Reasoning with PCP-nets in a Multi-Agent Context

Cristina Cornelio
University of Padova
Padova, Italy
cornelio@math.unipd.it

Umberto Grandi*
University of Padova
Padova, Italy
umberto.grandi@ut-capitole.fr

Judy Goldsmith
University of Kentucky
Lexington, USA
goldsmit@cs.uky.edu

Nicholas Mattei
NICTA and UNSW
Sydney, Australia
nicholas.mattei@nicta.com.au

Francesca Rossi
University of Padova
Padova, Italy
frossi@math.unipd.it

K. Brent Venable
Tulane University and IHMC
New Orleans, USA
kvenable@tulane.edu

ABSTRACT

PCP-nets generalize CP-nets to model conditional preferences with probabilistic uncertainty. In this paper we use PCP-nets in a multi-agent context to compactly represent a collection of CP-nets, thus using probabilistic uncertainty to reconcile possibly conflicting qualitative preferences expressed by a group of agents. We then study two key preference reasoning tasks: finding an optimal outcome which best represents the preferences of the agents, and answering dominance queries. Our theoretical and experimental analysis demonstrates that our techniques are efficient and accurate for both reasoning tasks.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems

General Terms

Theory, Algorithms, Experimentation

Keywords

Preference modelling; preference reasoning; multi-agent systems; reasoning with uncertainty.

1. INTRODUCTION

Preferences are ubiquitous in our everyday life; the study of how to model and reason with them plays an important role in multi-agent systems and AI [5, 6, 11, 22]. The ability to express preferences in a faithful but compact way, which can be handled efficiently, is essential in many reasoning tasks including e-commerce, combinatorial optimization, multi-agent planning and agreement, and other scenarios where the number of options outstrips an agent's ability to view and rank all the available choices. Multi-attribute preference modeling and reasoning causes a combinatorial explosion, often leading to high computational cost [9, 10, 12]. The set of alternatives is often described as a product of multiple features, for example, a user's preferences over a set of cars, which can be described by their colors, technical specifications, cost, reliability, etc. Many compact representation languages that have been developed to tackle the computational challenges arising from these problems e.g., conditional preference nets (CP-nets) [4], soft constraints [3, 22], and GAI-nets [13].

In this paper we assume individual agents express their preferences as CP-nets, a qualitative preference modeling framework that allows for conditional preference statements. Agents may use these preferences to make decisions about joint plans, actions, or items in multi-agent environments: agents express their preferences over a set of alternative decisions, and such preferences are aggregated into a collective decision. Often in these settings we need to handle partial preferences of agents [21] or reconcile preferences that directly conflict. Voting has been used extensively to resolve these conflicts [16–19, 24]. Recently, CP-nets have been extended to allow for modeling of uncertainty over preferences [2, 7]. This uncertainty may be due to several different causes: an agent may be unsure about his preference ordering over certain items, or there could be noise in his preference structure due to lack of precision in elicitation or errors during data collection (e.g., measurement error from remote sensors). We investigate the use of PCP-nets as a structure to represent the preferences of a group of agents.

PCP-nets model uncertain preferences natively while maintaining the preferential dependency structure employed by CP-nets [2, 7, 8]. However, in a PCP-net, a preference ordering over a variable's domain is replaced by a probability distribution over all possible preference orderings of the variable's domain. Thus, a PCP-net defines a probability distribution over a collection of CP-nets: all those CP-nets that can be obtained from the PCP-net by choosing an ordering from each probability distribution over orderings.

Given a PCP-net, we focus on two collective reasoning tasks that agents may want to perform: finding the outcome that is most preferred by the agents or decide whether one outcome is collectively preferred to another (a dominance query). Given a PCP-net, one can define the optimal outcome in two natural ways: the most probable optimal outcome, or the optimal outcome of the most probable CP-net induced by the PCP-net. If the dependency structure of the PCP-net has bounded in-degree, both kinds of optimal outcomes can be found in time polynomial in the size of the PCP-net.

Previous efforts to tackle these collective reasoning problems over CP-nets have primarily focused on the question of optimality [16] or dominance [23] and have not dealt with uncertainty.

*Current affiliation: IRIT, University of Toulouse, France.

Appears in: *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015), Bordini, Elkind, Weiss, Yolum (eds.), May 4–8, 2015, Istanbul, Turkey.*

Generally, a voting method is defined in order to determine the most preferred alternative. These methods are usually sequential and work at the level of the individual variables [14–16, 24, 25]. In contrast, we propose to aggregate the collection of CP-nets into a single structure, a PCP-net, on which we directly perform collective reasoning tasks. This provides a formalism to perform other preference 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.

We consider two methods for aggregating a collection of CP-nets into a single PCP-net: by extracting the probability distribution directly from the CP-nets (proportional method, *PR*) or by minimizing a notion of error between the aggregated structure and the original set of CP-nets (least-squares method, *LS*). We combine these two methods with two ways of extracting an optimal outcome from a PCP-net: the most probable optimal outcome and the optimal outcome of the most probable induced CP-net. This gives us four approaches, each of which can be seen as a voting rule which takes as input a profile of individual CP-nets and outputs a most preferred outcome. We then analyze the methods experimentally, showing that when aggregating the agents' CP-nets, the proportion method yields a PCP-net which more accurately captures the preferences of the agents than the least-square method.

We observe that the outcome computed by the PR method on the PCP-net, combined with taking the optimal outcome of the most probable induced CP-net, *exactly coincides* with the result of sequential voting as proposed by Lang and Xia [14]. This indicates that, even though in the construction of the PCP-net we make several simplifying assumptions, we are able to recover the result of an established voting method for the set of individual CP-nets which make up the PCP-net. This is not surprising, as Lang and Xia's sequential method was designed to obtain the outcome which best satisfies the preferences of the whole collection of agents. Since we can obtain this outcome by generating a PCP-net representing the collection of given CP-nets we do more than just find a collectively optimal outcome: we can also use the PCP-net to answer dominance queries

We then study dominance. Given two outcomes, it is in general computationally difficult to compute the probability that one dominates the other one, since this is the sum of all probabilities of the CP-nets, induced by the PCP-net, where dominance holds [2]. For PCP-nets whose dependency graph is a polytree, for which dominance is still difficult, we give a polynomial time approximation in the form of a lower and an upper bound to the correct probability value for dominance. We show, experimentally, that often the range between the lower and upper bound is small, and in particular that the lower bound is very close to the correct value. When the PCP-net is obtained by aggregating a collection of CP-nets via the proportion method, we show experimentally that our approximation of dominance closely models dominance computed on the collection of given CP-nets. The same result holds for a deterministic form of dominance we call MP-dominance.

Our results suggest that the PCP-net obtained via the proportion method from a given collection of CP-nets is a compact and accurate aggregation model of the input profile. Using the PCP-net representation of a profile of CP-nets allows us to not only save space but also answer both optimality and dominance queries using accurate and efficient (for the case of polytrees) heuristics.

2. PCP-NETS

We give a brief introduction to PCP-nets, as defined by Bigot et al. [2] and Cornelio et al. [7]. We assume that the domain of each variable is binary and that the induced width k of the dependency graph is bounded by a constant.

DEFINITION 1. *A **PCP-net** (Probabilistic CP-net) is a directed graph where each node represents a variable (often called feature) $F = \{X_1, \ldots, X_n\}$ each with binary domains $\mathcal{D}(X_1), \ldots, \mathcal{D}(X_n)$. For each feature X_i, there is a set of* parent *features* $\mathrm{Pa}(X_i)$ *that can affect the preferences over the values of X_i. This defines a* dependency graph *in which each node X_i has edges from all features in* $\mathrm{Pa}(X_i)$. *Given this structural information, for each feature X_i, instead of giving a preference ordering over the domain of X_i (as in the CP-nets), we give a probability distribution over the set of all preference orderings (total orderings over $\mathcal{D}(X_i)$) for each complete assignment on* $\mathrm{Pa}(X_i)$.

DEFINITION 2. *Given a feature X in a PCP-net, its* PCP-table *is a 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 probability distributions in $\{0, 1\}$ we recover the definition of CP-nets [4].

Given a PCP-net $\mathcal{Q}$, a *CP-net induced by* $\mathcal{Q}$ has the same variables, with the same domains, as $\mathcal{Q}$. The edges of the induced CP-net are a subset of the edges in the PCP-net. Thus, CP-nets induced by the same PCP-net may have different dependency graphs. Moreover, the CP-tables are generated accordingly for the chosen edges: for each independent feature, one ordering over its domain (i.e., a row in its PCP-table) is selected; for dependent features, an ordering is selected for each combination of the values of the parent features. 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.

More precisely, given a PCP-net, we have a probability p for each row $u : x > \bar{x}$ of each PCP-table, while the probability of $u : \bar{x} > x$ corresponds to $1 - p$.

DEFINITION 3. *Given a PCP-net $\mathcal{Q}$ and an induced CP-net C, we can define C by its CPTs (Conditional Preference Tables). Let $\bar{q}$ be the set of all rows of the CPTs that define C. We define the probability of C,* $\mathrm{fp}_C(\bar{q})$, *to be the product of the probabilities of the $q_i \in \bar{q}$.*

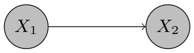
Hence, a PCP-net induces a probability distribution over the set of all induced CP-nets.

EXAMPLE 1. *Consider the PCP-net in Figure 1 with two features, X_1 and X_2, with domains $\mathcal{D}_{X_i} = \{x_i, \bar{x}_i\}$. On the left side a graph that pictures a preferential dependency between variable X_1 and X_2, and on the right side the preference tables, showing for each ordering of the values of X_1 the corresponding probabilities, and for each value of X_1 the probability of observing a given ordering over values of X_2. Figure 2 describes a CP-net that has been induced from the PCP-net of Figure 1. The probability associated to this CP-net is defined by the following formula:* $\mathrm{fp}_C(\bar{q}) = [(1 - q_1^1) \cdot (1 - q_1^2) \cdot q_2^2]$.

2.1 Optimality

An *outcome* in a PCP-net is a complete assignment to all the variables. We consider two notions of optimality for outcomes:

- **The most probable optimal outcome:** the outcome with the highest probability of being optimal. The probability of an outcome o being optimal is the sum of the probabilities of the induced CP-nets that have o as the optimal outcome.



X_1 orderings	$\mathbb{P}$
$x_1 > \bar{x}_1$	q_1^1
$\bar{x}_1 > x_1$	$1 - q_1^1$

X_1 values	X_2 orderings	$\mathbb{P}$
x_1	$x_2 > \bar{x}_2$	q_1^2
	$\bar{x}_2 > x_2$	$1 - q_1^2$
$\bar{x}_1$	$x_2 > \bar{x}_2$	q_2^2
	$\bar{x}_2 > x_2$	$1 - q_2^2$

Figure 1: A PCP-net.

X_1 orderings
$\bar{x}_1 > x_1$

X_1 values	X_2 orderings
x_1	$\bar{x}_2 > x_2$
$\bar{x}_1$	$x_2 > \bar{x}_2$

Figure 2: A CP-net induced by the PCP-net in Figure 1.

• **The optimal outcome of the most probable induced CP-net:** the optimal outcome of the induced CP-net with the highest probability.

Notice that the optimal outcome of the most probable induced CP-net may be different from the most probable optimal outcome [7]. To compute these two outcomes, it is possible to generate two Bayesian networks and compute their maximal joint probability [20]. Computing the result for either notion of optimal outcome has polynomial computational complexity if the induced width of the dependency graph of the PCP-net is bounded [7].

2.2 Dominance

DEFINITION 4 (DOMINANCE FOR PCP-NETS). *Given a PCP-net $\mathcal{Q}$ and a pair of outcomes o and o', a* dominance query *returns the probability that $\mathcal{Q} \models o > o'$. That is:*

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

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

Bigot et al. [2] prove that dominance for PCP-nets is #P-hard, even if the the dependency structure is acyclic, the longest path has length 3 and each node has at most one outgoing edge and at most 4 parents. They give an algorithm to obtain dominance for a binary-valued tree structured PCP-net that takes $O(2^{2s^2}n)$ where n is the number of features and s the Hamming distance between o and o'.

Boutilier et al. [4] prove that evaluating a dominance query for binary-valued CP-nets has complexity $O(n)$ for tree structured CP-nets, polynomial for polytrees, NP-complete for acyclic CP-nets that are directed path singly connected, and NP-complete for CP-nets with bounded path number between two nodes. The difficult cases remain of course difficult for PCP-nets since CP-nets are a restricted form of PCP-nets.

3. AGGREGATION METHODS

We consider collections of CP-nets [4], also called *profiles* in voting theory terminology. A profile of CP-nets is a set of CP-nets on the same set of variables: $P = (C_1, \cdots, C_m)$. We focus on O-legal profiles [14] of acyclic CP-nets where each variable has a binary domain.

DEFINITION 5. *(O-legality) A profile of m CP-nets over n variables is said to be O-legal if all the dependency graphs of the CP-nets share a topological ordering of the variables.*

This assumption is widely used in computational social choice and preference reasoning, and is often sufficient for tractability.

We start from an O-legal profile of m CP-nets over variables $X_1, \cdots, X_n$ each with a binary domain. We define the *relative frequency* of a CP-net C_i, written as $freq_i$ as the percentage of times C_i appears in P. In the following, a profile will be written as $P = ((C_1, freq_1), \cdots, (C_k, freq_k))$, with $\sum_{i=1}^{k} freq_i = 1$.

The use of relative frequencies is relevant in domains with a high number of variables and a low number of voters as in some cases (e.g. political parties) many agents may express the same preferences. We use the same formulation in the case where CP-net is unique, we can "count" all these as 1's and we merely introduce the counts in this way to ease discussion.

Given a profile P of CP-nets, there may be no PCP-net which induces exactly the same distribution over the CP-nets in P. This can be seen in the following example.

EXAMPLE 2. *Given a profile of CP-nets over two variables X_1 and X_2:*
$(C_1, 0.5)$: $(x_1 > \bar{x}_1), (x_1 : x_2 > \bar{x}_2)$ *and* $(\bar{x}_1 : \bar{x}_2 > x_2)$
$(C_2, 0.4)$: $(x_1 > \bar{x}_1), (x_1 : \bar{x}_2 > x_2)$ *and* $(\bar{x}_1 : x_2 > \bar{x}_2)$
$(C_3, 0.1)$: $(x_1 > \bar{x}_1)$ *and* $(x_2 > \bar{x}_2)$.
The PCP-net representing such a profile must satisfy the following system of equations, where $\bar{q} = (q_1^1, q_1^2, q_2^2)$, q_1^1 is the probability of $x_1 > \bar{x}_1$, q_1^2 is the probability of $x_1 : x_2 > \bar{x}_2$, and q_2^2 is the probability of $\bar{x}_1 : x_2 > \bar{x}_2$:

$$\begin{cases} \mathrm{fp}_{C_1}(\bar{q}) = & q_1^1 q_1^2 (1 - q_2^2) \\ \mathrm{fp}_{C_2}(\bar{q}) = & q_1^1 (1 - q_1^2) q_2^2 \\ \mathrm{fp}_{C_3}(\bar{q}) = & q_1^1 q_1^2 q_2^2 \end{cases} \Rightarrow \begin{cases} \mathrm{fp}_{C_1}(\bar{q}) = & 0.5 \\ \mathrm{fp}_{C_2}(\bar{q}) = & 0.4 \\ \mathrm{fp}_{C_3}(\bar{q}) = & 0.1 \end{cases}$$

This system has no solution for $\bar{q} \in [0, 1]^3$.

In general, the system is over-constrained and will rarely admit a solution. Therefore, we need to define aggregation methods that work even when there is no PCP-net that exactly recovers the input profile of CP-nets.

We now define two methods to represent a profile of CP-nets using a PCP-net. As we are not guaranteed to find a PCP-net representing the exact distribution of induced CP-nets in the profile we must resort to methods approximating this ideal distribution. The first method we propose generates a PCP-net by taking the union of the dependency graphs of the given CP-nets and determining the probabilities in the PCP-tables from the frequencies of the CP-nets in the profile.

DEFINITION 6. *Given a profile of CP-nets $P = (C_i, \mathrm{freq}_i)$, 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:*

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

(and $\mathbb{P}(\bar{x} > x|u) = 1 - \mathbb{P}(x > \bar{x}|u)$)), i.e., the probability of the ordering $x > \bar{x}$ for variable X, given assignment u of $\mathrm{Pa}(X)$, is the sum of probabilities of the CP-nets that have that particular ordering over the domain of X, given u.

The second 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 7. *Let $P = (C_i, \text{freq}_i)$ be a profile of CP-nets. The* Least Square (LS) *aggregating 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 that solve the following problem:*

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

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 k 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 3.

The *LS* method requires solving a linear program with a linear number of equations (one for each CP-net in the initial profile). However, computing fp_{C_i} may require exponential time as the PCP-net resulting from a generic profile may have an exponential number of cp-statements. We can ensure that the union graph of an O-legal profile has bounded width – making *LS* a polynomial method – by assuming the following O-boundedness condition: for each feature j there are sets $PP(X_j) \subseteq \{X_1, \ldots, X_n\}$ of possible parents such that (1) $|PP(X_j)| < k$ for all j, and (2) for all individuals i, $Pa_i(X_j) \subseteq PP(X_j)$.

Computing the PCP-net using method *PR* may also require exponential time, as for *LS*, because 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 O-boundedness condition.

4. COMPUTING OPTIMALITY

Let $\mathcal{P}$ be the set of all CP-net profiles P of m voters over a set of alternatives X, a CP-voting rule $r : \mathcal{P} \to X$ is a function that maps each profile P into an alternative $r(P) \in X$.[1]

We define four voting rules by combining the two aggregation methods *PR* and *LS* presented in Definition 6 and 7 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 the optimal outcome for PR_O and PR_I is polynomial if the graph of the resulting PCP-net has bounded width. This property can be obtained via the O-boundedness condition.

The four methods may produce different outcomes:

OBSERVATION 1. *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 P such that

$$\text{PR}_O(P) \neq \text{PR}_I(P) \quad (\textit{or } \text{LS}_O(P) \neq \text{LS}_I(P)).$$

It is interesting to observe that PR_I returns the same result as the *sequential voting rule with majority* [14], that consists of applying the majority rule "locally" on each issue in the order given by O.

THEOREM 1. *Given any profile of CP-nets,* PR_I *produces the same result as sequential voting with majority.*

[1] In what follows we assume that CP-voting rules use lexicographic tie-breaking to return a unique winner.

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:

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

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:

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

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. $\square$

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. But this kind of optimality is computationally hard and thus we focus on optimal outcomes that can be computed in polynomial time.

4.1 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 named *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 two different scoring functions, each of which 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 O-legal CP-profiles. Given a profile P, we first define three parameters: $d_P^{>}(F,G) = |\{C \in P : F(P) >_C G(P)\}|$, $d_P^{<}(F,G) = |\{C \in P : G(P) >_C F(P)\}|$ and $d_P^{\bowtie}(F,G) = |\{C \in P : F(P) \bowtie_C G(P)\}|$. Note that $d_P^{>} + d_P^{<} + d_P^{\bowtie} = |P|$. We then define the function $Dom_P(F,G)$ as:

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

We use 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 $[-1, 1]$. Our second scoring function is inspired by Copeland scoring:

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

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

For all our experiments we randomly generate profiles of CP-nets, and PCP-nets. Generating PCP-nets i.i.d. is non-trivial [1] and

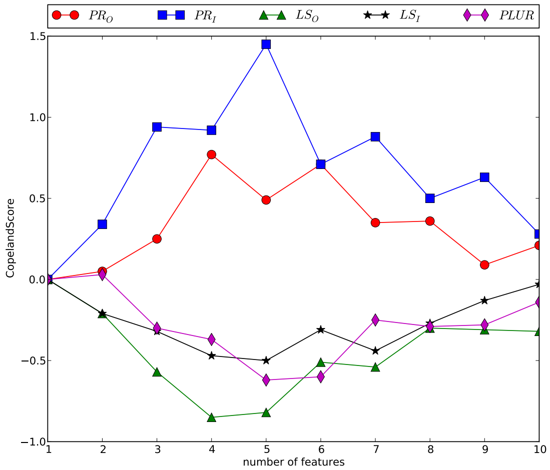


Figure 3: $CopelandScore$ **vs. the number of features.**

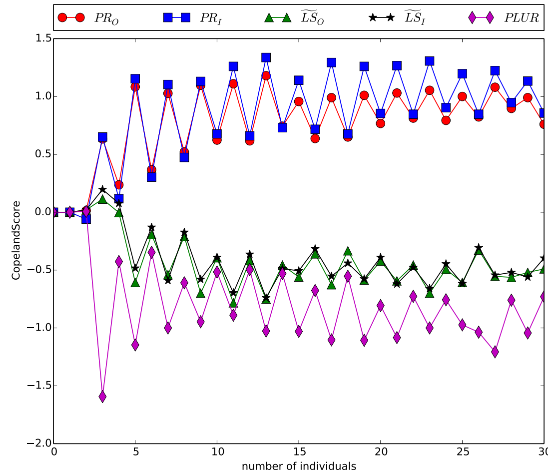


Figure 4: $CopelandScore$ **vs. the number of CP-nets.**

therefore we use an approximation method for random generation of CP-nets and probabilities.

To generate O-legal profiles of CP-nets, we consider a fixed ordering $X_1, \cdots, X_n$ of features. We also take as input the maximum in-degree for each feature, k. For each CP-net in the profile we first generate its acyclic dependency graph. For each feature X_i, we randomly choose its in-degree d, $0 \leq d \leq \min\{k, i-1\}$. Next, we randomly choose d parents from the features $\{X_1, \cdots, X_{i-1}\}$. When the graph is built, we fill in the CP tables choosing randomly one element of the domain (since the domain is binary). For a PCP-net, we generate the dependency graph and CP tables as for a CP-net, and then we randomly assign probabilities to CPT rows.

We show results for some specific parameter values, similar results were observed also for other values of n and k.

In both the experiments in Figure 3 and Figure 4, we compute the mean *CopelandScore* over 100 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 3 we investigate the quality of all the voting rules, varying the number of features and fixing the number of individuals in the profile. In Figure 4 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 3) 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 $CopelandScore$ score, 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 variance in Figure 3 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 between even and odd numbers of features (and even and odd numbers of individuals in Figure 4). Our conjecture is that an odd number of features (resp. individuals) leads to more decisiveness in the voting rules.

In the second set of experiments (Figure 4) 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.

5. DOMINANCE IN PCP-NETS

In this section we analyse how *PR* performs on dominance testing. We then study algorithms to compute an approximation of the dominance value, whose exact computation is hard.

The aggregation method *PR* is also efficient 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 relative frequency of CP-nets that entail $o > o'$. In our experiments, in Table 1, 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 1: Difference between the value of dominance in the initial profile of CP-nets and in the PCP-net aggregated with PR.

Computing the probability of dominance for separable PCP-nets (with no edges in their dependency graph) is polynomial, 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 5.1) and an upper bound (Section 5.2) in polynomial time for PCP-nets where the dependency graph is a polytree.

5.1 Lower Bound

We consider only binary-valued PCP-nets that have a polytree structure. This category includes directed trees.

From O-legality, we can assume we have an order O over n features X_i such that $\forall i \in \{1, \cdots, n\}$, $Pa(X_i) \subseteq \{X_1, \cdots, X_{i-1}\}$.

Notation: Given an outcome $o = o_1 o_2 \cdots o_n$ let $o \restriction_{X_i}$ indicate o_i and $o \restriction_S$ indicate $o \restriction_{s_1} \cdots o \restriction_{s_m}$ where $S = \{s_1, \cdots, s_m\} \subseteq \{X_1, \cdots, X_n\}$ 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 $o \restriction_S > o' \restriction_S$ 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.

DEFINITION 8. *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'. The lower bound for dominance for $o > o'$ in $\mathcal{Q}$ is defined as follows:*

$$\mathbb{P}^L_{\mathcal{Q}}(o > o') = \prod_{i \in \text{Diff}} (1 - p_i)$$

where $\forall i \in$ Diff, p_i *are defined as:*

$$p_i = \begin{cases} \prod_{u \in U_i^{o,o'}} \mathbb{P}(o' \restriction_{X_i} > o \restriction_{X_i} | u) & \textit{if } \mathrm{Pa}(X_i) \neq \emptyset \\ \mathbb{P}(o' \restriction_{X_i} > o \restriction_{X_i}) & \textit{if } \mathrm{Pa}(X_i) = \emptyset \end{cases}$$

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

We require Definition 9 and the following lemmas in the proof of Theorem 2 (proofs omitted for space).

LEMMA 1. *Given an acyclic CP-net C with n nodes and two outcomes o and o', if $o \bowtie o'$ then there exists index $i \in \{1, \cdots, n\}$ such that $o \restriction_{X_1 \cdots X_j} \not\bowtie o' \restriction_{X_1 \cdots X_j} \quad \forall j < i \quad$ and $o \restriction_{X_1 \cdots X_l} \bowtie o' \restriction_{X_1 \cdots X_l} \ \forall l \geq i$.*

DEFINITION 9. *Given two oucomes o and o' such that $o \bowtie o'$, we say that they are incomparable for the index i if i satisfies Lemma 1.*

LEMMA 2. *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 $[(o \restriction_{C_1} > o' \restriction_{C_1}$ and $o \restriction_{C_2} < o' \restriction_{C_2})$ or $(o \restriction_{C_1} < o' \restriction_{C_1}$ and $o \restriction_{C_2} > o' \restriction_{C_2})]$ then $o \bowtie o'$.*

THEOREM 2. *Given a PCP-net $\mathcal{Q}$ and two outcomes o and o', $\mathbb{P}^L_{\mathcal{Q}}(o > o')$ is a lower bound for $\mathbb{P}_{\mathcal{Q}}(o > o')$.*

PROOF. Given o and o' such that $o = (x_1 x_2 \cdots x_n)$ and $o' = y_1 y_2 \cdots y_n$ the formula can be written as: $(1-p_{i_0})(1-p_{i_1}) \cdots (1-p_{i_m})$ where $i_j \in$ *Diff* are the ordered indices of variables that change value from o to o'. Computing some products we obtain the following equivalent formula:

$$[(1 - p_{i_0})] - [(1 - p_{i_0})(p_{i_1})] - [(1 - p_{i_0})(1 - p_{i_1})(p_{i_2})] - \cdots$$

$$\cdots - [(1 - p_{i_0})(1 - p_{i_1}) \cdots (1 - p_{i_{m-1}})(p_{i_m})].$$

Consider the first term: $1 - p_{i_0}$. If the variable X_{i_0} is independent then $p_{i_0} = \mathbb{P}(o' \restriction_{X_{i_0}} > o \restriction_{X_{i_0}})$, so $(1 - p_{i_0}) = \mathbb{P}(o \restriction_{X_{i_0}} > o' \restriction_{X_{i_0}})$. Otherwise the set $U_{i_0}^{o,o'}$ has only one element u. Thus $p_{i_0} = \mathbb{P}(o' \restriction_{X_{i_0}} > o \restriction_{X_{i_0}} |u)$ then $(1 - p_{i_0}) = \mathbb{P}(o \restriction_{X_{i_0}} > o' \restriction_{X_{i_0}} |u)$. So, we call q_{i_0} the probability $\mathbb{P}(o \restriction_{X_{i_0}} > o' \restriction_{X_{i_0}})$ (or $\mathbb{P}(o \restriction_{X_{i_0}} > o' \restriction_{X_{i_0}} |u)$). Then the formula is equivalent to:

$$[q_{i_0}] - [(q_{i_0})(p_{i_1})] - [(q_{i_0})(1 - p_{i_1})(p_{i_2})] - \cdots$$

$$\cdots - [(q_{i_0})(1 - p_{i_1}) \cdots (1 - p_{i_{m-1}})(p_{i_m})].$$

We note that each term of the sum contains q_{i_0}. The probability q_{i_0}, used in computing the probabilities of CP-nets induced by $\mathcal{Q}$, is the probability in $\mathcal{Q}$ that $x_{i_0} > y_{i_0}$ (or $x_{i_0} > y_{i_0}|u$) and corresponds to the probability of the set of all CP-nets that have that row in their CP-tables. Let's call this set S_0.

When we restrict to the set S_0, we restrict our attention to induced CP-nets for which $o > o'$. We exclude those for which $o' > o$ by insisting that they contain the CPT row that says that $x_{i_0} > y_{i_0}$ (or $x_{i_0} > y_{i_0}|u$). The idea is to remove from S_0 all the induced CP-nets in which $o \bowtie o'$. In each step of the sum we remove a subset of S_0. Analyzing a generic j-term of the sum:

$$[(q_{i_0})(1 - p_{i_1}) \cdots (1 - p_{i_{j-1}})(p_{i_j})]$$

we observe that we are removing from S_0 a subset $\bar{S}_j$, the set of induced CP-nets defined by $[(q_{i_0})(1 - p_{i_1}) \cdots (1 - p_{i_{j-1}})(p_{i_j})]$. We consider also the set S_j of the induced CP-nets defined by $[(q_{i_0})(1 - p_{i_1}) \cdots (1 - p_{i_{j-1}})(1 - p_{i_j})]$ and we observe that S_j and $\bar{S}_j$ are a partition of the set S_{j-1}. That implies that we do not remove the same CP-net more than once, because we are removing partitions. In particular when we remove 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).

We prove that each set $\bar{S}_j$ contains all the induced CP-nets that have $o \bowtie o'$ for index i_j, and thus the set S_j contains only CP-nets such that: $(x_1 x_2 \cdots x_{i_j} z > y_1 y_2 \cdots y_{i_j} z \ \ \forall z)$ and $(x_1 x_2 \cdots x_{i_j} z_1 \gtrless y_1 y_2 \cdots y_{i_j} z_2 \ \forall z_1 \neq z_2)$ where z, z_1, z_2 are a completion for a complete outcome over all the variables $X_1, \cdots, X_n$. (Given a partial outcome $\tilde{o}$ over variables $X_1, \cdots, X_j$ a completion of $\tilde{o}$ is an assignment to the remaining variables $X_{j+1}, \cdots, X_n$.)

We prove that, given an induced CP-net C from $\mathcal{Q}$ and two outcomes o and o', if for C they are incomparable for index i_j then $C \in \bar{S}_j$. Let o and o' be outcomes that are incomparable for index i_j for the induced CP-net C. We prove that $C \in \bar{S}_j$. We prove this by contradiction. If $C \notin \bar{S}_j$ then either (1) $C \in \bar{S}_k$ with $k < j$ or (2) $C \in S_j$. If we consider (1) $C \in \bar{S}_k$ with $k < j$, this implies that o and o' are incomparable for $i_k < i_j$ and it is a contradiction because i_j is the minimum index for the incomparability. If we consider (2) $C \in S_j$, we have that $o \restriction_{X_1, \cdots, X_{i_j - 1}} \not\bowtie o' \restriction_{X_1, \cdots, X_{i_j - 1}}$, because o and o' are incomparable for index i_j. We suppose that $o \restriction_{X_1, \cdots, X_{i_j - 1}} > o' \restriction_{X_1, \cdots, X_{i_j - 1}}$ (the case with '$<$' is symmetric). Then we prove that also $o \restriction_{X_1, \cdots, X_{i_j}} > o' \restriction_{X_1, \cdots, X_{i_j}}$, to reach a contradiction.

If X_{i_j} is an **independent node** it is easy to show that $o \restriction_{X_1, \cdots, X_{i_j}} > o' \restriction_{X_1, \cdots, X_{i_j}}$. We take a worsening path P from $o \restriction_{X_1, \cdots, X_{i_j - 1}}$ to $o' \restriction_{X_1, \cdots, X_{i_j - 1}}$: $P = o \restriction_{X_1, \cdots, X_{i_j - 1}} \rightarrow O_1 \rightarrow O_2 \rightarrow \cdots \rightarrow o' \restriction_{X_1, \cdots, X_{i_j - 1}}$. The path $\bar{P} = o \restriction_{X_1, \cdots, X_{i_j - 1}} o_{i_j} \rightarrow O_1 o_{i_j} \rightarrow O_2 o_{i_j} \rightarrow \cdots \rightarrow o' \restriction_{X_1, \cdots, X_{i_j - 1}} o_{i_j} \rightarrow o' \restriction_{X_1, \cdots, X_{i_j - 1}} o'_{i_j}$ is a worsening path from $o \restriction_{X_1, \cdots, X_{i_j}}$ $(= o \restriction_{X_1, \cdots, X_{i_j - 1}} o_{i_j})$ to $o' \restriction_{X_1, \cdots, X_{i_j}}$ $(= o' \restriction_{X_1, \cdots, X_{i_j - 1}} o'_{i_j})$.

Now we consider the case in which X_{i_j} is **not an independent node**. Because C is polytree structured, the sub-CP-net $\tilde{C}$ of the nodes $X_1, \cdots, X_{i_j - 1}$ is formed by $|Pa(X_{i_j})|$ connected components, each one containing exactly one parent of X_{i_j}. Thus, there aren't conflicts with the needs of the ancestors.

We have two cases. The **first case** is the case in which X_{i_j} has no shared parents. In this case, for each $u \in U_{i_j}^{o,o'}$ we can have a worsening path P from $o \restriction_{X_1, \cdots, X_{i_j - 1}}$ to $o' \restriction_{X_1, \cdots, X_{i_j - 1}}$ that contains u because the parents belong to different connected components and we can permute the order in which we make the changes to the single connected components. Then we have $P = o \restriction_{X_1, \cdots, X_{i_j - 1}} \rightarrow o_1 \rightarrow o_2 \rightarrow \cdots \rightarrow o_l \rightarrow \cdots \rightarrow o' \restriction_{X_1, \cdots, X_{i_j - 1}}$ where o_l contains u $(o_l \restriction_{Pa(X_{i_j})} = u)$. Thus, if a CP-net belongs to S_j, that means that there exists at least one $u \in U_{i_j}^{o,o'}$ such that in the CP-table of X_{i_j} there is the row $u : x_{i_j} > y_{i_j}$, then a worsening path from $o \restriction_{X_1, \cdots, X_{i_j}}$ to $o' \restriction_{X_1, \cdots, X_{i_j}}$ is $\tilde{P} =$

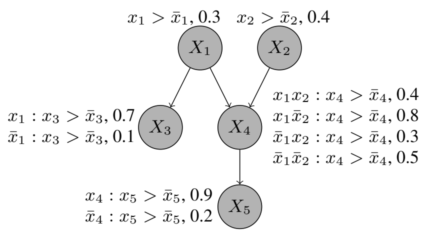


Figure 5: A PCP-net.

$o \restriction_{X_1,\cdots,X_{i_j-1}} x_{i_j} \rightarrow o_1 x_{i_j} \rightarrow o_2 x_{i_j} \rightarrow \cdots \rightarrow o_l x_{i_j} \rightarrow o_l y_{i_j} \rightarrow \cdots \rightarrow o' \restriction_{X_1,\cdots,X_{i_j-1}} y_{i_j}$.

The **second case** is the case in which X_{i_j} has shared parents. In this case X_{i_j} can have only a unique shared parent node (otherwise it introduces an undirected cycle). We suppose that also in this case, there exists a path P from $o \restriction_{X_1,\cdots,X_{i_j-1}}$ to $o' \restriction_{X_1,\cdots,X_{i_j-1}}$ that contains u, for each $u \in U_{i_j}^{o,o'}$. We build a path $\tilde{P}$ as in the previous point, but we have to consider the interaction of the other nodes that have shared parents with X_{i_j}. If X_{i_j} and X_c share a parent, X_p, there are two cases: *(A)* X_c needs the same value for X_p as X_{i_j}. In this case we can change the value of X_{i_j} and X_c in an arbitrary order. *(B)* X_c needs a different value for X_p as X_{i_j}. In this case in the path the value of X_p changes, for example, from a value x_p to $\bar{x}_p$. If X_c needs x_p then first we change its value then we change X_p and finally X_{i_j}. If X_c needs $\bar{x}_p$ then first we change the value of X_{i_j} then we change X_p and finally X_c. It is possible because X_{i_j} and X_c have only a shared parent. If the two nodes have more than one shared parent maybe they need two incomparable assignments of the shared parent and in that case there does not exist a path between them. Thus $C \in \bar{S}_j$. $\square$

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

EXAMPLE 3. *Consider a PCP-net $\mathcal{G}$ as defined in Figure 5 and the outcomes $o = x_1 x_2 \bar{x}_3 x_4 x_5$ and $o' = x_1 \bar{x}_2 x_3 x_4 \bar{x}_5$. The lower bound is:* $\mathbb{P}^L_{\mathcal{G}}(o > o') = [1 - \mathbb{P}(\bar{x}_2 > x_2)] \cdot [1 - \mathbb{P}(x_3 > \bar{x}_3 | x_1)] \cdot [1 - \mathbb{P}(\bar{x}_5 > x_5 | x_4)] = 0.4 \cdot 0.3 \cdot 0.9 = 0.108$.

We observe that 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. First we prove the following theorem.

THEOREM 3. *Given a PCP-net $\mathcal{Q}$ over n features $X_1, \cdots, X_n$ with binary domains $\mathcal{D}_i = \{x_i, \bar{x}_i\} \forall i \in \{1, \cdots, n\}$ and given two outcomes o and o' then*

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

with the convention that for a feature X_i with domain $\mathcal{D}_i = \{x_i, \bar{x}_i\}$ we have $p_i = \mathbb{P}(x_i \succ \bar{x}_i)$ in the PCP-net (and thus $(1 - p_i) = \mathbb{P}(\bar{x}_i \succ x_i)$) and where Diff$^+$ *and* Diff$^-$ *are a partition of* Diff. Diff$^+$ *is the set of indexes of variables such that $o \restriction_{X_i} = x_i$ and $o' \restriction_{X_i} = \bar{x}_i$ and* Diff$^-$ *is the set of indexes of variables such that $o \restriction_{X_i} = \bar{x}_i$ and $o' \restriction_{X_i} = x_i$.*

Now we can prove the following theorem.

THEOREM 4. *Given a PCP-net $\mathcal{Q}$ with a separable dependency graph and two outcomes o and o', then $\mathbb{P}_{\mathcal{Q}}(o > o') = \mathbb{P}^L_{\mathcal{Q}}(o > o')$.*

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 3. 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 $o \restriction_{X_i} = x_i$ and $o' \restriction_{X_i} = \bar{x}_i$. In the formulation of dominance for separable PCP-nets (Theorem 3), we have a factor that corresponds to q_i because the variable X_i belongs to the set *Diff*$^+(o, o')$. In the lower bound formulation we have a factor that corresponds to:

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

The second case is the case of a variable X_i such that $o \restriction_{X_i} = \bar{x}_i$ and $o' \restriction_{X_i} = x_i$. In the formulation of Theorem 3, we have a factor that corresponds to $(1 - q_i)$ because the variable X_i belongs to the set *Diff*$^-(o, o')$. In the lower bound formulation we have a factor that corresponds to:

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

Thus, the two formulations coincide. $\square$

5.2 Upper Bound

We say that a feature X in a PCP-net has a *fixed ancestor* over two outcomes o and o' if X is independent or all the variables $Y \in Anc(X)$ (ancestors of X) have the same value in o and o': $o \restriction_Y = o' \restriction_Y \; \forall Y \in Anc(X)$. We denote this set as the set of all fixed-ancestor variables FA. As before let *Diff* be the set of the indexes of features that have different value on o and o'.

DEFINITION 10. *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:*

$$\mathbb{P}^U_{\mathcal{Q}}(o > o') = \prod_{j \in (\text{Diff} \cap \text{FA})} \mathbb{P}(o \restriction_{X_j} > o' \restriction_{X_j} | u)$$

where u is the assignment of the parents: $u = o \restriction_{\text{Pa}(X_j)} = o' \restriction_{\text{Pa}(X_j)}$.

The formula above is a product of at most n factors, whose computation takes $O(1)$, hence $O(n)$ in total.

THEOREM 5. *Given a PCP-net $\mathcal{Q}$ and two outcomes o and o', $\mathbb{P}^U_{\mathcal{Q}}(o > o')$ is an upper bound for $\mathbb{P}_{\mathcal{Q}}(o > o')$.*

PROOF. It is equivalent to prove that all CP-nets induced by $\mathcal{Q}$ that support $o > o'$ have the row $u : o \restriction_{X_j} > o' \restriction_{X_j}$ for all the variables $X_j \in$ (*Diff* $\cap$ *FA*). We prove this sentence by contradiction: we consider an induced CP-net C that contains the row $u : o' \restriction_X > o \restriction_X$ for a $X \in$ (*Diff* $\cap$ *FA*) and we prove that $C \models (o' > o) \vee (o \bowtie o')$.We have two cases: *(1)* X is an independent node. Then the flip $o \restriction_X \rightarrow o' \restriction_X$ 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 $Anc(X) = \{X_{i_1}, \cdots, X_{i_k}\}$ with $k \geq 1$ and all of them have fixed value on o and o'. We call u the assignment of *Pa*(X) in o and o'. Thus, each worsening path from

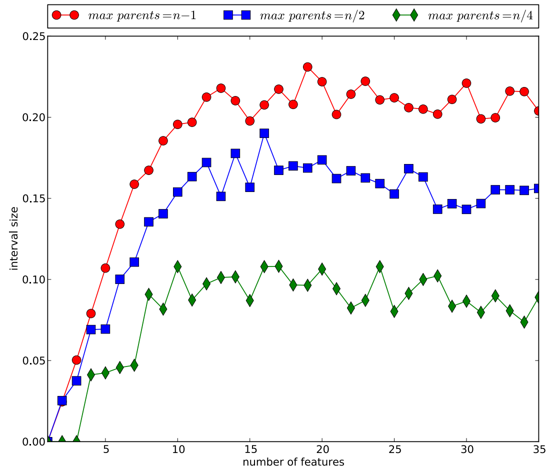


Figure 6: Interval size vs. number of features.

o has the value u for $Pa(X)$ in each step of the path, because all the ancestors of X are fixed. This implies that no worsening path from 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 $C \models (o' > o) \vee (o \bowtie o')$. □

EXAMPLE 4. *Consider the PCP-net $\mathcal{G}$ and the two outcomes o and o' as in Example 3.* Diff $= \{2, 3\}$ *as they have different values on o and o' and fixed ancestors. The variable X_5 has different values on o and o' and its ancestor X_2 has different values in o and o'. Thus the upper bound is:* $\mathbb{P}^U_{\mathcal{G}}(o > o') = \mathbb{P}(x_2 > \bar{x}_2) \cdot \mathbb{P}(\bar{x}_3 > x_3|x_1) = 0.4 \cdot 0.3 = 0.12.$

5.3 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. We tested experimentally the size of the interval varying the number of features and, fixing the number of features, varying the maximal number k of parents that a feature can have.

We generate polytree structured CP-nets and PCP-nets as we do for general acyclic CP-nets and PCP-nets. However, when we need to generate parents, for each node X_i we add parents, one by one, using rejection sampling on $X_1, \cdots, X_{i-1}$, rejecting those that add cycles in the underlying *undirected* graph, until we reach the in-degree or exhaust the set of possible parents.

Figure 6 shows the size of the interval as a function of the number of features, n, and of k. In this experiment we vary $n \in [0, 35]$ and fix the maximum k to $n - 1$, $n/2$ and $n/4$. We compute the mean of the dominance interval over 100 PCP-nets for each value of n and for each PCP-net we take the mean over 100 outcome pairs. We observe that the mean interval size is small.

Table 2 shows the distance between our lower bound and the true value of the dominance probability 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 20 PCP-nets for each value of n and for each PCP-net we take the mean over 25 outcome pairs (total mean over 100 cases). Observe the true value is very close to the lower bound and the maximal distance is 0.0034.

In our experiments, the distance between the lower bound to the true dominance probability is generally very small. It seems to grow as a function of the number of nodes and parents. Thus, the lower bound can be considered a good approximation of the true dominance probability with a maximal error equal to the size of the interval.

$n:$	1	2	3	4	5	6	7
$Pa: n-1$	0.0	0.0	0.002	0.002	0.002	0.008	0.004
$Pa: n/2$	0.0	0.0	0.008	0.006	0.0	0.004	0.004
$Pa: n/4$	0.0	0.0	0.0	0.0	0.0034	0.002	0.006

Table 2: Distance between our lower bound and the true probability of dominance.

5.4 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.We define the following notion of approximate dominance: *MP-dominance*.

DEFINITION 11 (MP-DOMINANCE FOR PCP-NETS). *Given a PCP-net $\mathcal{Q}$ and a pair of outcomes o and o', a* most probable dominance query (MP-dominance) *asks a* dominance query *to the most probable CP-net induced by $\mathcal{Q}$.*

Given a profile of CP-nets and two outcomes, we compare MP-dominance queries on the PCP-net aggregated with *PR* with the answers to the true/false dominance queries on both (1) the initial profile of CP-nets and (2) the profile of CP-nets induced by the PCP-net. A profile of CP-nets returns $True$ if the frequency of CP-nets that entail dominance is greater then the frequency of those that do not, and $False$ otherwise. Our experiments (results omitted for space) show that the percentage of times these different methods give the same yes/no result is above 90%. This supports the fact that aggregating CP-nets with the *PR* method to get a PCP-net is also reasonable for dominance seen as a decision-making problem.

6. CONCLUSIONS

In this paper we evaluated the use of PCP-nets as a compact representation language for the preferences of a set of agents. Starting from a profile of individual CP-nets, we introduced and evaluated two aggregation methods for the definition of a PCP-net, a first one based on relative frequencies of pairwise preferences (*PR*) and a second one based on the exact distribution of CP-nets (*LS*). Our theoretical and experimental results suggest that using the *PR* method in the input profile to construct a PCP-net is accurate with respect to answering both optimality and dominance queries. Since optimality queries under this method can be shown to be equivalent to performing sequential voting, our proposed aggregation method is a direct generalisation of this setting. By generating a compact representation of the full preference profile using PCP-nets, we are also able to perform either exact or approximate dominance reasoning on a profile of individual CP-nets. Moreover, for the case of polytree PCP-nets, we showed that our proposed approximation techniques yield results that are very close to the probability of dominance in the initial profile of individual CP-nets.

Acknowledgments

The work of Cristina Cornelio, Umberto Grandi and Francesca Rossi was partially supported by the University of Padova project "Incorporating patients' preferences in kidney transplant decision protocols." NICTA is funded by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Centre of Excellence Program.

REFERENCES

[1] T. Allen, J. Goldsmith, and N. Mattei. Counting, ranking, and randomly generating CP-nets. In *Proc. of the 8th Multidisciplinary Workshop on Advances in Preference Handling (MPREF)*, 2014.

[2] D. Bigot, H. Fargier, J. Mengin, and B. Zanuttini. Probabilistic conditional preference networks. In *Proc. of the 29th International Conference on Uncertainty in Artificial Intelligence (UAI)*, 2013.

[3] S. Bistarelli, U. Montanari, and F. Rossi. Semiring-based constraint satisfaction and optimization. *Journal of the ACM (JACM)*, 1997.

[4] C. Boutilier, R. Brafman, C. Domshlak, H. Hoos, and D. Poole. CP-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements. *Journal of Artificial Intelligence Research (JAIR)*, 21:135–191, 2004.

[5] R. Brafman and C. Domshlak. Preference handling-an introductory tutorial. *AI Magazine*, 30(1):58, 2009.

[6] Y. Chevaleyre, U. Endriss, J. Lang, and N. Maudet. Preference handling in combinatorial domains: From AI to social choice. *AI Magazine*, 29(4):37, 2008.

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

[8] C. Cornelio, U. Grandi, J. Goldsmith, N. Mattei, F. Rossi, and K. Venable. Voting with CP-nets using a probabilistic preference structure. In *Proc. of the 5th International Workshop on Computational Social Choice (COMSOC)*, 2014.

[9] C. Domshlak and R. Brafman. CP-nets: Reasoning and consistency testing. In *Proc. 8th International Conference on Principles of Knowledge Representation and Reasoning (KR)*, 2002.

[10] C. Domshlak, F. Rossi, K. Venable, and T. Walsh. Reasoning about soft constraints and conditional preferences: complexity results and approximation techniques. In *Proc. of the 18th International Joint Conference on Artificial Intelligence (IJCAI)*, 2003.

[11] J. Fürnkranz and E. Hüllermeier. *Preference Learning: An Introduction*. Springer, 2010.

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

[13] C. Gonzales and P. Perny. GAI networks for utility elicitation. In *Proc. of the 10th International Conference on Principles of Knowledge Representation and Reasoning (KR)*, 2004.

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

[15] M. Li, Q. Vo, and R. Kowalczyk. An efficient procedure for collective decision-making with CP-nets. In *Proc. of the 19th European Conference on Artificial Intelligence (ECAI)*, 2010.

[16] M. Li, Q. Vo, and R. Kowalczyk. 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)*, 2011.

[17] A. Maran, N. Maudet, M. Pini, F. Rossi, and K. Venable. A framework for aggregating influenced CP-nets and its resistance to bribery. In *Proc. of the 27th AAAI Conference on Artificial Intelligence (AAAI)*, 2013.

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

[19] N. Mattei, M. S. Pini, F. Rossi, and K. B. Venable. Bribery in voting over combinatorial domains is easy. In *Proc. 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2012.

[20] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[21] F. Rossi, K. Venable, and T. Walsh. mCP-nets: representing and reasoning with preferences of multiple agents. In *Proc. of the 19th AAAI Conference on Artificial Intelligence (AAAI)*, 2004.

[22] F. Rossi, K. Venable, and T. Walsh. *A Short Introduction to Preferences: Between Artificial Intelligence and Social Choice*. Morgan & Claypool Publishers, 2011.

[23] G. R. Santhanam, S. Basu, and V. Honavar. Dominance testing via model checking. In *Proc. of the 24th AAAI Conference on Artificial Intelligence (AAAI)*, 2010.

[24] L. Xia, V. Conitzer, and J. Lang. Voting on multiattribute domains with cyclic preferential dependencies. In *Proc. of the 23rd AAAI Conference on Artificial Intelligence (AAAI)*, 2008.

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