The Thirty-Third International
FLAIRS Conference (FLAIRS-33)

A Lexicographic Strategy for Approximating Dominance in CP-Nets

Michael Huelsman
University of Kentucky
Lexington, Kentucky, USA
michael.huelsman@uky.edu

Mirosław Truszczyński
University of Kentucky
Lexington, Kentucky, USA
mirek@cs.uky.edu

Abstract

To represent and reason about preferences over elements of a combinatorial domain it is necessary to have a compact preference model. One of the most extensively studied models for that setting is the conditional preference network (CP-net). A major problem with CP-nets is that some tasks that are critical to decision making are computationally hard if the preference ordering is given by a CP-net. To overcome this difficulty we propose to approximate CP-nets with other concise preference models that are equally intuitive but have better computational properties. In this paper, we focus on approximations of CP-nets using modified lexicographic preference models (LPMs). We show an acyclic CP-net's dominance relation can be approximated in polynomial time and present several results on the accuracy of the approximation.

Introduction

An agent's decisions are determined by their preferences. In many cases decisions are made over many possible alternatives and the sheer number of alternatives may lay a large cognitive burden on the agent. In order to provide support for decision making one needs to have a compact method of representing preferences over large domains. Over the years many representations have been studied. Conditional preference networks (CP-nets) (Boutilier et al. 2004) have remained a popular choice for study. CP-nets have an intuitive, principled approach for their construction, encode conditional preference information, and allow for polynomial time computation of optimal and pessimal alternatives (Boutilier et al. 2004). These advantages are not without drawbacks. The dominance problem, determining which of two alternatives is preferred, is computationally hard for CP-nets. This holds even for CP-nets with acyclic dependency graphs, where dominance is NP-hard (Boutilier et al. 2004). In general, dominance testing for CP-nets is PSPACE-complete (Goldsmith et al. 2008).

This work looks at efficiently approximating the dominance problem for acyclic CP-nets. We build our approximation by extracting importance information from the dependency graph of a CP-net. In short, we assume that attributes whose preferences are "less" conditionally dependent, either directly or indirectly, are more important to decision making. By combining this importance information with the conditional preference tables of the CP-net we can build a different model, called a *conditional lexicographic preference model* (CLPM) (Huelsman and Truszczyński 2019), which is similar to several models such as lexicographic next-value predicates (Wilson 2014), leximin evaluation of possibilistic CP-nets (Dubois, Prade, and Touazi 2013), and lexicographic preference trees (Booth et al. 2010).

For CLPMs dominance testing is in P, but a CLPM built from a CP-net only provides an *approximation* of the CP-net's dominance relation. It reproduces *all* strict dominance relations in the original CP-net, but may make some incomparable alternatives comparable. In other words, the approximating CLPM preference relation *strengthens* the CP-net preference relation. The approximation can be improved by considering multiple approximating CLPMs and, in fact, even all such models, which maximizes the accuracy of the approximation. We show that dominance testing is still in polynomial time when considering all applicable CLPMs.

This work contains five sections. The second section contains background information, the third section contains our results, the fourth section discusses related works, and the fifth section concludes and introduces further discussion.

Background

Preference reasoning deals with ordering objects. A *preorder* on a set U is defined by any reflexive and transitive binary relation, denoted $\succeq$. It becomes an *order* (on U) if it is antisymmetric. A preorder is *total* if for every pair of elements $a, b \in U$ either $a \succeq b$ or $b \succeq a$. We define the *strict* counterpart to $\succeq$, $\succ$, by setting $a \succ b$ if and only if $a \succeq b$ and $b \not\succeq a$. Similarly, the associated *incomparability* relation $\bowtie$ is defined by setting $a \bowtie b$ if and only if $a \not\succeq b$ and $b \not\succeq a$. To avoid confusion we may denote incomparability by $\|$.

A function $r : U \to [1 \dots k]$ is a *ranking* if it is an onto function, that is, for any $i \in [1 \dots k]$ there exists at least one alternative α such that $r(\alpha) = i$. Each ranking function determines a preorder $\succeq_r$ on U by setting $\alpha \succeq_r \beta$ precisely when $r(\alpha) \leq r(\beta)$. Conversely, each preorder $\succeq$ on U determines a ranking r such that $\succeq$ and $\succeq_r$ are the same. Given a preorder $\succeq$, we say that a ranking r is *consistent* with $\succeq$ if $\alpha \succ b$ implies $r(\alpha) < r(\beta)$.

A *combinatorial domain* is defined by a set of *attributes*,

$\mathcal{V} = \{v_1, v_2, \ldots, v_n\}$, and their domains $D_1, D_2, \cdots, D_n$ (the finite sets of values they can take). The shorthand $D(v_i)$ denotes the domain of an attribute v_i. A specific alternative α, is given by a tuple with one value from each attribute.

$$\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_n) \in \mathcal{D}(v_1) \times \mathcal{D}(v_2) \times \cdots \times \mathcal{D}(v_n).$$

For example, the space of pizzas can formally be described as a combinatorial domain, as follows:

$$\mathcal{V} = \{\text{Sauce,Meat,Vegetable}\}$$
$$\mathcal{D}(\text{Sauce}) = \{\text{Tomato, Alfredo, Cheese}\}$$
$$\mathcal{D}(\text{Meat}) = \{\text{None, Sausage, Pepperoni}\}$$
$$\mathcal{D}(\text{Vegetable}) = \{\text{None, Spinach, Mushrooms}\}.$$

The tuple (Tomato,Sausage,Spinach) represents a pizza with tomato sauce, sausage, and spinach. The size of the domain is $3^3 = 27$. This domain is small, but the size of a combinatorial domain, in general, is exponential in the number of attributes. Thus, explicit preference representations become impractical and a subset of attributes must be considered. We denote the projection of an alternative α onto a set of attributes A as $\alpha[A]$. If we are projecting onto a single attribute a we denote $\alpha[\{a\}]$ as $\alpha[a]$ for simplicity.

Many preference representations for combinatorial domains have been proposed. Two popular models are *conditional preference networks* (CP-nets) (Boutilier et al. 2004) and *lexicographic preference models* (LPM) (Fishburn 1974). Huelsman and Truszczyński introduced a model called *conditional lexicographic preference models* (CLPM), which are also similar to LP-trees (Booth et al. 2010) (which will not be discussed here). This work extends the initial work done on CLPMs and CP-nets with several novel results.

CP-nets capture conditional preferences in a graphical manner. Preferring white wine with fish and red wine with beef is a conditional preference because the choice of wine is dependent on the protein being served. CP-nets represent preferences using a "ceteris paribus", or all other things being equal semantics (Boutilier et al. 2004). Formally, a CP-net on a set $\mathcal{V}$ of attributes is a triple $C = (\mathcal{V}, E, T)$, where $(\mathcal{V}, E)$ defines the *dependency* graph, which specifies for each attribute $v \in \mathcal{V}$ those attributes that v depends on, with an edge from v' to v indicating that preferences over the values of v depend on the value of v', and T is a set of conditional preference tables (CPTs). The set T of CPTs contains a CPT for each attribute in $\mathcal{V}$. A CPT for an attribute v contains an entry for each possible value assignment of the attributes that v depends on. Each entry gives a strict total order over the domain of v. For an alternative α, $\succ_{v,\alpha}$ denotes the order in the entry of the CPT table for v determined by the values in α of the attributes that v depends on.

Consider a CP-net $C = (\mathcal{V}, E, T)$. An alternative α is *locally cp-preferred* to an alternative β if β is identical to α except on a single attribute v and $\alpha[v] \succ_{v,\alpha} \beta[v]$. We could also use $\succ_{v,\beta}$ because α and β coincide outside of v thus, $\succ_{v,\alpha}$ and $\succ_{v,\beta}$ are the same. Informally, α is locally cp-preferred to β if the two differ on exactly one attribute and on that attribute α has a better value than β based on

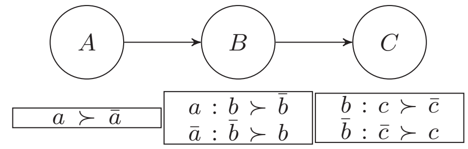

Figure 1: A CP-Net

that attribute's CPT. We denote this relation by $\alpha \rightarrow_C \beta$ and refer to the transformation of α into β as a *worsening* flip.

The preference relation of a CP-net C, called *cp-preference* and written $\succeq_C$, is the transitive closure of the relation $\rightarrow_C$. In other words, for two alternatives α and β, $\alpha \succeq_C \beta$ if and only if there is a sequence of worsening flips in C that starts in α and ends in β.

A CP-net is *acyclic* if its dependency graph is acyclic. For each acyclic CP-net C, the relation $\succeq_C$ is an order (nonempty cycles of worsening flips are impossible). In this work, we only consider acyclic CP-nets. We write $\succ_C$ and $\bowtie_C$ for the corresponding strict order and incomparability relations for $\succeq_C$.

Using the CP-net in Fig. 1 to compare the alternatives (a, b, c) and $(\bar{a}, b, \bar{c})$ we look for a sequence of worsening flips from (a, b, c) to $(\bar{a}, b, \bar{c})$. One such sequence is $(a, b, c) \rightarrow_C (\bar{a}, b, c) \rightarrow_C (\bar{a}, b, \bar{c})$. Thus, $(a, b, c) \succeq_C (\bar{a}, b, \bar{c})$.

We move on to another model of interest in this work. Let $\mathcal{V}$ be the set of attributes of a combinatorial domain. A *conditional* lexicographic preference model (CLPM) is a tuple $\pi = (\mathcal{V}, E, T, r)$, where $\mathcal{V}$, E and T are as in CP-nets (that is, $(\mathcal{V}, E, T)$ *is* a CP-net), and r is an importance ranking of attributes satisfying the *consistency* requirement: for every pair of attributes $v, w \in \mathcal{V}$ if $(v, w) \in E$ (in other words, w depends on v), $r(v) < r(w)$. A CLPM $\pi = (\mathcal{V}, E, T, r)$ defines the dominance relation by considering attributes according to their importance and taking into account conditional preference information contained in CPTs in T.

Definition 1. *Given a CLPM $\pi = (\mathcal{V}, E, T, r)$, $\alpha \succeq_\pi \beta$ if $\alpha = \beta$, or if for some attribute $v \in \mathcal{V}$ (i) $\alpha[v] \succ_{v,\alpha} \beta[v]$, (ii) for all attributes $w \in \mathcal{V}$ such that $r(w) \leq r(v)$, $\alpha[w] \succeq_{w,\alpha} \beta[w]$, and (iii) for all attributes $u \in \mathcal{V}$ such that $r(u) < r(v)$, $\alpha[u] = \beta[u]$.*

The notion of a CLPM generalizes a lexicographic preference model(LPM) (Fishburn 1974). An LPM is a CLPM with a strict attribute ranking and with no conditional dependencies between attributes. The following result illustrates that CLPMs are a valid preference representation.

Proposition 1. *For every CLPM π, $\succeq_\pi$ is an order. If the ranking of attributes in π is strict, $\succeq_\pi$ is a total order.*

We write $\succ_\pi$ and $\bowtie_\pi$ for the strict order and incomparability relations associated with $\succeq_\pi$. A key property of CLPMs is that the dominance problem (determining the preference relation between two alternatives) can be solved in polynomial time. This is contrasted by CP-nets, where the problem is much harder (unless, of course, P=NP).

Theorem 1. *There is a polynomial time algorithm that, given a CLPM π and two alternatives α and β, decides* *whether $\alpha \succeq_\pi \beta$ (in other words, solves the dominance problem for CLPMs).*

A CLPM $(\mathcal{V}, E, T, r)$ can be given as a pair (C, r) where C is a CP-net $(\mathcal{V}, E, T)$ and r is a ranking on $\mathcal{V}$ consistent with attribute dependencies in C. We call such CLPMs *consistent* with C. They can be viewed from the perspective of possibilistic logic (Dubois, Prade, and Touazi 2013). Specifically, a CLPM consistent with a CP-net C produces the same order as the leximin evaluation of a certain possibilistic encoding of C. In general, there may be several rankings of the attributes of a CP-net consistent with the attribute dependencies of that CP-net.

Definition 2. *The set $CLPM(C)$ for a CP-net $C = (\mathcal{V}, E, T)$ consists of all CLPMs (C, r) such that r is a ranking on $\mathcal{V}$ consistent with attribute dependencies in C (or, simply, all CLPMs consistent with C).*

We focus on using CLPMs from $CLPM(C)$ to approximate the cp-preference relation $\succeq_C$ defined by a CP-net C. We also define a preference relation over sets of CLPMs in $CLPM(C)$.

Definition 3. *Let $\mathcal{L}$ be a nonempty subset of $CLPM(C)$. The aggregated preference relation given by $\mathcal{L}$, $\succeq_\mathcal{L}$ is the intersection of all relations $\succeq_\pi$, for $\pi \in \mathcal{L}$. That is, $\alpha \succeq_\mathcal{L} \beta$ if and only if $\alpha \succeq_\pi \beta$ holds for every $\pi \in \mathcal{L}$*

Informally, $\alpha \succeq_\mathcal{L} \beta$ if α is preferred to β unanimously by all preference orders $\succeq_\pi$ (with $\pi \in \mathcal{L}$). It is clear that $\succeq_\mathcal{L}$ is an order. It is also clear that for the corresponding incomparability relation $\bowtie_\mathcal{L}$ we have $\alpha \bowtie_\mathcal{L} \beta$ if and only if there are CLPMs π and π' in $\mathcal{L}$ such that $\alpha \succ_\pi \beta$ and $\beta \succ_{\pi'} \alpha$, or $\beta \bowtie_\pi \alpha$. When $\mathcal{L} = CLPM(C)$ ($\mathcal{L}$ consists of all CLPMs in $CLPM(C)$), we write $_C$ and $||_C$ for the aggregated and incomparability relations.

Results

We first look at the relationship between a CP-net C and a single CLPM consistent with C. Afterwards, we apply what we know about individual CLPMs consistent with C to the relation aggregating several CLPMs consistent with C.

Lemma 1. *Let C be a CP-net and let $\pi \in CLPM(C)$. For every pair of alternatives α and β, if $\alpha \to_C \beta$ then $\alpha \succ_\pi \beta$.*

Lem. 1 shows that a consistent CLPM replicates the local cp-preferences of the CP-net. We also know, by Prop. 1, that a CLPM produces a transitive relation. It follows that a consistent CLPM strengthens the $\succ$ relation of a CP-net.

Theorem 2. *Let C be a CP-net and let $\pi \in CLPM(C)$. Then for every two alternatives, if $\alpha \succ_C \beta$ then $\alpha \succ_\pi \beta$.*

Proof. Consider an arbitrary worsening flipping sequence of alternatives, according to C: $o_1, o_2, o_3, \cdots, o_k$. This means that $o_i \to_C o_{i+1}$, and by transitivity $o_1 \succ_C o_k$. Since π is a consistent CLPM we know by Lem. 1 that $o_i \succ_\pi o_{i+1}$. Furthermore, given that $\succ_\pi$ is transitive, by Prop. 1, we know that $o_1 \succ_\pi o_k$. $\square$

Corollary 1. *Let C be a CP-net and let $\pi \in CLPM(C)$ then for every two alternatives α and β, if $\alpha \bowtie_\pi \beta$ then $\alpha \bowtie_C \beta$.*

Thm. 2 and Cor. 1 show the accuracy guarantees we can make about the relation $\succ_\pi$ when approximating $\succ_C$, where C is a CP-net and $\pi \in CLPM(C)$. Simply, $\succ_\pi$ overestimates $\succ_C$ and $\bowtie_\pi$ underestimates $\bowtie_C$. All errors are of one type: expressing dominance under π when the underlying CP-net C produces incomparability. We write $E(C, \pi)$ for the set of errors. The quantity $|E(C, \pi)|$ is related to the Kendall's τ distance modified to the case of preorders (Kendall 1938; Loreggia et al. 2018) but we will not pursue this connection here.

Orders defined by CLPMs consistent with a CP-net C may make considering some CLPMs in $CLPM(C)$ redundant. We can identify some of these redundancies using the definition below.

Definition 4. *A ranking r' of elements in a set U is a strict extension of a ranking r of U if for all pairs of elements $a, b \in U$ if $r(a) < r(b)$ then $r'(a) < r'(b)$ and there exists elements $a', b' \in U$ such that $r(a') = r(b')$, $r'(a') < r'(b')$.*

A useful property of strict extension is that it preserves consistency wrt a CP-net.

Proposition 2. *Given a CP-net $C = (\mathcal{V}, E, T)$ if r is an importance ranking of $\mathcal{V}$ consistent with C and r' is a strict extension of r then r' is also an importance ranking of $\mathcal{V}$ that is consistent with C.*

This observation leads to the following theorem and its two corollaries that describe the relation between the accuracy of CLPMs given by two rankings, one being an extension of the other.

Theorem 3. *Let $\pi = (C, r)$ be a CLPM in $CLPM(C)$ and let r' is a strict extension of r. Then $\pi' = (C, r')$ is a CLPM in $CLPM(C)$ and, if $\alpha \succ_\pi \beta$, $\alpha \succ_{\pi'} \beta$ holds, too.*

Corollary 2. *Let $\pi = (C, r)$ be a CLPM in $CLPM(C)$ and let r' is a strict extension of r. Then $\pi' = (C, r')$ is a CLPM in $CLPM(C)$ and, if $\alpha \bowtie \pi' \beta$, $\alpha \bowtie_\pi \beta$ holds, too.*

Corollary 3. *Given a CP-net C, a CLPM $\pi = (C, r)$ and a strict extension r' of r, $\pi' = (C, r') \in CLPM(C)$ and $E(C, \pi) \subseteq E(C, \pi')$.*

These results show that rankings that are strict extensions of other rankings can inflate the number of errors, see Cor. 3. CLPMs based on such rankings do not offer any additional information about the order $>_C$ over what can be gleaned out of CLPMs given by "non-extending" rankings.

In general, incomparabilities produced by a CLPM come in sets, as shown by the following result.

Proposition 3. *Given a CLPM $\pi = (\mathcal{V}, E, T, r)$, and two alternatives α and β such that $\alpha \bowtie_\pi \beta$ and the incomparability is decided on rank i, then for every two alternatives γ and ζ if $\gamma[V] = \alpha[V]$ and $\zeta[V] = \beta[V]$, where $V = \{v \in \mathcal{V} | r(v) \leq i\}$, it follows that $\gamma \bowtie_\pi \zeta$.*

Prop. 3 and Cor. 1 imply a fast method to identify some pairs of incomparable alternatives wrt the order defined by a CP-net. This method can be further strengthened by considering the order obtained by aggregating all CLPMs in $CLPM(C)$. Thm. 2, Cor. 1 and Def. 3 lead to the following result:

Theorem 4. *Let C be an acyclic CP-net and $A \subseteq B \subseteq CLPM(C)$. For every pair of alternatives α, β, if $\alpha \bowtie_A \beta$ then $\alpha \bowtie_B \beta$.*

Thm. 4 shows that the larger the subset of $CLPM(C)$ used the more incomparabilities are correctly decided. Since errors only occur when incomparable alternatives, according to the CP-net, are rendered comparable the greater the number of incomparabilities generated by an aggregation, the more accurate that approximation. This means that aggregation over the entire set $CLPM(C)$ produces the most accurate approximation.

To estimate $|CLPM(C)|$, let us consider a CP-net whose dependency graph has no edges. In this case, any ranking of the attributes is trivially consistent with the dependencies. The number of only strict rankings is $|\mathcal{V}|!$. In fact, one can show that even the number of CLPMs given by rankings that are minimal wrt strict extension is exponential in $|\mathcal{V}|$. Thus, a naive approach to computing the aggregated order given by all CLPMs in $CLPM(C)$ is, in general, infeasible.

Algorithm 1 The aggregated lexicographic evaluation of a CP-net $C = (\mathcal{V}, E, T)$, and two alternatives α and β, $\alpha \neq \beta$.

procedure LEX-EVAL(α, β, C)
  $c \leftarrow 0$
  $r \leftarrow \emptyset$
  while $\exists v \in \mathcal{V}$ such that IN-DEG$(v) = 0$ and $\alpha[v] = \beta[v]$ **do**
    for all $v \in \mathcal{V}$ where IN-DEG$(v) = 0$ and $\alpha[v] = \beta[v]$ **do**
      Remove v from $(\mathcal{V}, E)$
      $r(v) \leftarrow c$
      $c \leftarrow c + 1$
    end for
  end while
  $d \leftarrow \emptyset$
  for all $v \in \mathcal{V}$ where IN-DEG$(v) = 0$ **do**
    $d \leftarrow d \cup \{v\}$
  end for
  $s \leftarrow \emptyset$
  for all $v \in d$ **do**
    if ($\alpha[v] \succ_{v,\alpha} \beta[v]$ and $s = <_C$) or ($\beta[v] \succ_{v,\alpha} \alpha[v]$ and $s = >_C$) **then**
      Return $\|_C$
    else if $\alpha[v] \succ_{v,\alpha} \beta[v]$ and $s = \emptyset$ **then**
      $s \leftarrow >_C$
    else if $\beta[v] \succ_{v,\alpha} \alpha[v]$ and $s = \emptyset$ **then**
      $s \leftarrow <_C$
    end if
  end for
  Return s
end procedure

This difficulty can be overcome. We show that although obtaining an explicit representation of the approximation $\succeq_C$, by aggregating all CLPMs consistent with a CP-net C, may be infeasible that we can compute, in polynomial time, the dominance relation for that approximation. The method is shown in Algorithm 1. The following result shows that each pair of alternatives that are incomparable wrt $\|_C$ can be identified using a single CLPM, rather than two CLPMs consistent with C (cf. the discussion after Def. 3).

Theorem 5. *Given a CP-net C, two CLPMs $\pi, \pi' \in CLPM(C)$, and two alternatives α and β if $\alpha \succ_\pi \beta$ and $\beta \succ_{\pi'} \alpha$ then there exists a consistent CLPM $\pi'' \in CLPM(C)$ such that $\alpha \bowtie_{\pi''} \beta$.*

Thm. 5 shows we need not search for two CLPMs which show contradictory relations between a pair of alternatives to prove two elements are incomparable because we can instead search for a single consistent CLPM that demonstrates incomparability. LEX-EVAL tries to find such a CLPM in $CLPM(C)$ for the given alternatives. If none can be found then by Def. 3 and Thm. 5, all CLPMs in $CLPM(C)$ agree on how to order the alternatives.

Informally, LEX-EVAL builds a ranking from a CP-net's dependency graph by removing sets of attributes which have an in-degree of 0, removing them, and giving them the next available rank. We call the set of attributes removed in each iteration d. We proceed in this way as long as α and β have the same values on all attributes in the (current) d. Once this is not the case (which must happen since $\alpha \neq \beta$), we assign these attributes the next available rank. It is clear that the partial ranking built in this way contains enough information to decide how α and β are related by $>_C$ since no matter how the partial ranking is completed, the attributes in the last d are sufficient to decide dominance as α and β must differ over d. If for every attribute in d, α is preferred to β according to the CPTs we return $\alpha >_C \beta$. If for every attribute, β is preferred to α we return $\beta >_C \alpha$. Otherwise, we return $\alpha ||_C \beta$.

Theorem 6. *Given a CP-net $C = (\mathcal{V}, E, T)$ and two alternatives α and β* LEX-EVAL *correctly decides dominance, with respect to the aggregated lexicographic evaluation, for two alternatives α and β such that $\alpha \neq \beta$.*

The algorithm takes time linear in the size of the dependency graph, assuming constant time table lookup. Note that CPTs are part of the CP-net and as long as table lookup is polynomial in the size of the table our algorithm takes polynomial time, in the size of the CP-net, and thus is a polynomial time approximation of an NP-hard problem.

Proposition 4. LEX-EVAL *takes time linear, $\mathcal{O}(|V|+|E|)$, in the size of the dependency graph, assuming constant time table lookup.*

We ran experiments using randomly generated CP-nets, with each line in Fig. 2 averaging 1000 CP-nets, generated using software by Allen et al.. LEX-EVAL performance is measured by counting the number of correctly identified incomparabilities in a CP-net order. Fig. 2 denotes the type of CP-nets generated using two integers. The entry $(1, 6)$ indicates a CP-net with six binary attributes and each attribute depends on at most one other attribute. As an additional method of comparison we also performed our analysis with a single consistent CLPM which was built by minimizing the rank of each node in the dependency graph (a greedy heuristic to produce a single CLPM in $CLPM(C)$ with a promise of identifying possibly many of the incomparabilities).

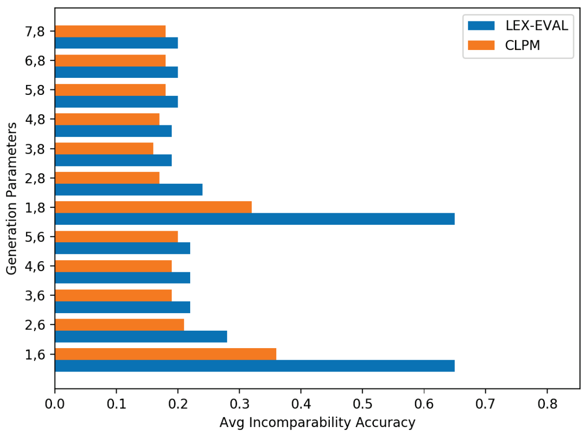

Figure 2: The average proportion of correctly decided incomparabilities using LEX-EVAL

Fig. 2 shows both LEX-EVAL and the single CLPM performs best with a CP-net in-degree bound of 1. In this case the aggregated order is capable of identifying about 65% of all incomparable alternative pairs, and a single "greedy" CLPM identifies about 32% of them (an improvement of about 85% more correct incomparabilities). The improvement falls to about 30% for CP-nets with in-degree 2, and falls further as the in-degree grows, but never below 10%.

It follows that the proposed approximation resolves some of the incomparabilities in the CP-net order, that is, orders them in some way. Still, a good number of CP-net incomparabilities are correctly identified. How to improve this approximation, while preserving good computational properties, is an interesting open challenge.

A related theoretical question is to study when the aggregated lexicographic evaluation will perfectly reproduce the dominance relation of a CP-net. Such a class of CP-nets exists. This class relies on the fact that some CP-nets can be separated into smaller independent CP-nets, see Def. 5.

Definition 5. *Given an acyclic CP-net $C = (\mathcal{V}, E, T)$, the set $V \subset \mathcal{V}$ is an independent subnetwork of C if for all $v \in V$ there does not exist $w \in (\mathcal{V} \backslash V)$ such that $(v, w) \in E$ or $(w, v) \in E$*

The class of CP-nets which are perfectly reproducible by the aggregated lexicographic evaluation are those that can be split into a series of independent subnetworks, where each subnetwork defines a linear order, see Thm. 7.

Theorem 7. *Given an acyclic CP-net C composed of independent subnetworks $\{c_1, \ldots, c_k\}$ if each $\succ_{c_i}$ produces a linear order then $>_C$ perfectly reproduces $\succ_C$.*

Thm. 7 is the direct result of the following results. Subnetworks can be viewed as CP-nets in their own right, with alternatives projected onto that subnetwork's attributes.

Definition 6. *Given an acyclic CP-net $C = (\mathcal{V}, E, T)$, an independent subnetwork V, and two alternatives α and β, $\alpha \succ_V \beta$ if there exists a flipping sequence from $\alpha[V]$ to $\beta[V]$ which does not change any values in $\mathcal{V} \setminus V$.*

Each subnetwork defines its own dominance relation, see Def. 6. These subnetwork dominance relations can be used to better understand how CLPMs and CP-nets interact.

Lemma 2. *Given an acyclic CP-net $C = (\mathcal{V}, E, T)$, an independent subnetwork V, and two alternatives α and β, if $\alpha \succ_V \beta$ then there exists a consistent CLPM, $\pi = (C, r)$ where $\alpha \succ_\pi \beta$.*

Lemma 3. *Given an acyclic CP-net $C = (\mathcal{V}, E, T)$, two independent subnetworks of C, V and W, and two alternatives α and β, $\alpha \bowtie_C \beta$ if $\alpha \succ_V \beta$ and $\beta \succ_W \alpha$.*

Corollary 4. *Given an acyclic CP-net $C = (\mathcal{V}, E, T)$, two independent subnetworks of C, V and W, and two alternatives α and β, if $\alpha \succ_V \beta$ and $\beta \succ_W \alpha$ then $\alpha \parallel_C \beta$.*

Lem. 2 shows that there exists a CLPM which approximates the dominance relation of any given independent subnetwork. Combined with Thm. 5 the aggregated lexicographic evaluation of a CP-net will reproduce incomparability when two independent subnetworks disagree. Cor. 4 shows these relations are identical in both the CP-net and its aggregated lexicographic evaluation.

Using CP-nets identified by Wilson (Wilson 2011) as independent subnetworks yields a rich class of CP-nets for which our approximation is exact. That class includes, in particular, all weakly separable CP-nets, this is, CP-nets that have no edges in their dependency graph.

Related Work

Using importance to simplify dominance testing on CP-nets is not new. Brafman, Domshlak, Carmel and Shimony introduced the concept of trade-off enhanced CP-nets, or TCP-nets (Brafman, Domshlak, and Shimony 2006). TCP-nets solve several problems of CP-nets by extending the CP-net representation with additional information. The difference between their approach and ours is that we extract importance information directly from the CP-net without relying on any additional elicited information.

Using the induced importance of attributes to help with dominance testing in CP-nets has been explored by Ahmed and Mouhoub. That work focused on the problem of finding optimal solutions based on a CP-net under a restricted domain (Ahmed and Mouhoub 2018). They define the term induced importance order for CP-nets, equivalent to rankings in this work. Their work dealt with finding an optimal alternative under domain constraints and not with finding a more general approximation of dominance. They also looked at eliciting additional importance information to fill out their representation. This work does not concern itself with additional information and uses only information provided by the CP-net itself.

Dubois, Prade, and Touzai investigated the use of possibilistic logics to evaluate CP-nets (Dubois, Prade, and Touazi 2013). They define how to encode a CP-net using possibilistic logic and also two ways of approximating dominance given the possibilistic representation of a CP-net. Their definition of leximin dominance produces the same order as a single CLPM. This is due to the similarity between our consistency constraint and the constraint on the probabilities associated with attributes. Under the leximin evaluation these probabilities convert nicely into an importance ranking. Evaluating a possibilistic representation under leximin replicates a single consistent CLPM. While such an approximation is important, Dubois, Prade, and Touazi do not further evaluate leximin beyond establishing it as an upper approximation of dominance. Our work extends beyond what Dubois, Prade, and Touazi did, in that we consider aggregated orders, establish an efficient evaluation algorithms, evaluate experimentally the quality of approximations (individual and aggregated), and identify some cases when the approximations are perfect.

Multiple attributes being similarly important, which we use in our work, is called attribute grouping. Grouping was described by Wilson (Wilson 2009) as the collecting of multiple single dimensional attributes into a single high dimensional attribute where preferences are expressed on the Cartesian product of the attribute domains. In the context of lexicographic preferences Bräuning, Hüllermeier, Keller, Glaum, and Martin (Bräuning et al. 2017) studied grouping for learning lexicographic preferences. Their concept of grouping is similar to how we define CLPMs, but our models do not require a linear order over the Cartesian product.

Wilson described the use of lexicographic inference of conditional preferences (Wilson 2014). This work dealt with a more general representation of conditional preferences than CP-nets. Wilson introduces lexicographic inference using Next Variable Predicates (NVPs) which define a series of singleton lexicographic inferences, that is partial lexicographic orders over a single attribute. He later extends NVPs to allow for grouping, just as our CLPMs allow for grouping. NVPs require a total order over all combinations of the grouped attribute values. CLPMs do not require this restrictive type of order and so our representation can be seen as related, but different. It is interesting if approach to approximating CP-nets can be extended to offer approximations to orders resulting from the framework proposed by Wilson.

Conclusion

This work shows how acyclic CP-nets can be approximated by CLPMs. Our main result is Thm. 2 which shows that any consistent CLPM will reproduce all strict dominance relations of a CP-net. This leads to Cor. 1 which shows that incomparability is implied in the opposite direction. Inaccuracies are limited to incomparable alternatives becoming comparable under the CLPM. Approximation accuracy can be increased by aggregating all consistent CLPMs. Dominance for the aggregated evaluation can be computed in polynomial time using our LEX-EVAL algorithm, specified in Algorithm 1. The data shows a benefit to LEX-EVAL over a singular CLPM. This leaves an open question about how this evaluation method compared to other CP-net approximations and heuristics.

Acknowledgment

This work was funded by the NSF under the grant number IIS-1618783.

References

Ahmed, S., and Mouhoub, M. 2018. Constrained optimization with preferentially ordered outcomes. In *2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI)*, 307–314. IEEE.

Allen, T. E.; Goldsmith, J.; Justice, H. E.; Mattei, N.; and Raines, K. 2016. Generating CP-nets uniformly at random. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Booth, R.; Chevaleyre, Y.; Lang, J.; Mengin, J.; and Sombattheera, C. 2010. Learning conditionally lexicographic preference relations. In *ECAI*, 269–274.

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

Brafman, R. I.; Domshlak, C.; and Shimony, S. E. 2006. On graphical modeling of preference and importance. *Journal of Artificial Intelligence Research* 25:389–424.

Bräuning, M.; Hüllermeier, E.; Keller, T.; and Glaum, M. 2017. Lexicographic preferences for predictive modeling of human decision making: A new machine learning method with an application in accounting. *European Journal of Operational Research* 258(1):295–306.

Dubois, D.; Prade, H.; and Touazi, F. 2013. Conditional preference nets and possibilistic logic. In *European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, 181–193. Springer.

Fishburn, P. C. 1974. Exceptional paper—lexicographic orders, utilities and decision rules: A survey. *Management science* 20(11):1442–1471.

Goldsmith, J.; Lang, J.; Truszczynski, M.; and Wilson, N. 2008. The computational complexity of dominance and consistency in CP-nets. *Journal of Artificial Intelligence Research* 33:403–432.

Huelsman, M., and Truszczyński, M. 2019. Approximating dominance in CP-nets by lexicographic evaluation. In *Proceedings of the 6th International Conference on Algorithmic Decision Theory*, 170–171. Springer.

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

Loreggia, A.; Mattei, N.; Rossi, F.; and Venable, K. B. 2018. On the distance between CP-nets. In *Proceedings of the 17th International Conference on Autonomous Agents and Multi-Agent Systems*, 955–963. International Foundation for Autonomous Agents and Multiagent Systems.

Wilson, N. 2009. Efficient inference for expressive comparative preference languages. In *Twenty-First International Joint Conference on Artificial Intelligence*.

Wilson, N. 2011. Computational techniques for a simple theory of conditional preferences. *Artificial Intelligence* 175(7-8):1053–1091.

Wilson, N. 2014. Preference inference based on lexicographic models. In *ECAI*, volume 14, 18–22.