_2$ and $\bar{x}_1 : \bar{x}_2 > x_2$, thus x_1x_2 is the optimal outcome. However, the most probable optimal outcome of the PCP-net is $\bar{x}_1\bar{x}_2$. Therefore, PR_O (respectively, LS_O) will differ from PR_I (resp. LS_I) on a profile that produces this PCP-net. \square

Theorem 12. *Given a profile of CP-nets, PR_I produces the same result as sequential voting with majority.*

Proof. Consider a variable X_i with domain $\{x_i, \bar{x}_i\}$, and an assignment u for the parents of X_i . With sequential voting we choose the value of the domain that corresponds to the first value of the ordering that maximizes the following:

$$\max_{j \in \{1, \dots, m\}} \left\{ \left[\sum_{C_j: x_i > \bar{x}_i | u} \mathbb{P}(C_j) \right], \left[1 - \sum_{C_j: x_i > \bar{x}_i | u} \mathbb{P}(C_j) \right] \right\}.$$

With PR_I , we create a PCP-net that has, for the row in the PCP-table of X_i corresponding to assignment u for its parents, the probability $\sum_{C_j: x_i > \bar{x}_i | u} \mathbb{P}(C_j)$ for $x_i > \bar{x}_i$ and $1 - \sum_{C_j: x_i > \bar{x}_i | u} \mathbb{P}(C_j)$ for $\bar{x}_i > x_i$. To compute the CP-tables of the most probable induced CP-net, we choose the orderings with maximal probability: for each variable X_i , given u , we choose the ordering that maximizes the following probability:

$$\max_{j \in \{1, \dots, m\}} \left\{ \left[\sum_{C_j: x_i > \bar{x}_i | u} \mathbb{P}(C_j) \right], \left[1 - \sum_{C_j: x_i > \bar{x}_i | u} \mathbb{P}(C_j) \right] \right\}.$$

To compute the optimal outcome of the most probable induced CP-net, we choose the greater literal in the ordering that appears in the CP-table, given u . This is for a generic variable X_i and assignment u , thus is true for all the variables and assignment of their parents. \square

Theorem 13. *The following holds:*

(i) PR_O , LS_O , and LS_I satisfy homogeneity;

(ii) PR_I and PR_O satisfy participation.

(iii) PR_I satisfies consensus and PR_O satisfies consensus over a single feature.

Proof. i) For PR_O , consider a profile $\bar{P} = (C_1, f_1), \dots, (C_k, f_k)$ from which we can get the normalised frequencies $(C_1, \frac{f_1}{m}), \dots, (C_k, \frac{f_k}{m})$. Considering each CP-net N times, we obtain the following distribution over Nm CP-nets, $\bar{P}' = (C_1, Nf_1), \dots, (C_k, Nf_k)$, with the following normalised frequencies $(C_1, \frac{Nf_1}{Nm}), \dots, (C_k, \frac{Nf_k}{Nm})$. The probability of a generic CP-net C_i in \bar{P}' is $\frac{Nf_i}{Nm} = \frac{f_i}{m}$ which is the same as the probability generated by \bar{P} . For LS_O , and LS_I , consider the fact that if, in the previous proof, we added N copies of each CP-net to the original profile it would result in the same probability distribution over the CP-nets. This fact is true for any collection of CP-nets, therefore, we generate the same set of equations to minimize in the definition of LS , and thus the same PCP-net solution.

ii) We first consider the case $r = PR_I$. We provide a proof for non-anonymous profiles, for ease of readability. The version for anonymous profiles can easily be deduced. Consider a profile $P = (C_1, \dots, C_m)$ and an additional CP-net C . We have to prove that $r(P \cup \{C\}) >_C r(P)$. PR gives us a PCP-net Q for P and a PCP-net Q' for $P \cup \{C\}$. Since P and $P \cup \{C\}$ are \mathcal{O} -legal, let X be the first variable according to \mathcal{O} such that $r(P \cup \{C\})|_X \neq r(P)|_X$ and let $u = r(P)|_{PA(X)}$. This means that in the most probable induced CP-net of Q there is a preference statement $u : r(P)|_X > r(P \cup \{C\})|_X$ and in Q' there is $u : r(P \cup \{C\})|_X > r(P)|_X$. Thus the probability $\mathbb{P}[u : r(P)|_X > r(P \cup \{C\})|_X] \geq 0.5$ in Q , but ≤ 0.5 in Q' . This means that C has the row $u : r(P \cup \{C\})|_X > r(P)|_X$ in X 's CP-table. Thus the outcome $r(P \cup \{C\}) \geq_C r(P)$.

A similar reasoning applies to the case $r = PR_O$. In the PCP-tables of Q' , the probabilities of the rows corresponding to the CP-tables of C increase with respect to Q . Thus, for each feature X and each assignment u of its parents, the probability to obtain the first ranked value in the u rows of C CP-tables increases. This means that it improves the result for C .

iii) We first consider the case $r = PR_I$. Consider profile \bar{P} and assume there is an alternative o s.t. $o >_{C_i} r(\bar{P}) \forall i \in \{1, \dots, m\}$. Since \bar{P} is \mathcal{O} -legal, let X be the first variable according to \mathcal{O} such that $o|_X \neq r(\bar{P})|_X$. Let u be the assignment to X 's parents in o and $r(\bar{P})$ ($o|_{PA(X)} = r(\bar{P})|_{PA(X)}$ since X is the first variable according to \mathcal{O} in which they differ). Since $o >_{C_i} r(\bar{P}) \forall i \in \{1, \dots, m\}$, it must be that $u : o|_X > r(\bar{P})|_X \forall C_i$. Let us now consider the PCP-net Q obtained from \bar{P} by PR . It is easy to see that in the PCP-table of X the probability of $u : o|_X > r(\bar{P})|_X$ will be strictly higher than the probability of $u : o|_X < r(\bar{P})|_X$, which implies that the most probable induced CP-net must have the row $u_i : o|_X > r(\bar{P})|_X$. The optimal outcome of the most probable induced CP-net must have the assignment $o|_X$ for the variable X because is ranked first in the table in the u row. But $r(\bar{P})|_X \neq o|_X$ and we have a contradiction.

A similar reasoning applies to the case $r = PR_O$, but in a weaker version. We will prove that, for any profile \bar{P} , there is no alternative o such that o differs from $r(\bar{P})$ on only a single variable and $o >_{C_i} r(\bar{P}) \forall C_i$. Assume that there were an alternative o such that $o >_{C_i} r(\bar{P})$, $\forall C_i$ and o differs from $r(\bar{P})$ only on the variable X . The probability of $u : o|_X > r(\bar{P})|_X$ in the PCP-table of X is equal to 1 because all the CP-nets in the profile prefer o to $r(\bar{P})$. Thus the probability of $r(\bar{P})$ is equal to 0, and we have a contradiction. \square