Representation and Reasoning with Probabilistic TCP-nets

Sultan Ahmed[1] & Malek Mouhoub[1]

[1] Department of Computer Science, University of Regina, Regina, Saskatchewan, Canada

Correspondence: Malek Mouhoub, Department of Computer Science, University of Regina, Regina, Saskatchewan, Canada S4S 0A2. E-mail: mouhoubm@uregina.ca

Received: September 7, 2018 Accepted: September 26, 2018 Online Published: October 20, 2018

doi:10.5539/cis.v11n4p9 URL: https://doi.org/10.5539/cis.v11n4p9

Abstract

TCP-nets are graphical tools for modeling user's preference and relative importance statements. We propose the *Probabilistic TCP-net* (PTCP-net) model that can aggregate a set of TCP-nets, in a compact form, sharing the same set of variables and their domains but having different preference and relative importance statements. In particular, the PTCP-net is able to aggregate the choices of multiple users such as, in recommender systems. The PTCP-net can also be seen as an extension of the TCP-net with uncertainty on preference and relative importance statements. We adopt the Bayesian Network as the reasoning tool for PTCP-nets especially when answering the following two queries (1) finding the most probable TCP-net and (2) finding the most probable optimal outcome. We also show that the PTCP-net is applicable in collaborative filtering type recommender systems.

Keywords: preference reasoning, relative Importance, CP-nets, TCP-nets, Multi-Agent Systems, Uncertainty

1. Introduction

Reasoning with preferences using graphical models has attracted great interest in many real world applications, e.g., recommender systems (Jin et al., 2002) and product configuration (Brafman and Domshlak, 2002). A comprehensive review of graphical models for preference representation can be found in (Amor et al., 2016; Pigozzi et al., 2016). A *Conditional Preference Network* (CP-net) (Boutilier et al., 2004) is a graphical model that uses the notion of conditional preferential independence for representing and reasoning about qualitative preference statements in a compact and intuitive manner. A directed graph is used to encode preferential dependencies. For each variable in a CP-net, there is a Conditional Preference Table (CPT) listing the preference order over the variable domain, for each combination of the parent assignment set. A *Tradeoff-enhanced CP-net* (TCP-net) (Brafman et al., 2006) is an extension of the CP-net that accounts for user's qualitative and conditional relative importance statements by using the notion of conditional relative importance. For each conditional importance relation in a TCP-net, there is a Conditional Importance Table (CIT) listing the conditional relative importance relations between pairs of variables.

Aggregating users' preferences (Rossi et al., 2004; Yager, 2001) is often a necessary task for real world scenarios. For example, we consider a simple recommender system (Adomavicius and Tuzhilin, 2005; Breese et al., 1998) for a library. There are m books that are to be recommended to customers. Each book has two attributes, namely, Genre={Fiction, Nonfiction} and Media={Paper copy, Electronic copy}. There are n customers where each has preferences over the domain of each attribute, e.g., Fiction is preferred to Nonfiction. Each customer might also give more importance to Genre than to Media.

Let us assume that there are n_1 customers with the following preferences: (1) Fiction is preferred to Nonfiction (denoted as Fiction $\succ$ Nonfiction), (2) Paper copy is preferred to Electronic copy, and (3) Genre is more important than Media (denoted as Genre $\triangleright$ Media). The TCP-net representation of these statements is given in Figure 1 (i). Note that 'Genre is more important than Media' is depicted using a dashed directed arc. Now, consider a second group of n_2 customers having the same preferences as the first group but without any relative importance over the attributes. The TCP-net of this second group is depicted in Figure 1 (ii). Similarly, there is actually a total of 12 possible groups of customers with 12 different sets of unconditional preference and relative importance statements. 12 distinct TCP-nets are indeed needed to model all the customers' preference and relative importance statements. Our goal in this paper is to develop a model that can represent and explore a probability distribution over the set of possible TCP-nets.

To reach the above goal, we aggregate this set of TCP-nets with a single model extending TCP-nets with a probability for each preference or relative importance statement. We call this aggregated model the *Probabilistic TCP-net* (PTCP-net). The PTCP-net for the library recommender system is illustrated in Figure 1 (iii). Note that Genre $\bowtie$ Media denotes that 'Neither Genre is more important than Media nor Media is more important than Genre'. The probability of each statement is computed based on all customers. For example, if n_f out of n customers prefer Fiction to Nonfiction, then Pr(Fiction is preferred to Nonfiction)=n_f/n, which is denoted as p_1. Similarly, if to n_g customers, Genre is more important than

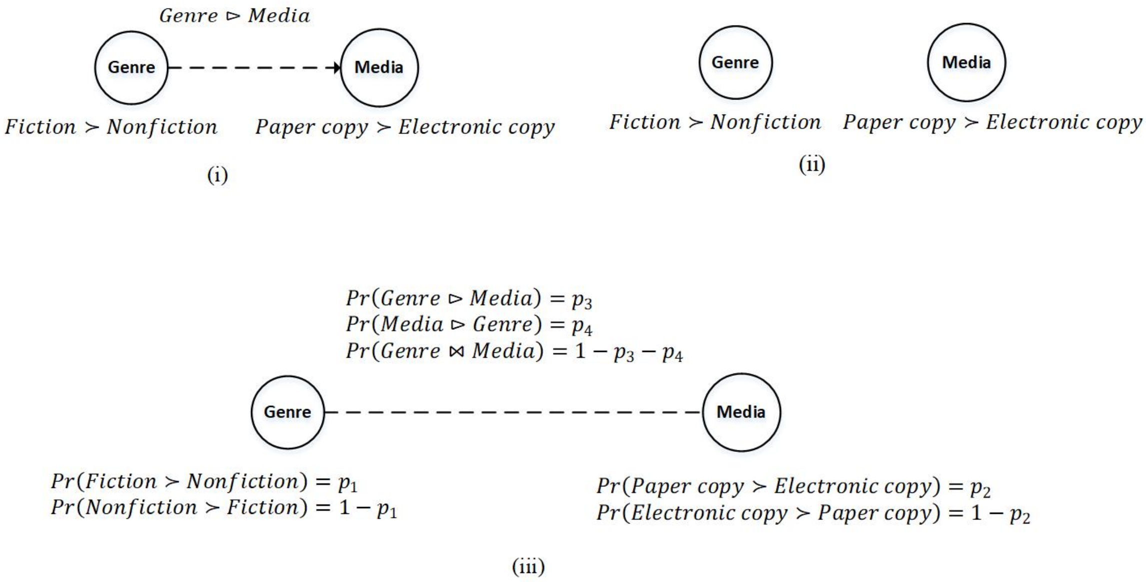

Genre ⊳ Media
Genre
Media
Fiction ≻ Nonfiction
Paper copy ≻ Electronic copy
(i)
Genre
Media
Fiction ≻ Nonfiction
Paper copy ≻ Electronic copy
(ii)
Pr(Genre ⊳ Media) = p3
Pr(Media ⊳ Genre) = p4
Pr(Genre ⋈ Media) = 1 − p3 − p4
Genre
Media
Pr(Fiction ≻ Nonfiction) = p1
Pr(Nonfiction ≻ Fiction) = 1 − p1
Pr(Paper copy ≻ Electronic copy) = p2
Pr(Electronic copy ≻ Paper copy) = 1 − p2
(iii)

Figure 1. Graphical representation of our library recommender system.

Media, then Pr(Genre is more important than Media)=n_g/n, which is denoted as p_3. We assume that the statements are mutually independent from one another. More generally, our PTCP-net extends the TCP-net with a probability distribution over preference orders, relative importance relations, existence of preferential dependencies and existence of relative importance relations. Instead of representing the choices of a single user, the PTCP-net is able to aggregate the choices of several users, each encoded in a possible TCP-net, in a compact manner. Moreover, the PTCP-net can also represent a single user's preference and relative importance statements with uncertainty. In this case, the probability comes from the elicitation process, e.g., a user provides 0.8 as the probability of 'Fiction is preferred to Nonfiction'. This is particularly important since, during the elicitation process in real world problems, preferences often come with noise (Liu et al., 2013). Note that all the possible TCP-nets encoded in a PTCP-net are not necessarily satisfiable due to the possible presence of cycles in the graph. We identify a particular sub class of PTCP-nets (that we call conditionally acyclic PTCP-nets) in which all TCP-nets are satisfiable. The focus of this paper is on this sub class.

After a PTCP-net is built, two queries need to be answered: (1) finding the most probable TCP-net and (2) finding the most probable optimal outcome. In a multi-agents case (such as recommender systems (Adomavicius and Tuzhilin, 2005; Burke et al., 2011; Konstan et al., 1997; Resnick et al., 1994)), the most probable TCP-net most likely represents the preference and relative importance statements of an unknown user. Thus, it facilitates the recommendation of items to the unknown user. On the other hand, the most probable optimal outcome most likely corresponds to an unknown item. Thus, this unknown item is recommended to each user such that the user's preference and relative importance statements are modeled using the TCP-net in which the optimal outcome is most probable. For finding the most probable TCP-net, we define the General-net (G-net) of a given PTCP-net. The G-net is literally a *Bayesian Network* (BN) (Pearl, 1988). To find the most probable optimal outcome, we define the Transformed-net (Trans-net) and the Optimal-net (Opt-net) graphical models. The Opt-net is also a BN. We show that the problem of finding the most probable TCP-net and the most probable optimal outcome is equivalent to finding the assignment with the maximal probability from the G-net and the Opt-net correspondingly. However, for practical use, we propose an algorithm, namely FindMPOO, that returns the most probable optimal outcome.

Finally, we present an application of our PTCP-net model in recommender systems (Adomavicius and Tuzhilin, 2005; Burke et al., 2011; Konstan et al., 1997; Resnick et al., 1994; Breese et al., 1998; Jin et al., 2002). Recommender systems consist of a set of items that are subject to be recommended to a set of users based on estimated rating. In a *Collaborative Filtering* (CF) recommender system, the rating of an item to a user is estimated based on the ratings on the same item from similar users. CF has a number of limitations (Adomavicius and Tuzhilin, 2005; Burke et al., 2011). Firstly, when there are new items or new users, recommendation cannot be made without relying on some additional knowledge sources (Adomavicius and Tuzhilin, 2005; Burke et al., 2011). Secondly, users with similar interests can have very different rating patterns, i.e., some users tend to assign a higher rating to all items than other users. This results in low accuracy. On the other hand, it is very natural in real world problems that, an item consists of many features (e.g., a book consists of its Genre, Media, Language, Volume, etc.), and users express conditional preferences over the features (e.g., if the book has Small Volume, the user prefers an Electronic copy to a Print copy) [1]. There is no approach in literature that utilizes users' conditional preferences on the item attributes.

We propose an intuitive model of recommender system that we call *Collaborative Filtering with Conditional Preferences* (CFCP). CFCP is a collaborative filtering type recommender system in which users' preference and relative importance statements are aggregated using a PTCP-net. We argue that CFCP will overcome two prominent issues of recommender systems: tackling new users and tackling new items. When there is a new user and the user's preference and relative important statements are unknown, the user is assumed to have the preference and relative importance statements that are encoded in the most probable TCP-net. Based on this, the CFCP estimates the rating of every item to the user, and recommends the items with highest rating. When there is a new unknown item in the system, this latter is recommended to every user such that the TCP-net that encodes the user's preference and relative importance statements has the most probable optimal outcome.

The PTCP-net can be considered as an extension of the Probabilistic CP-net (PCP-net) (Cornelio et al., 2013; Bigot et al., 2013; El Fidha et al., 2017) with relative importance relations. In a PCP-net, for each variable, there is a Probabilistic CPT (PCPT) that gives a probability distribution over the set of all preference orders for each assignment of the parent set. Bigot et al. (Bigot et al., 2013) introduce the PCP-net, but in their definition, preferential dependencies are considered deterministic. Cornelio et al. (Cornelio et al., 2013) give a general definition and show that BN is useful for the reasoning tasks in PCP-net. We follow their methodology of using BN for reasoning, but in the case of PTCP-net. Cornelio et al. (Cornelio et al., 2015) evaluated the PCP-net model in a multi-agent context. However, relative importance statements were not considered.

Rossi et al. (2004) proposed an extension of CP-net, called mCP-net, to represent qualitative and conditional preference of multiple agents. mCP-net is composed of many partial CP-nets, one for every agent. In this case, one agent's preferences may depend on the preferences of other agents. However, if many agents have a shared preference order, it occurs once in the mCP-net. So, a mCP-net does not give the strength of a preference order. Our PTCP-net overcomes it by adding probability on every preference order. On the other hand, PTCP-net also deals with relative importance statements, while mCP-net does not.

The rest of the paper is organized as follow. In Section 2, we describe the necessary background. The PTCP-net model is defined in Section 3 and the reasoning tasks with PTCP-nets are described in Section 4. In Section 5, we report the application of the PTCP-net in recommender systems. We finally conclude and outline some future research in Section 6.

2. Background

We start by defining preference and relative importance. Then, we discuss CP-nets and TCP-nets. We also briefly discuss BNs given that they are important tools for reasoning with PTCP-nets.

2.1 Preference and Relative Importance

In this subsection, we give the definitions with respect to a single user, i.e., in a deterministic case. A *preference relation* is defined as a preference order (also known as strict partial order). A preference order is a binary relation over outcomes, which is anti-reflexive, anti-symmetric and transitive. Given two outcomes o and o', we write $o \succ o'$ to denote that o is strictly preferred to o'. The types of outcomes, we are concerned with, consist of possible assignments to some set of variables. Let us consider the set $V = \{X_1, X_2, \cdots, X_n\}$ of variables with corresponding domains $D(X_1), D(X_2), \cdots, D(X_n)$. The set of possible outcomes is then $O = D(X_1) \times D(X_2) \times \cdots \times D(X_n)$. Since the number of possible outcomes is exponential in the number of variables, direct assessment of the preference order is usually not practical. In this regard, the notions of *preferential independence* and *conditional preferential independence* play a key role to represent the preference order compactly.

Definition 1. (Boutilier et al., 2004) Let $x_1, x_2 \in D(X)$ for some $X \subseteq V$, and $y_1, y_2 \in D(Y)$, where $Y = V - X$. X is *preferentially independent* of Y iff, for all x_1, x_2, y_1, y_2, we have that $x_1y_1 \succ x_2y_1$ iff $x_1y_2 \succ x_2y_2$.

Definition 2. (Boutilier et al., 2004) Let X, Y and Z be a partition of V and let $z \in D(Z)$. X is *conditionally preferentially independent* of Y given z iff, for all x_1, x_2, y_1, y_2, we have that $x_1y_1z \succ x_2y_1z$ iff $x_1y_2z \succ x_2y_2z$. X is conditionally preferentially independent of Y given Z, iff X is conditionally preferentially independent of Y given every assignment $z \in D(Z)$.

We now define the notion of *relative importance* of variables. The ordering of outcomes induced by this notion is relatively stronger than that of the preferential independence (Brafman et al., 2006).

[1] We do not mean that a user expresses conditional preferences on every item. Rather, as part of the profile, the user will be asked his or her preferences on the item attributes.

Definition 3. (Brafman et al., 2006) Let a pair of variables X and Y be mutually preferentially independent given $W = V - \{X, Y\}$. We say that X is more important than Y, denoted as $X \rhd Y$, iff for every assignment $w \in D(W)$ and for every $x_i, x_j \in D(X)$, $y_a, y_b \in D(Y)$, such that $x_i \succ x_j$ given w, we have that: $x_i y_a w \succ x_j y_b w$.

Definition 4. (Brafman et al., 2006) Let X and Y be a pair of variables from V, and let $Z \subseteq W - \{X, Y\}$. We say that X is more important than Y given $z \in D(Z)$ iff, for every assignment w' on $W' = V - (\{X, Y\} \cup Z)$ we have: $x_i y_a z w' \succ x_j y_b z w'$ whenever $x_i \succ x_j$ given zw'. We denote this relation by $X \rhd_z Y$. Finally, if for some $z \in D(Z)$ we have either $X \rhd_z Y$ or $Y \rhd_z X$, then we say that the relative importance of X and Y is conditioned on Z, and write $RI(X, Y|Z)$. Z is called *selector set* of (X, Y) and is denoted as $S(X, Y)$.

2.2 CP-nets and TCP-nets

A CP-net (Boutilier et al., 2004) is a graphical model for representing and reasoning about conditional preferences in a compact, intuitive and structural manner. This model allows users to express their preferences in a qualitative way, which is more natural and convenient for users compared to quantitative descriptions. This graphical model exploits the notions of preferential independence and conditional preferential independence in structuring a user's preferences under the *ceteris paribus* ("all else being equal") assumption. CP-nets are the first graphical model based on the notions of purely qualitative preferential independence captured by the *ceteris paribus* assumption, and bear a superficial similarity to BNs (Pearl, 1988). In a CP-net, preferential dependencies over the set V of variables are represented using directed arcs. An arc $\overrightarrow{(X_i, X_j)}$ for $X_i, X_j \in V$ indicates that the preference orders over $D(X_j)$ depend on the actual value of X_i. For each variable $X \in V$, there is a *Conditional Preference Table* (CPT). $CPT(X)$ gives the preference orders over $D(X)$ for each $u \in D(Pa(X))$, where $Pa(X)$ is the set of X's parents.

Definition 5. (Boutilier et al., 2004) A Conditional Preference Network (CP-net) over variables $V = \{X_1, X_2, \cdots, X_n\}$ is a directed graph over $X_1, X_2, \cdots, X_n$ whose nodes are annotated with $CPT(X_i)$ for each $X_i \in V$.

The semantics of a CP-net is defined in terms of the preference orders that are consistent with the preferences imposed by the CPTs. A preference order $\succ$ on the outcomes of a CP-net N satisfies the CPT of a variable X, iff $\succ$ orders every two outcomes that differ only on the value of X consistently with the preference order on $D(X)$ for each $u \in D(Pa(X))$. $\succ$ satisfies N iff $\succ$ satisfies each CPT of N. If o_1 and o_2 are two outcomes of N, we say that N entails $o_1 \succ o_2$, written as $N \models o_1 \succ o_2$, iff $o_1 \succ o_2$ holds in every preference order that satisfies N.

A TCP-net (Brafman et al., 2006) is an extension of the CP-net in order to compactly represent both preference and relative importance relations over a set V of variables. The TCP-net exploits the notion of conditional relative importance in structuring user's relative importance relations. In a TCP-net, the directed arcs that are used to represent preferential dependencies are called *cp-arcs*. In addition, an unconditional relative importance relation between two preferentially independent variables is represented using a directed arc, namely *i-arc*. An i-arc $\overrightarrow{(X_i, X_j)}$ in a TCP-net indicates that $X_i \rhd X_j$ holds. A conditional relative importance relation is represented using an undirected arc, namely *ci-arc*. A ci-arc (X_i, X_j) in a TCP-net indicates that $RI(X_i, X_j|S(X_i, X_j))$ holds. There is a *Conditional Importance Table* (CIT) corresponding to each ci-arc (X_i, X_j). $CIT(X_i, X_j)$ gives the relative importance relations over X_i and X_j for each $z \in D(S(X_i, X_j))$.

The semantics of a TCP-net is defined in terms of the set of preference orders consistent with the set of restrictions imposed by the preference and relative importance relations captured by the TCP-net. A TCP-net is said to be satisfied by a preference order $\succ$ if, (i) for each $X \in V$, $\succ$ satisfies $CPT(X)$, (ii) for each i-arc $\overrightarrow{(X_i, X_j)}$, $\succ$ satisfies the assertion that X is more important than Y if given every two complete assignments that differ on the value of X and Y only, $\succ$ prefers the assignment which provides X with a better value, and (iii) for each ci-arc (X_i, X_j) given selector set Z, $\succ$ satisfies the assertion that X is more important than Y given some assignment z to variable set Z if given every two complete assignments that differ on the value X and Y only, in which Z is assigned z, $\succ$ prefers the assignment which provides X with a better value. A TCP-net is satisfiable if there is at least a preference order over the outcomes that satisfies the TCP-net.

Every TCP-net is not necessarily satisfiable. A large class of TCP-nets are *conditionally acyclic*. Conditionally acyclic TCP-nets are guaranteed to be satisfiable (Brafman et al., 2006). To define conditionally acyclic TCP-nets, Brafman et al. (Brafman et al., 2006) defined dependency graphs and w-directed graphs. The dependency graph is an extension of the graph associated with a TCP-net in which for every ci-arc (X_i, X_j), two directed arcs $\overrightarrow{(Z, X_i)}$ and $\overrightarrow{(Z, X_j)}$ are added for every $Z \in S(X_i, X_j)$. The w-directed graph given an assignment w to all selector variables of a TCP-net, is defined using the dependency graph such that, every ci-arc (X_i, X_j) in the dependency graph is replaced with $\overrightarrow{(X_i, X_j)}$ if $X_i \rhd X_j$ holds given w, or with $\overrightarrow{(X_j, X_i)}$ if $X_j \rhd X_i$ holds given w. If all w-directed graphs with respect to a TCP-net are acyclic, the TCP-net is conditionally acyclic.

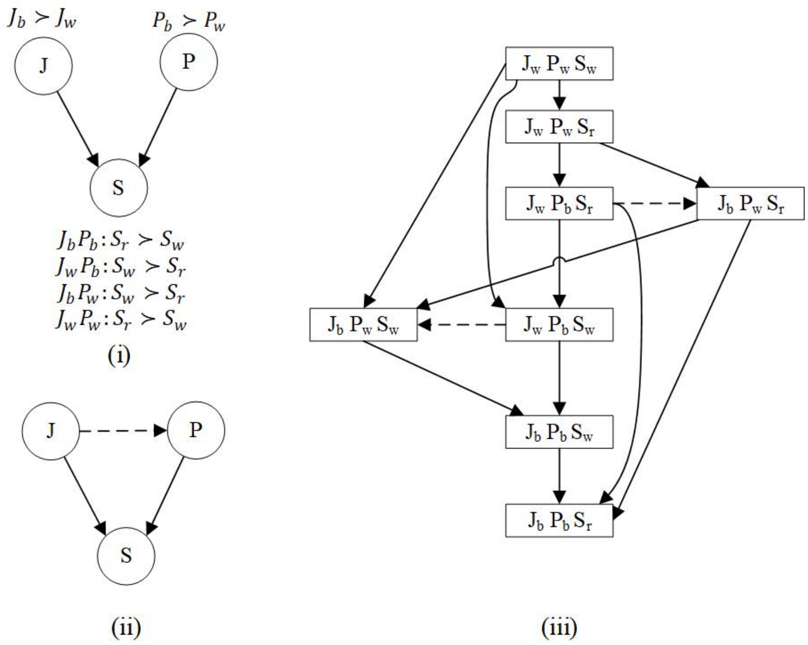


Figure 2. "Evening Dress" CP-net and TCP-net.

Example 1. *(Evening Dress (Boutilier et al., 2004; Brafman et al., 2006)). Figure 2 (i) presents a CP-net that consists of three variables J, P, and S, standing for the jacket, pants, and shirt, respectively. A user prefers black to white as a color for both the jacket and the pants, while the preference for the shirt color (red or white) is conditioned on the color combination of jacket and pants: If they are of the same color, a white shirt will make the dress too colorless, therefore, red shirt is preferable. Otherwise, if the jacket and the pants are of different colors, a red shirt will probably make the evening dress too flashy, therefore, a white shirt is preferable. The solid lines in Figure 2 (iii) show the preference relation induced directly by the information captured by this CP-net. The top and the bottom elements are the worst and the best outcomes, respectively, and the arrows are directed from less preferred to more preferred outcomes.*

Figure 2 (ii) depicts a TCP-net that extends this CP-net by adding an i-arc from J to P, i.e., having black jacket is (unconditionally) more important than having black pants. This induces additional relations among outcomes, captured by the dashed lines in Figure 2 (iii). □

2.3 BNs

BNs (Pearl, 1988) are probabilistic graphical models that have been used to model large and complex problem domains, which are subject to reasoning under uncertainty. A BN utilizes the notion of *probabilistic conditional independence* and the *chain rule* to compactly represent a *joint probability distribution*. A BN on $V = \{X_1, X_2, \cdots, X_n\}$ is a pair (B, C), where B is a *directed acyclic graph* (DAG) with node set V and C is a set of *conditional probability tables* (PTs) $\{Pr(X_i|Pa(X_i))|i = 1, 2, \cdots, n\}$. Each vertex in the DAG of a BN represents a random variable. A random variable denotes an attribute, feature, or hypothesis about which we may be uncertain. Each random variable has a set of mutually exclusive and collectively exhaustive possible values. That is, exactly one of the possible values is or will be the actual value, and we are uncertain about which one it is.

There are three standard inference tasks in BNs. (1) *Belief updating* is to find the probability table of a variable or a set of variables given evidence. (2) *Most probable explanation* (MPE) is to find the most probable assignment for all the variables given evidence. (3) *Maximum a-posteriori hypothesis* (MAP) is to find the most probable assignment for a subset of variables given evidence.

3. Probabilistic TCP-net Model

In a context of multiple users, the preferences on the values of a variable vary from user to user. For example, for the Evening Dress in Example 1, a black jacket is preferable to a white jacket for a user, while for another user, a white jacket is preferable to a black jacket. We use probabilities to aggregate the varying preferences of all users on the same set of variables. On the other hand, preferential dependencies also vary. For a user, the preferences on shirt colors depend on the pants colors, as in Example 1. For another user, the preferences on the pants colors depend on the shirt colors, e.g., if the shirt color is white, the user prefers a black pants to a white pants. Even more, for another user, the preferences on shirt colors depend on the pants colors and the preferences on pants colors depend on the shirt colors, i.e., a cycle exists [2]. In general, we consider that the preferential dependencies between two variables may exist in one way, in both ways, or may not exist at all. To represent the probable existence of a dependency, we add a probability on the cp-arcs of the corresponding TCP-net. The *Probabilistic CPT* (PCPT) for each variable is defined, accordingly, based on the existence of the cp-arcs.

The relative importance relation over two variables X and Y can vary from user to user, again, in a context of multiple users. For example, for l out of n users in a recommender system, X is more important than Y. For the remaining users, Y is more important than X. Then, the probability of $X \rhd Y$, denoted as $Pr(X \rhd Y)$, is l/n. To aggregate the varying importance relations of all users, we define the *Probabilistic CIT* (PCIT). A PCIT of unconditional relative importance relations (i.e., selector set is *null*) over two variables X and Y gives the probability distribution over the relative importance relations between X and Y. A PCIT of conditional relative importance relations $RI(X, Y|Z)$ gives a probability distribution over the relative importance relations between X and Y for every $z \in D(Z)$. If there is no importance relation between two variables X and Y, we denote it as $X \bowtie Y$.

Definition 6. (PTCP-net) A *Probabilistic TCP-net* (PTCP-net) $\mathcal{N}$ is a tuple $\langle V, pcp, pcpet, pcpt, pci, pciet, pcit\rangle$, where

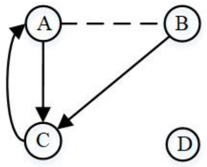
1. V is a set of nodes, corresponding to the problem variables.
2. pcp is a set of directed pcp-arcs $\alpha_{\mathcal{N}}$ (where pcp stands for *probabilistic conditional preference*). A pcp-arc $\overrightarrow{(X_i, X_j)}$ belongs to $\mathcal{N}$ iff the preferences over the values of X_j probably depend on the actual value of X_i.
3. $pcpet$ associates a PCPET (*pcp-arc existence table*) with every pcp-arc $\alpha \in \alpha_{\mathcal{N}}$. $PCPET(\alpha)$ represents the probability of existence for α, denoted as $Pr(\alpha)$.
4. $pcpt$ associates a PCPT (*probabilistic CPT*) with every node $X \in V$. The $PCPT(X)$ gives a probability distribution over the set of all preference orders on $D(X)$ for each $u \in D(Y : Y \subseteq Pa(X))$ (including a *null* value to account for the possible non-existence of the pcp-arcs).
5. pci is a set of pci-arcs (where pci stands for *probabilistic conditional importance*). A pci-arc (X_i, X_j) belongs to $\mathcal{N}$ iff we probably have $RI(X_i, X_j|Z)$ for some $Z \subseteq V - \{X_i, X_j\}$. Z is called the selector set of (X_i, X_j) and is denoted by $S(X_i, X_j)$. A pci-arc is unconditional if Z is *null*, otherwise it is conditional. We use $\beta_{\mathcal{N}}$ to denote the set of unconditional pci-arcs and $\gamma_{\mathcal{N}}$ for conditional pci-arcs.
6. $pciet$ associates a PCIET (*pci-arc existence table*) with every pci-arc (X_i, X_j). $PCIET(X_i, X_j)$ represents the probability of existence for (X_i, X_j), denoted as $Pr(X_i, X_j)$.
7. $pcit$ associates a PCIT (*probabilistic CIT*) with every pci-arc (X_i, X_j). $PCIT(X_i, X_j)$ gives a probability distribution over the relative importance relations on X_i and X_j for each $z \in D(S(X_i, X_j))$. The probability of $X_i \rhd X_j$ given $z \in D(S(X_i, X_j))$ is denoted as $Pr(X_i \rhd X_j|z)$.

The PTCP-net is a generalized formulation of the PCP-net. When *pci*, *pciet* and *pcit* are empty, the PTCP-net corresponds to a PCP-net. Note that, Brafman et al. (Brafman et al., 2006) defined a directed i-arc to denote an unconditional relative importance relation between two variables. An i-arc $\overrightarrow{(X_i, X_j)}$ belongs to a TCP-net iff $X_i \rhd X_j$. In a deterministic TCP-net, a directed arc is suitable to denote an unconditional relative importance relation, because if $X_i \rhd X_j$ holds, then $X_j \rhd X_i$ does not hold. However, in a PTCP-net, we use undirected pci-arc to denote the probable existence of unconditional relative importance relations. In a PTCP-net, an undirected arc is suitable to indicate a probability distribution on unconditional relative importance relations, because for some users $X_i \rhd X_j$ holds, and for some other users $X_j \rhd X_i$ holds. For example, a directed arc is appropriate to represent $A \rhd B$ in case of a TCP-net, whereas an undirected arc is appropriate to represent $A \rhd B$ with probability 0.6 and $B \rhd A$ with probability 0.4 for a PTCP-net.

Example 2. *A PTCP-net with four binary variables A, B, C and D, and their corresponding domains* $\{a_1, a_2\}$, $\{b_1, b_2\}$, $\{c_1, c_2\}$ *and* $\{d_1, d_2\}$ *is presented in Figure 3. Three pcp-arcs* $\overrightarrow{(A, C)}$, $\overrightarrow{(C, A)}$ *and* $\overrightarrow{(B, C)}$ *are denoted as solid directed arcs. Their probabilities of existence are listed in Figure 3 (ii). The arc* $\overrightarrow{(A, C)}$ *indicates that the preferences on the values of C probably depend on the actual value of A and this probability is* 0.4.

The PCPTs corresponding to the each variable are shown in Figure 3 (iii). The variable B has no parents. So, $PCPT(B)$ *gives the probability distribution over* $b_1 \succ b_2$ *and* $b_2 \succ b_1$*. It is similar for the variable D. A has a parent that is C.*

[2] No enough consideration was made regarding the existence of such cycles in Probabilistic TCP-nets (Ahmed and Mouhoub, 2017). In this paper, we emphasize to identify a sub class of Probabilistic TCP-nets in which such cycles do not exist.



(i)

PCPET	
pcp-arc	Pr
$\overrightarrow{(A,C)}$	0.4
$\overrightarrow{(B,C)}$	0.7
$\overrightarrow{(C,A)}$	0.1

(ii)

PCIET	
pci-arc	Pr
(A,B)	0.6

(iv)

PCIT				
Variables	S	*pci-arc*	Importance	Pr
A,B	D	(A,B)	$A \rhd B \mid d_1$	0.7
			$B \rhd A \mid d_1$	0.3
			$A \bowtie B \mid d_1$	0
			$A \rhd B \mid d_2$	0.4
			$B \rhd A \mid d_2$	0.6
			$A \bowtie B \mid d_2$	0
		Null	$A \rhd B \mid d_1$	0
			$B \rhd A \mid d_1$	0
			$A \bowtie B \mid d_1$	1
			$A \rhd B \mid d_2$	0
			$B \rhd A \mid d_2$	0
			$A \bowtie B \mid d_2$	1

(v)

PCPT			
Variable	Pa	Preference order	Pr
A	C	$a_1 \succ a_2 \mid c_1$	0.4
		$a_2 \succ a_1 \mid c_1$	0.6
		$a_1 \succ a_2 \mid c_2$	0.1
		$a_2 \succ a_1 \mid c_2$	0.9
	Null	$a_1 \succ a_2$	0.8
		$a_2 \succ a_1$	0.2
B	Null	$b_1 \succ b_2$	0.7
		$b_2 \succ b_1$	0.3
C	A,B	$c_1 \succ c_2 \mid a_1 b_1$	0.9
		$c_2 \succ c_1 \mid a_1 b_1$	0.1
		$c_1 \succ c_2 \mid a_1 b_2$	0.7
		$c_2 \succ c_1 \mid a_1 b_2$	0.3
		$c_1 \succ c_2 \mid a_2 b_1$	0.6
		$c_2 \succ c_1 \mid a_2 b_1$	0.4
		$c_1 \succ c_2 \mid a_2 b_2$	0.8
		$c_2 \succ c_1 \mid a_2 b_2$	0.2
	A	$c_1 \succ c_2 \mid a_1$	0.3
		$c_2 \succ c_1 \mid a_1$	0.7
		$c_1 \succ c_2 \mid a_2$	0.9
		$c_2 \succ c_1 \mid a_2$	0.1
	B	$c_1 \succ c_2 \mid b_1$	0.4
		$c_2 \succ c_1 \mid b_1$	0.6
		$c_1 \succ c_2 \mid b_2$	0.2
		$c_2 \succ c_1 \mid b_2$	0.8
	Null	$c_1 \succ c_2$	0.6
		$c_2 \succ c_1$	0.4
D	Null	$d_1 \succ d_2$	0.3
		$d_2 \succ d_1$	0.7

(iii)

Figure 3. A PTCP-net.

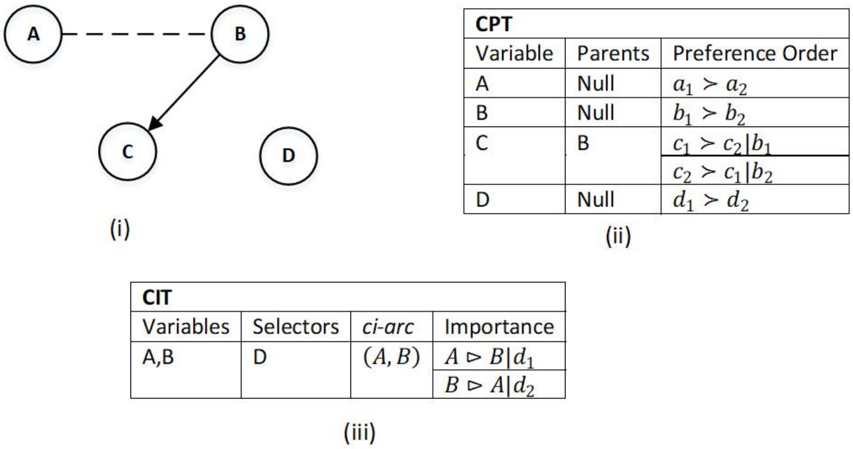
$PCPT(A)$ gives the probability distribution over $a_1 \succ a_2$ and $a_2 \succ a_1$ for c_1, c_2 and null. Similarly, C has two parents A and B. $PCPT(C)$ represents the probability distribution over $c_1 \succ c_2$ and $c_2 \succ c_1$ for a_1b_1, a_1b_2, a_2b_1, a_2b_2, a_1, a_2, b_1, b_2 and null.

A pci-arc is a dashed undirected arc, e.g., (A, B). $PCIT(A, B)$ indicates that the relative importance relations over A and B depend on the actual value of D. So, (A, B) is a conditional pci-arc and its selector is D. The $PCIET(A, B)$ and $PCIT(A, B)$ are given in Figure 3 (iv) and (v) correspondingly. □

Our intention is not to represent preference and relative importance relations using a PTCP-net. Rather, the PTCP-net represents a probability distribution over a set of TCP-nets. Each TCP-net in the set, is called a compatible TCP-net with respect to the PTCP-net.

Definition 7. (Compatible TCP-net) A TCP-net N is said to be compatible with respect to a PTCP-net $\mathcal{N}$, if the following conditions are satisfied.

1. N has the same variables and domains as $\mathcal{N}$.
2. The set of cp-arcs α_N in N is a subset of, or equal to, the set of pcp-arcs $\alpha_{\mathcal{N}}$ in $\mathcal{N}$.
3. For each $X \in V$, a $PCPT(X)$ in $\mathcal{N}$ has a corresponding $CPT(X)$ in N. Given the cp-arcs in N, a $CPT(X)$ is formed from $\mathcal{N}$ by selecting a preference order over $D(X)$ for each $u \in D(Pa(X))$ in N.
4. The set of i-arcs β_N in N is a subset of, or equal to, the set of unconditional pci-arcs $\beta_{\mathcal{N}}$ in $\mathcal{N}$. For an i-arc between X_i and X_j in N, a relative importance relation, either $X_i \rhd X_j$ or $X_j \rhd X_i$, is selected from $\mathcal{N}$. If $X_i \rhd X_j$ is selected, the i-arc in N is $\overrightarrow{(X_i, X_j)}$. If $X_j \rhd X_i$ is selected, the i-arc in N is $\overrightarrow{(X_j, X_i)}$.



(i)

CPT		
Variable	Parents	Preference Order
A	Null	$a_1 \succ a_2$
B	Null	$b_1 \succ b_2$
C	B	$c_1 \succ c_2 \mid b_1$ $c_2 \succ c_1 \mid b_2$
D	Null	$d_1 \succ d_2$

(ii)

CIT			
Variables	Selectors	*ci-arc*	Importance
A,B	D	(A, B)	$A \rhd B \mid d_1$ $B \rhd A \mid d_2$

(iii)

Figure 4. A TCP-net that is compatible with the PTCP-net of Figure 3.

5. The set of ci-arcs γ_N in N is a subset of, or equal to, the set of conditional pci-arcs $\gamma_{\mathcal{N}}$ in $\mathcal{N}$. For each ci-arc (X_i, X_j) in N, there is a $CIT(X_i, X_j)$ in N that is derived from the corresponding $PCIT(X_i, X_j)$ of $\mathcal{N}$. A $CIT(X_i, X_j)$ is formed by selecting a relative importance relation, either $X_i \rhd X_j$ or $X_j \rhd X_i$, over X_i and X_j for each $z \in D(S(X, Y))$.

Example 3. *A compatible TCP-net of the PTCP-net in Figure 3 is shown in Figure 4. The TCP-net has the same variables and domains as the PTCP-net. The pcp-arc $\overrightarrow{(B, C)}$ exists in the TCP-net as a cp-arc, while $\overrightarrow{(A, C)}$ and $\overrightarrow{(C, A)}$ do not exist (Figure 4 (i)). The CPTs of the variables are in Figure 4 (ii). For the variables A, B and D, the preference orders $a_1 \succ a_2$, $b_1 \succ b_2$ and $d_1 \succ d_2$ are selected correspondingly. The preference order of the dependent variable C depends on the actual value of B. The preference orders $c_1 \succ c_2$ and $c_2 \succ c_1$ are selected for $B = b_1$ and $B = b_2$ correspondingly. The conditional pci-arc (A, B) in the PTCP-net becomes a ci-arc in the TCP-net. The selector is D. When $D = d_1$, $A \rhd B$ is chosen. When $D = d_2$, $B \rhd A$ is chosen. The $CIT(A, B)$ is in Figure 4 (iii).* □

Let $\succ^u_X$ denotes a preference order over $D(X)$ for $X \in V$ and $u \in D(Pa(X))$ in N. $\rhd^u_{(X,Y)}$ denotes a relative importance relation over X and Y for $(X, Y) \in \beta_N$ and $u \in D(S(X, Y))$ in N.

Definition 8. (Probability of compatible TCP-net) If a TCP-net N is compatible with a PTCP-net $\mathcal{N}$, the probability of N with respect to $\mathcal{N}$ is defined as: $Pr_{\mathcal{N}}(N) = \prod_{\alpha\in\alpha_{\mathcal{N}},\alpha\in\alpha_N} Pr(\alpha)\cdot\prod_{\alpha\in\alpha_{\mathcal{N}},\alpha\notin\alpha_N}(1 - Pr(\alpha))\cdot\prod_{\beta\in\beta_{\mathcal{N}},\beta\in\beta_N} Pr(\beta)\cdot\prod_{\beta\in\beta_{\mathcal{N}},\beta\notin\beta_N}(1 - Pr(\beta))\cdot\prod_{\gamma\in\gamma_{\mathcal{N}},\gamma\in\gamma_N} Pr(\gamma)\cdot\prod_{\gamma\in\gamma_{\mathcal{N}},\gamma\notin\gamma_N}(1 - Pr(\gamma))\cdot\prod_{X\in V,\succ^u_X inN} Pr(\succ^u_X)\cdot\prod_{(X,Y)\in\beta_N\gamma_N,\rhd^u_{(X,Y)}inN} Pr(\rhd^u_{(X,Y)})$.

Example 4. *The probability of the compatible TCP-net in Figure 4 is computed as:* $Pr(\overrightarrow{B, C})\cdot(1 - Pr(\overrightarrow{A, C}))\cdot(1 - Pr(\overrightarrow{C, A}))\cdot Pr(A, B)\cdot Pr(a_1 \succ a_2)\cdot Pr(b_1 \succ b_2)\cdot Pr(c_1 \succ c_2|b_1)\cdot Pr(c_2 \succ c_1|b_2)\cdot Pr(d_1 \succ d_2)\cdot Pr(A \rhd B|d_1)\cdot Pr(B \rhd A|d_2) = 0.7\cdot(1 - 0.4)\cdot(1 - 0.1)\cdot 0.6\cdot 0.8\cdot 0.7\cdot 0.4\cdot 0.8\cdot 0.3\cdot 0.7\cdot 0.6$. □

Theorem 1. *If $\mathcal{N}$ is a PTCP-net and $\mathbb{N}$ is the set of all compatible TCP-nets, $Pr_{\mathcal{N}}(N)$ gives a probability distribution over $N \in \mathbb{N}$.*

Proof. Let $\mathbb{N}_1 \subset \mathbb{N}$, where for each $N \in \mathbb{N}_1$, we have $\alpha_N = \alpha_{\mathcal{N}}, \beta_N = \beta_{\mathcal{N}}$ and $\gamma_N = \gamma_{\mathcal{N}}$ (i.e., the structure of each $N \in \mathbb{N}_1$ is the same). $\succ_{X_1}$ denotes all preference orders over $D(X_1)$ for $X_1 \in V$ in $\mathcal{N}$. $\rhd_{(X_1,Y_1)}$ denotes all relative importance relations over X_1 and Y_1 for $(X_1, Y_1) \in \beta_{\mathcal{N}}\gamma_{\mathcal{N}}$. We can write:

$$\begin{aligned}
& \sum_{N\in\mathbb{N}_1} Pr_{\mathcal{N}}(N) \\
= & \sum_{N\in\mathbb{N}_1}(\prod_{\alpha\in\alpha_{\mathcal{N}}} Pr(\alpha)\cdot\prod_{\beta\in\beta_{\mathcal{N}}} Pr(\beta)\cdot\prod_{\gamma\in\gamma_{\mathcal{N}}} Pr(\gamma)\cdot\prod_{X\in V,\succ^u_X inN} Pr(\succ^u_X)\cdot\prod_{(X,Y)\in\beta_N\gamma_N,\rhd^u_{(X,Y)}inN} Pr(\rhd^u_{(X,Y)})) \\
= & \prod_{\alpha\in\alpha_{\mathcal{N}}} Pr(\alpha)\cdot\prod_{\beta\in\beta_{\mathcal{N}}} Pr(\beta)\cdot\prod_{\gamma\in\gamma_{\mathcal{N}}} Pr(\gamma)\cdot\sum_{N\in\mathbb{N}_1}(\prod_{X\in V,\succ^u_X inN} Pr(\succ^u_X)\cdot\prod_{(X,Y)\in\beta_N\gamma_N,\rhd^u_{(X,Y)}inN} Pr(\rhd^u_{(X,Y)})) \\
= & \prod_{\alpha\in\alpha_{\mathcal{N}}} Pr(\alpha)\cdot\prod_{\beta\in\beta_{\mathcal{N}}} Pr(\beta)\cdot\prod_{\gamma\in\gamma_{\mathcal{N}}} Pr(\gamma)\cdot\sum_{u\in D(Pa(X_1))}(\sum_{\succ^u_{X_1}\in\succ_{X_1}} Pr(\succ^u_{X_1}) \\
& \cdot\sum_{N\in\mathbb{N}_1}(\prod_{X\in V,\succ^u_X inN,u\notin D(Pa(X_1))} Pr(\succ^u_X)\cdot\prod_{(X,Y)\in\beta_N\gamma_N,\rhd^u_{(X,Y)}inN} Pr(\rhd^u_{(X,Y)})))
\end{aligned}$$

Since, for each $u \in D(Pa(X_1))$, $\sum_{\succ^u_{X_1}\in\succ_{X_1}} Pr(\succ^u_{X_1}) = 1$, we get:

$$\prod_{\alpha\in\alpha_{\mathcal{N}}} Pr(\alpha)\cdot\prod_{\beta\in\beta_{\mathcal{N}}} Pr(\beta)\cdot\prod_{\gamma\in\gamma_{\mathcal{N}}} Pr(\gamma)\cdot\sum_{N\in\mathbb{N}_1}(\prod_{X\in V-X_1,\succ^u_X inN} Pr(\succ^u_X)\cdot\prod_{(X,Y)\in\beta_N\gamma_N,\rhd^u_{(X,Y)}inN} Pr(\rhd^u_{(X,Y)}))$$

Similarly, $Pr(\succ^u_X)$ can be eliminated from the term for each $X \in V$. We get:

$$\begin{aligned}
& \prod_{\alpha\in\alpha_{\mathcal{N}}} Pr(\alpha)\cdot\prod_{\beta\in\beta_{\mathcal{N}}} Pr(\beta)\cdot\prod_{\gamma\in\gamma_{\mathcal{N}}} Pr(\gamma)\cdot\sum_{N\in\mathbb{N}_1}(\prod_{(X,Y)\in\beta_N\gamma_N,\rhd^u_{(X,Y)}inN} Pr(\rhd^u_{(X,Y)})) \\
= & \prod_{\alpha\in\alpha_{\mathcal{N}}} Pr(\alpha)\cdot\prod_{\beta\in\beta_{\mathcal{N}}} Pr(\beta)\cdot\prod_{\gamma\in\gamma_{\mathcal{N}}} Pr(\gamma)\cdot\sum_{u\in D(S(X_1,Y_1))}(\sum_{\rhd^u_{(X_1,Y_1)}\in\rhd_{(X_1,Y_1)}} Pr(\rhd^u_{(X_1,Y_1)}) \\
& \cdot\sum_{N\in\mathbb{N}_1}(\prod_{(X,Y)\in\beta_N\gamma_N,\rhd^u_{(X,Y)}inN,u\notin D(S(X_1,Y_1))} Pr(\rhd^u_{(X,Y)})))
\end{aligned}$$

Since, for each $u \in D(S(X_1, Y_1))$, $\sum_{\rhd^u_{(X_1,Y_1)}\in\rhd_{(X_1,Y_1)}} Pr(\rhd^u_{(X_1,Y_1)}) = 1$, we get:

$$\prod_{\alpha\in\alpha_{\mathcal{N}}} Pr(\alpha)\cdot\prod_{\beta\in\beta_{\mathcal{N}}} Pr(\beta)\cdot\prod_{\gamma\in\gamma_{\mathcal{N}}} Pr(\gamma)\cdot\sum_{N\in\mathbb{N}_1}(\prod_{(X,Y)\in\beta_N\gamma_N-(X_1,Y_1),\rhd^u_{(X,Y)}inN} Pr(\rhd^u_{(X,Y)}))$$

Similarly, $Pr(\rhd^u_{(X,Y)})$ can be eliminated from the term for each $(X, Y) \in \beta_N\gamma_N$. The final term will be: $\prod_{\alpha\in\alpha_{\mathcal{N}}} Pr(\alpha) \cdot \prod_{\beta\in\beta_{\mathcal{N}}} Pr(\beta) \cdot \prod_{\gamma\in\gamma_{\mathcal{N}}} Pr(\gamma)$. Let α_1 be a pcp-arc in $\mathcal{N}$. We assume that $\mathbb{N}_2 \subset \mathbb{N}$, where for each $N \in \mathbb{N}_2$, we have $\alpha_N = \alpha_{\mathcal{N}} - \alpha_1, \beta_N = \beta_{\mathcal{N}}$ and $\gamma_N = \gamma_{\mathcal{N}}$. It can be shown that: $\sum_{N\in\mathbb{N}_2} Pr_{\mathcal{N}}(N) = (1 - Pr(\alpha_1)) \cdot \prod_{\alpha\in\alpha_{\mathcal{N}}-\alpha_1} Pr(\alpha) \cdot \prod_{\beta\in\beta_{\mathcal{N}}} Pr(\beta) \cdot \prod_{\gamma\in\gamma_{\mathcal{N}}} Pr(\gamma)$. We get: $\sum_{N\in\mathbb{N}_1} Pr_{\mathcal{N}}(N) + \sum_{N\in\mathbb{N}_2} Pr_{\mathcal{N}}(N) = \prod_{\alpha\in\alpha_{\mathcal{N}}-\alpha_1} Pr(\alpha) \cdot \prod_{\beta\in\beta_{\mathcal{N}}} Pr(\beta) \cdot \prod_{\gamma\in\gamma_{\mathcal{N}}} Pr(\gamma)$. Similarly, by adding the terms for all possible combination of existence and non-existence of pcp-arcs and pci-arcs, we can show that: $\sum_{N\in\mathbb{N}} Pr_{\mathcal{N}}(N) = 1$. Thus, $Pr_{\mathcal{N}}(N)$ is a probability distribution. □

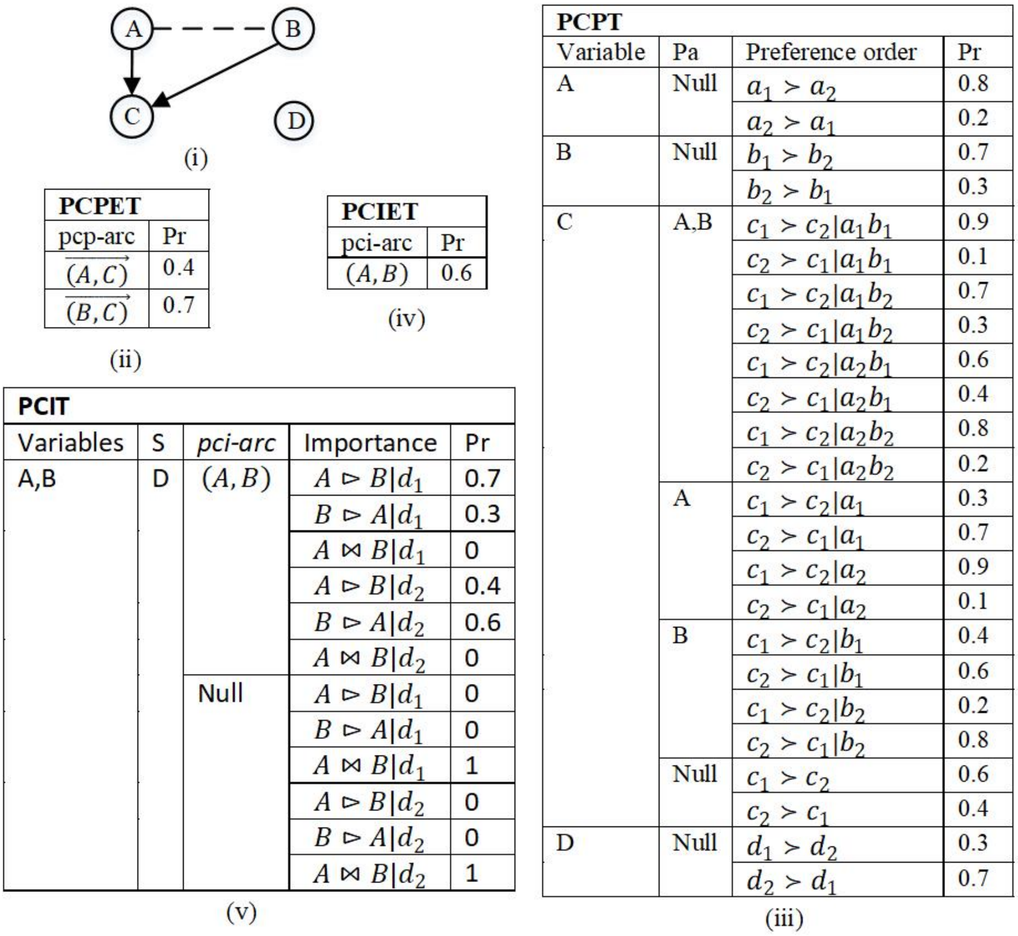
Note that a compatible TCP-net is not necessarily conditionally acyclic. For example, if we obtain a compatible TCP-net of the PTCP-net in Figure 3 by considering that both the pcp-arcs $\overrightarrow{(A, C)}$ and $\overrightarrow{(C, A)}$ exist in the TCP-net as cp-arcs, then it will give us a TCP-net with a cycle. Having a cycle in the TCP-net can make it non satisfiable. We intend to identify a class of PTCP-nets in which all compatible TCP-nets are conditionally acyclic, and thus satisfiable.

Definition 9. (Selector extended graph of a PTCP-net) The selector extended graph $\mathcal{N}^S$ of a PTCP-net $\mathcal{N}$ is formed as follows:

1. $\mathcal{N}^S$ contains all nodes, pcp-arcs and pci-arcs of $\mathcal{N}$.

2. For every conditional pci-arc (X_i, X_j) and every $X_k \in S(X_i, X_j)$ in $\mathcal{N}$, $\mathcal{N}^S$ contains a pair of directed arcs $\overrightarrow{(X_k, X_i)}$ and $\overrightarrow{(X_k, X_j)}$ if these arcs are not already in $\mathcal{N}$.

The selector extended graph of a PTCP-net is similar to the dependency graph of a TCP-net (Brafman et al., 2006).



(i)

PCPET	
pcp-arc	Pr
$\overrightarrow{(A,C)}$	0.4
$\overrightarrow{(B,C)}$	0.7

(ii)

PCPT			
Variable	Pa	Preference order	Pr
A	Null	$a_1 \succ a_2$	0.8
		$a_2 \succ a_1$	0.2
B	Null	$b_1 \succ b_2$	0.7
		$b_2 \succ b_1$	0.3
C	A,B	$c_1 \succ c_2 \vert a_1 b_1$	0.9
		$c_2 \succ c_1 \vert a_1 b_1$	0.1
		$c_1 \succ c_2 \vert a_1 b_2$	0.7
		$c_2 \succ c_1 \vert a_1 b_2$	0.3
		$c_1 \succ c_2 \vert a_2 b_1$	0.6
		$c_2 \succ c_1 \vert a_2 b_1$	0.4
		$c_1 \succ c_2 \vert a_2 b_2$	0.8
		$c_2 \succ c_1 \vert a_2 b_2$	0.2
	A	$c_1 \succ c_2 \vert a_1$	0.3
		$c_2 \succ c_1 \vert a_1$	0.7
		$c_1 \succ c_2 \vert a_2$	0.9
		$c_2 \succ c_1 \vert a_2$	0.1
	B	$c_1 \succ c_2 \vert b_1$	0.4
		$c_2 \succ c_1 \vert b_1$	0.6
		$c_1 \succ c_2 \vert b_2$	0.2
		$c_2 \succ c_1 \vert b_2$	0.8
	Null	$c_1 \succ c_2$	0.6
		$c_2 \succ c_1$	0.4
D	Null	$d_1 \succ d_2$	0.3
		$d_2 \succ d_1$	0.7

(iii)

PCIET	
pci-arc	Pr
(A,B)	0.6

(iv)

PCIT				
Variables	S	*pci-arc*	Importance	Pr
A,B	D	(A,B)	$A \rhd B \vert d_1$	0.7
			$B \rhd A \vert d_1$	0.3
			$A \bowtie B \vert d_1$	0
			$A \rhd B \vert d_2$	0.4
			$B \rhd A \vert d_2$	0.6
			$A \bowtie B \vert d_2$	0
		Null	$A \rhd B \vert d_1$	0
			$B \rhd A \vert d_1$	0
			$A \bowtie B \vert d_1$	1
			$A \rhd B \vert d_2$	0
			$B \rhd A \vert d_2$	0
			$A \bowtie B \vert d_2$	1

(v)

Figure 5. A modified PTCP-net of the PTCP-net in Figure 3.

Example 5. *Figure 5 shows a modified version of the PTCP-net in Figure 3. To find the selector extended graph (please see Figure 6 (i)), we consider the nodes A, B, C and D; the pcp-arcs $\overrightarrow{(A,C)}$ and $\overrightarrow{(B,C)}$; and the pci-arc (A,B). For the conditional pci-arc (A,B) and its selector D, we get two directed arcs $\overrightarrow{(D,A)}$ and $\overrightarrow{(D,B)}$. The selector extended graph is shown in Figure 6 (i).* □

Definition 10. (Extended graph of a PTCP-net) An extended graph $\mathcal{N}^*$ of a PTCP-net $\mathcal{N}$ can be obtained from its selector extended graph $\mathcal{N}^S$ as follows:

1. For every undirected pci-arc (X_i, X_j) in $\mathcal{N}^S$, (a) $\overrightarrow{(X_i, X_j)}$ is added in $\mathcal{N}^*$ if $\overrightarrow{(X_i, X_j)}$ is not already in $\mathcal{N}^S$, (b) $\overrightarrow{(X_j, X_i)}$ is added in $\mathcal{N}^*$ if $\overrightarrow{(X_j, X_i)}$ is not already in $\mathcal{N}^S$, or (c) neither $\overrightarrow{(X_i, X_j)}$ is added nor $\overrightarrow{(X_j, X_i)}$ is added in $\mathcal{N}^*$.

2. Every undirected pci-arc (X_i, X_j) in $\mathcal{N}^S$ is deleted.

Example 6. *To find an extended graph of the PTCP-net in Figure 5, we consider its selector extended graph in Figure 6 (i). For the pci-arc (A,B) in Figure 6 (i), we consider that $\overrightarrow{(A,B)}$ is added in the extended graph and also (A,B) is deleted. The extended graph is shown in Figure 6 (ii). Similarly, two other possible extended graphs are shown in Figure 6 (iii)-(iv).* □

Definition 11. (Conditionally acyclic PTCP-net) If all extended graphs of a PTCP-net $\mathcal{N}$ are acyclic, the PTCP-net is called conditionally acyclic.

Example 7. *The PTCP-net in Figure 5 is conditionally acyclic since its three extended graphs in Figure 6 (ii)-(iv) are acyclic.* □

Lemma 1. *Every conditionally acyclic PTCP-net $\langle V, pcp, pcpet, pcpt, pci, pciet, pcit\rangle$ contains at least one variable $X \in V$, such that for every $Y \in V - \{X\}$, there is no pcp-arc $\overrightarrow{(Y,X)}$ and there is no conditional pci-arc (X,Y).*

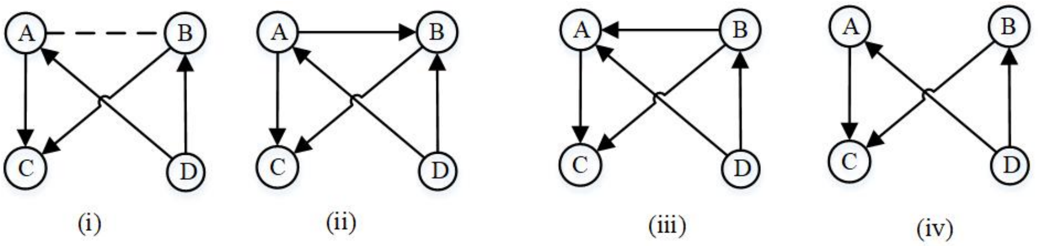


Figure 6. (i) The selector extended graph and (ii)-(iv) the extended graphs of the PTCP-net in Figure 5.

Proof. This proof is a consequence of the proof of Lemma 3 in (Brafman et al., 2006), given the selector extended graph of a PTCP-net corresponds to the dependency graph of a TCP-net. □

Theorem 2. *Every compatible TCP-net with respect to a conditionally acyclic PTCP-net is also conditionally acyclic.*

Proof. Let N be a compatible TCP-net of a conditionally acyclic PTCP-net $\mathcal{N}$. $\mathbb{N}^*$ be the set of the all extended graphs of $\mathcal{N}$. By definition, the cp-arcs in N are subset of, or equal to, the pcp-arcs in $\mathcal{N}$. It implies that all cp-arcs in N are also directed arcs in every $\mathcal{N}^* \in \mathbb{N}^*$.

The i-arcs in N are subset of, or equal to, the unconditional pci-arcs in $\mathcal{N}$. The ci-arcs in N are subset of, or equal to, the conditional pci-arcs in $\mathcal{N}$. From N, we can find its dependency graph (Brafman et al., 2006) that contains the all nodes and edges of N, and additionally, for every ci-arc (X_i, X_j) and every $X_k \in S(X_i, X_j)$ in N, the dependency graph contains a pair of directed arcs $\overrightarrow{(X_k, X_i)}$ and $\overrightarrow{(X_k, X_j)}$ if these arcs are not already in N. We get by definition of the extended graph that, there is at least one $\mathcal{N}^* \in \mathbb{N}^*$ such that the all directed arcs in the dependency graph are subset of, or equal to, the directed arcs in $\mathcal{N}^*$.

Now, we consider that $S(N)$ be the union of all selector sets of N. Given an assignment w to all nodes in $S(N)$, we can find the *w-directed* graph (Brafman et al., 2006) of the dependency graph that consists of all the nodes and directed edges of the dependency graph, and in addition it has a directed arc $\overrightarrow{(X_i, X_j)}$ if such an arc is not already in the dependency graph, and (X_i, X_j) is a ci-arc of N and the CIT for (X_i, X_j) specifies that $X_i \rhd X_j$ given w. We find that, for every *w-directed* graph, there is at least one $\mathcal{N}^* \in \mathbb{N}^*$ such that the all directed arcs in the *w-directed* graph are subset of, or equal to, the directed arcs in $\mathcal{N}^*$. Since every $\mathcal{N}^* \in \mathbb{N}^*$ is acyclic, every *w-directed* graph is also acyclic. By the definition of conditionally acyclic TCP-net (Brafman et al., 2006), N is conditionally acyclic. □

We now have that every compatible TCP-net of a conditionally acyclic PTCP-net is satisfiable. Brafman et al. (Brafman et al., 2006) mentioned that a large class of TCP-nets are conditionally acyclic. In this regard, we argue that, to aggregate multiple users' preferences, a conditional acyclic PTCP-net is likely to occur. In the rest of the paper, we consider only the PTCP-nets which are conditionally acyclic.

4. Reasoning with PTCP-nets

Given a PTCP-net, we need to answer two basic queries: (1) finding the most probable compatible TCP-net and (2) finding the most probable optimal outcome. These two queries have significance in tackling new users and new items in collaborative filtering type recommender systems, which are well-known issues of recommender systems (Adomavicius and Tuzhilin, 2005; Burke et al., 2011; Konstan et al., 1997; Resnick et al., 1994). The most probable TCP-net most likely represents the preference and relative importance statements of an unknown user. Thus, it facilitates the recommendation of items to an unknown user as we explain in Section 5. On the other hand, the most probable optimal outcome most likely corresponds to an unknown item. Thus, this unknown item is recommended to each user such that the user's preference and relative importance statements are modeled using the TCP-net in which the optimal outcome is most probable. In the following subsections, we will address each of these two queries.

4.1 The Most Probable Compatible TCP-net

Cornelio et al. (Cornelio et al., 2013) introduced the concept of *General Network* (G-net) in the case of a PCP-net. In this paper, we define the G-net for a PTCP-net. The G-net is a representational structure that helps us understand the probability distribution on all compatible TCP-nets of a given PTCP-net.

Definition 12. (G-net) A G-net $\mathcal{G}_{\mathcal{N}}$ of a conditionally acyclic PTCP-net $\mathcal{N}$ is a BN, where the variables and their *Conditional Probability Tables* (PTs) are defined as follows.

1. Each independent variable (i.e., having no incoming pcp-arc) $X \in V$ in $\mathcal{N}$ is also an independent variable X in $\mathcal{G_N}$. $D(X)$ in $\mathcal{G_N}$ is the set of all preference orders over $D(X)$ in $\mathcal{N}$. In $\mathcal{G_N}$, $PT(X)$ is given by the $PCPT(X)$ in $\mathcal{N}$.

2. Each pcp-arc $\overrightarrow{(X_i, X_j)}$ in $\mathcal{N}$ is an independent variable $\underline{X_iX_j}$ in $\mathcal{G_N}$ with domain $\{0, 1\}$ where $Pr(0) = 1 - Pr\overrightarrow{(X_i, X_j)}$ and $Pr(1) = Pr\overrightarrow{(X_i, X_j)}$. We call it a pcp-variable.

3. For each dependent variable X in $\mathcal{N}$, there are as many dependent variables in $\mathcal{G_N}$ as for each $u \in D(Y : Y \subseteq Pa(X))$, including *null*. If X_u is a dependent variable in $\mathcal{G_N}$ for $u \in D(Y : Y \subseteq Pa(X))$, $D(X_u)$ is given by the set of all preference orders over $D(X)$ in $\mathcal{N}$ and a dummy value d. X_u depends on the pcp-variables where the corresponding pcp-arcs in $\mathcal{N}$ are directed to X. Note that, for u, there is a corresponding set of existent pcp-arcs in $\mathcal{N}$. This set corresponds to an assignment of $Pa(X_u)$ in $\mathcal{G_N}$. For that assignment of $Pa(X_u)$, the $PT(X_u)$ is given by the $PCPT(X)$ for u from $\mathcal{N}$ with the probability of the dummy value d setting to 0. For the other assignments of $Pa(X_u)$, the $PT(X_u)$ is given by setting the probability of the dummy value d to 1.

4. Each pci-arc (X_i, X_j) in $\mathcal{N}$ is an independent variable $\underline{X_iX_j}$ in $\mathcal{G_N}$ with domain $\{0, 1\}$ where $Pr(0) = 1 - Pr(X_i, X_j)$ and $Pr(1) = Pr(X_i, X_j)$. We call it a pci-variable.

5. For each $PCIT(X_i, X_j)$ in $\mathcal{N}$, there are as many dependent variables in $\mathcal{G_N}$ as for each $z \in D(S(X_i, X_j))$, excluding *null*. If $I_{\underline{X_iX_j}/z}$ is a dependent variable in $\mathcal{G_N}$ for $z \in D(S(X_i, X_j))$, $D(I_{\underline{X_iX_j}/z})$ is given by the set of all relative importance relations over X_i and X_j in $\mathcal{N}$. $I_{\underline{X_iX_j}/z}$ depends on the pci-variable $\underline{X_iX_j}$. $PT(I_{\underline{X_iX_j}/z})$ is given by the $PCIT(X_i, X_j)$ for z from $\mathcal{N}$.

Example 8. *A G-net $\mathcal{G_N}$, corresponding to the PTCP-net $\mathcal{N}$ in Figure 5, is shown in Figure 7. The independent variables are A, B and D. Their PTs are given by the corresponding PCPTs. The pcp-variables are $\underline{AC}$ and $\underline{BC}$. Their PTs are given by the corresponding PCPETs. For the dependent variable C in $\mathcal{N}$, there are 9 dependent variables in $\mathcal{G_N}$. They are $C_{a_1b_1}$, $C_{a_1b_2}$, $C_{a_2b_1}$, $C_{a_2b_2}$, C_{a_1}, C_{a_2}, C_{b_1}, C_{b_2} and C_{null}. Each of these dependent variables has domain of $c_1 \succ c_2$, $c_2 \succ c_1$ and d. Each of these dependent variables depends on the pcp-variables $\underline{AC}$ and $\underline{BC}$. $PT(C_{a_1b_1})$ is determined from Figure 5 (iii). Given a_1b_1 for $PCPT(C)$, the existent pcp-arcs are $\overrightarrow{(A, C)}$ and $\overrightarrow{(B, C)}$. When $\underline{AC} = 1$ and $\underline{BC} = 1$, $PT(C_{a_1b_1})$ is given by $PCPT(C)$ for a_1b_1. For other combinations of $\underline{AC}$ and $\underline{BC}$, we have: $Pr(c_1 \succ c_2|\underline{AC}, \underline{BC}) = 0$, $Pr(c_2 \succ c_1|\underline{AC}, \underline{BC}) = 0$ and $Pr(d|\underline{AC}, \underline{BC}) = 1$. The PTs of the other variables are similarly obtained.*

The only one pci-arc (A, B) in Figure 5 becomes a pci-variable $\underline{AB}$. The $PT(\underline{AB})$ is given by the corresponding PCIET.

For the $PCIT(A, B)$ in Figure 5 (v), the dependent variables in Figure 7 are $I_{\underline{AB}/d_1}$ and $I_{\underline{AB}/d_2}$. Their domains are $A \rhd B$, $B \rhd A$ and $A \bowtie B$. Both of the variables depend on $\underline{AB}$. $PT(I_{\underline{AB}/d_1})$ is given by $PCIT(A, B)$ for d_1 from Figure 5 (v). When $\underline{AB} = 1$, we have: $Pr(A \rhd B|\underline{AB}) = 0.7$, $Pr(B \rhd A|\underline{AB}) = 0.3$ and $Pr(A \bowtie B|\underline{AB}) = 0$. When $\underline{AB} = 0$, we have: $Pr(A \rhd B|\underline{AB}) = 0$, $Pr(B \rhd A|\underline{AB}) = 0$ and $Pr(A \bowtie B|\underline{AB}) = 1$. $PT(I_{\underline{AB}/d_2})$ is similarly determined. □

Intuitively, each assignment in $\mathcal{G_N}$ corresponds to a compatible TCP-net. The probability of the compatible TCP-net can be given by the joint probability of the corresponding assignment in $\mathcal{G_N}$. Therefore, given a conditionally acyclic PTCP-net $\mathcal{N}$, the problem of finding the most probable compatible TCP-net is mapped into the problem of finding the assignment with the maximal joint probability in $\mathcal{G_N}$. Since the problem is to find the MPE in a BN, it is NP-hard (Kwisthout, 2011).

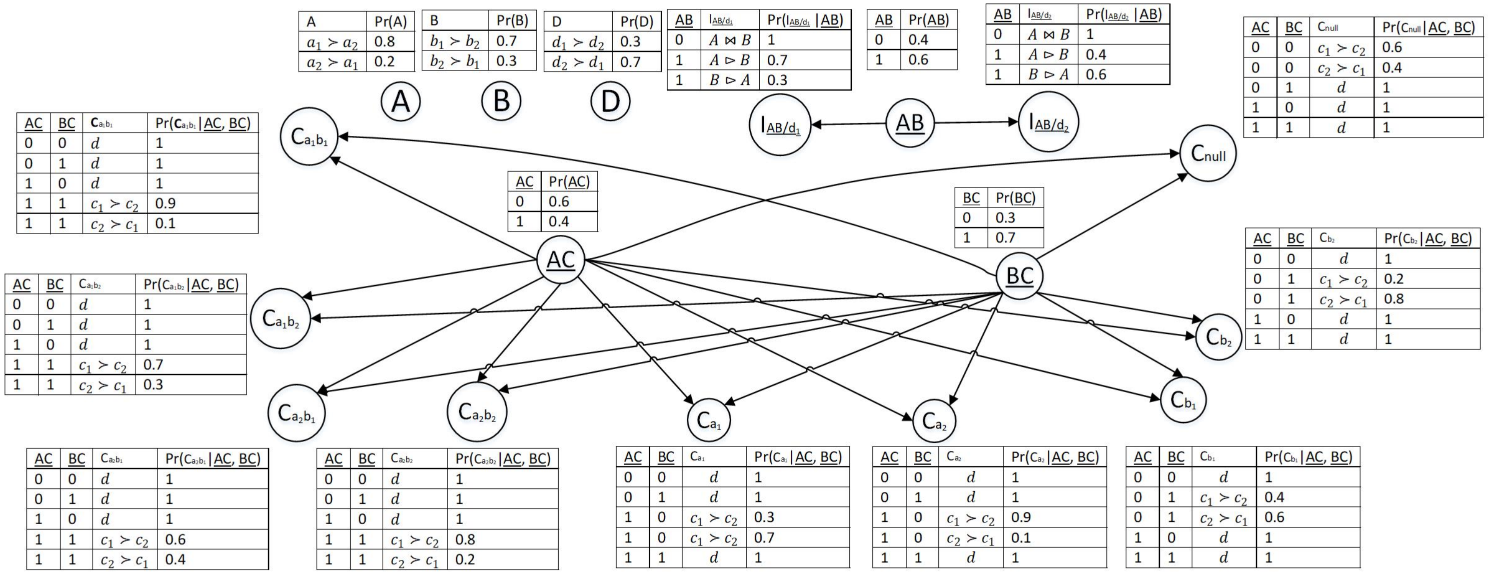
4.2 The most probable optimal outcome

Every conditionally acyclic TCP-net has one and only one optimal outcome (Brafman et al., 2006). While a conditionally acyclic PTCP-net represents a probability distribution over a set of conditionally acyclic TCP-nets, we use this distribution to compute the probability of an outcome to be optimal with respect to the PTCP-net.

Definition 13. (Probability of optimality) Given a conditionally acyclic PTCP-net $\mathcal{N}$ and the set of its compatible TCP-nets $\mathbb{N}$, the probability of an outcome o to be optimal with respect to $\mathcal{N}$, is defined as: $Pr(o) = \sum_{N \in \mathbb{N},\ \text{o is optimal in N}} Pr_{\mathcal{N}}(N)$.

In case of a PCP-net, the most probable optimal outcome is not necessarily the optimal outcome of the most probable compatible CP-net (Cornelio et al., 2013). This is indeed true for a PTCP-net too.

Example 9. *Consider a PTCP-net with a single variable A and its domain $\{a_1, a_2, a_3\}$. $PCPT(A)$ is given by $a_1 \succ a_2 \succ a_3$, $a_1 \succ a_3 \succ a_2$ and $a_2 \succ a_1 \succ a_3$ with their corresponding probabilities 0.3, 0.3 and 0.4. It is easy to see that the PTCP-net represents the probability distribution of 3 TCP-nets. The TCP-net with the order $a_2 \succ a_1 \succ a_3$ is most probable. In this TCP-net, the optimal outcome is a_2. Therefore, the probability of a_2 to be optimal with respect to the PTCP-net is 0.4. On the other hand, in the other 2 TCP-nets, the optimal outcome is a_1. Therefore, the probability of a_1 to be optimal with respect to the PTCP-net is 0.3 + 0.3 = 0.6.* □



A	Pr(A)
$a_1 \succ a_2$	0.8
$a_2 \succ a_1$	0.2

B	Pr(B)
$b_1 \succ b_2$	0.7
$b_2 \succ b_1$	0.3

D	Pr(D)
$d_1 \succ d_2$	0.3
$d_2 \succ d_1$	0.7

AB	$I_{\underline{AB}/d_1}$	Pr($I_{\underline{AB}/d_1}$ | AB)
0	$A \bowtie B$	1
1	$A \rhd B$	0.7
1	$B \rhd A$	0.3

AB	Pr(AB)
0	0.4
1	0.6

AB	$I_{\underline{AB}/d_2}$	Pr($I_{\underline{AB}/d_2}$ | AB)
0	$A \bowtie B$	1
1	$A \rhd B$	0.4
1	$B \rhd A$	0.6

AC	BC	C_{null}	Pr(C_{null} | AC, BC)
0	0	$c_1 \succ c_2$	0.6
0	0	$c_2 \succ c_1$	0.4
0	1	d	1
1	0	d	1
1	1	d	1

AC	BC	$C_{a_1b_1}$	Pr($C_{a_1b_1}$ | AC, BC)
0	0	d	1
0	1	d	1
1	0	d	1
1	1	$c_1 \succ c_2$	0.9
1	1	$c_2 \succ c_1$	0.1

AC	Pr(AC)
0	0.6
1	0.4

BC	Pr(BC)
0	0.3
1	0.7

AC	BC	C_{b_2}	Pr(C_{b_2} | AC, BC)
0	0	d	1
0	1	$c_1 \succ c_2$	0.2
0	1	$c_2 \succ c_1$	0.8
1	0	d	1
1	1	d	1

AC	BC	$C_{a_1b_2}$	Pr($C_{a_1b_2}$ | AC, BC)
0	0	d	1
0	1	d	1
1	0	d	1
1	1	$c_1 \succ c_2$	0.7
1	1	$c_2 \succ c_1$	0.3

AC	BC	$C_{a_2b_1}$	Pr($C_{a_2b_1}$ | AC, BC)
0	0	d	1
0	1	d	1
1	0	d	1
1	1	$c_1 \succ c_2$	0.6
1	1	$c_2 \succ c_1$	0.4

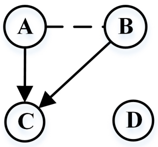
AC	BC	$C_{a_2b_2}$	Pr($C_{a_2b_2}$ | AC, BC)
0	0	d	1
0	1	d	1
1	0	d	1
1	1	$c_1 \succ c_2$	0.8
1	1	$c_2 \succ c_1$	0.2

AC	BC	C_{a_1}	Pr(C_{a_1} | AC, BC)
0	0	d	1
0	1	d	1
1	0	$c_1 \succ c_2$	0.3
1	0	$c_1 \succ c_2$	0.7
1	1	d	1

AC	BC	C_{a_2}	Pr(C_{a_2} | AC, BC)
0	0	d	1
0	1	d	1
1	0	$c_1 \succ c_2$	0.9
1	0	$c_2 \succ c_1$	0.1
1	1	d	1

AC	BC	C_{b_1}	Pr(C_{b_1} | AC, BC)
0	0	d	1
0	1	$c_1 \succ c_2$	0.4
0	1	$c_2 \succ c_1$	0.6
1	0	d	1
1	1	d	1

Figure 7: The G-net of the PTCP-net in Figure 5.



(i)

PCPET	
pcp-arc	Pr
$\overrightarrow{(A,C)}$	1
$\overrightarrow{(B,C)}$	1

(ii)

PCIET	
pci-arc	Pr
(A,B)	1

(iv)

PCT			
Variable	Pa	Preference order	Pr
A	Null	$a_1 \succ a_2$	0.8
		$a_2 \succ a_1$	0.2
B	Null	$b_1 \succ b_2$	0.7
		$b_2 \succ b_1$	0.3
C	A,B	$c_1 \succ c_2 \| a_1 b_1$	0.564
		$c_2 \succ c_1 \| a_1 b_1$	0.436
		$c_1 \succ c_2 \| a_1 b_2$	0.424
		$c_2 \succ c_1 \| a_1 b_2$	0.576
		$c_1 \succ c_2 \| a_2 b_1$	0.552
		$c_2 \succ c_1 \| a_2 b_1$	0.448
		$c_1 \succ c_2 \| a_2 b_2$	0.524
		$c_2 \succ c_1 \| a_2 b_2$	0.476
D	Null	$d_1 \succ d_2$	0.3
		$d_2 \succ d_1$	0.7

PCIT			
pci-arc	S	Importance	Pr
(A,B)	D	$A \rhd B \| d_1$	0.42
		$B \rhd A \| d_1$	0.18
		$A \bowtie B \| d_1$	0.40
		$A \rhd B \| d_2$	0.24
		$B \rhd A \| d_2$	0.36
		$A \bowtie B \| d_2$	0.40

(v)

(iii)

Figure 8. The Trans-net of the PTCP-net in Figure 5.

To find the most probable optimal outcome for a PCP-net, two networks, namely *Transformed PCP-net* (Trans-net) and *Optimal Network* (Opt-net), were defined (Cornelio et al., 2013). We extend these two networks in the case of the PTCP-net.

Definition 14. (Trans-net) The Trans-net $\mathcal{N}_T$ of a given PTCP-net $\mathcal{N}$ is also a PTCP-net with the following:

1. $\mathcal{N}_T$ has the same variables and their domains as $\mathcal{N}$.
2. All the pcp-arcs in $\mathcal{N}$ are also in $\mathcal{N}_T$ with probability of existence 1.
3. For an independent variable, the PCPTs in both $\mathcal{N}$ and $\mathcal{N}_T$ are the same. For a dependent variable, pcp-arc probabilities are transformed in computing PCPT. For the PCPT of a dependent variable X, the probability of a preference order $\succ^u_X$ in $\mathcal{N}_T$ is computed from $\mathcal{N}$ as: $\sum_{Y\subseteq Pa(X), v\in D(Y), v\subseteq u}(Pr(\succ^v_X)\cdot\prod_{Z\in Pa(X), D(Z)\cap v\neq\emptyset} Pr(\overrightarrow{Z,X})\cdot\prod_{Z\in Pa(X), D(Z)\cap v=\emptyset}(1-Pr(\overrightarrow{Z,X})))$.
4. All the pci-arcs in $\mathcal{N}$ are also in $\mathcal{N}_T$ with probability of existence 1.
5. The pci-arc probabilities are transformed in computing PCITs in $\mathcal{N}_T$. For the PCIT of a pci-arc (X_i, X_j), the probability of an importance relation $\rhd^z_{(X_i,X_j)}$ in $\mathcal{N}_T$ is computed from $\mathcal{N}$ as follow: $Pr(\rhd^z_{(X_i,X_j)})\cdot Pr(X_i,X_j)|(X_i,X_j)$ exists$+$ $Pr(\rhd^z_{(X_i,X_j)})\cdot(1-Pr(X_i,X_j))|(X_i,X_j)$ does not exist.

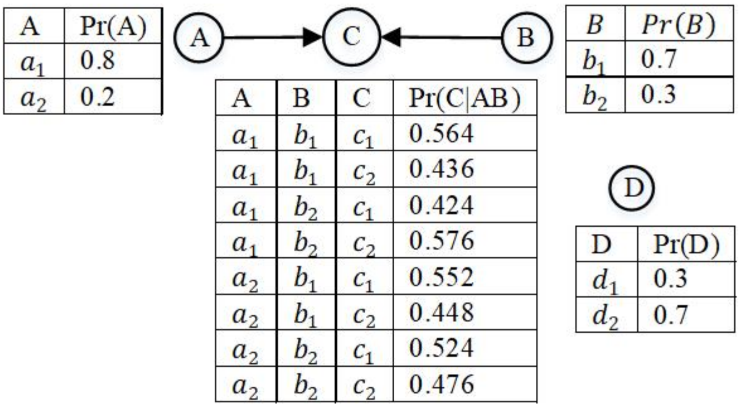
The Trans-net is basically a TCP-net with probabilities only on the preferences and no longer on the existence of the arcs.

Example 10. *The Trans-net $\mathcal{N}_T$, of the PTCP-net $\mathcal{N}$ in Figure 5, is shown in Figure 8. The probability of existence for both pcp-arcs $\overrightarrow{(A,C)}$ and $\overrightarrow{(B,C)}$ is* 1 *(see Figure 8 (ii)). The PCPTs for the independent variables A, B and D are same as in Figure 5. For the dependent variable C, $Pr(c_1 \succ c_2|a_1b_1)$ in Figure 8 (iii) is computed from Figure 5 (iii) as: $Pr(c_1 \succ c_2|a_1b_1)\cdot Pr(\overrightarrow{A,C})\cdot Pr(\overrightarrow{B,C})+Pr(c_1 \succ c_2|a_1)\cdot Pr(\overrightarrow{A,C})\cdot(1-Pr(\overrightarrow{B,C}))+Pr(c_1 \succ c_2|b_1)\cdot Pr(\overrightarrow{B,C})\cdot(1-Pr(\overrightarrow{A,C}))+$ $Pr(c_1 \succ c_2)\cdot(1-Pr(\overrightarrow{A,C}))\cdot(1-Pr(\overrightarrow{B,C}))$, which is* 0.564*. The probabilities of the other preference orders are computed similarly.*

The probability of existence for the pci-arc (A,B) is 1 *(see Figure 8 (iv)). For PCIT(A,B), $Pr(A \rhd B|d_1)$ in Figure 8 (v) is computed from Figure 5 (v) as: $Pr(A \rhd B|d_1)\cdot Pr(A,B)|(A,B)$ exists $+$ $Pr(A \rhd B|d_1)\cdot(1-Pr(A,B))|(A,B)$ does not exist, which is* 0.42*. The probabilities of the other relative importance relations are computed similarly.* □

Lemma 2. *Let $\mathcal{N}_T$ be a Trans-net of a PTCP-net $\mathcal{N}$. If $\mathcal{N}$ is conditionally acyclic, $\mathcal{N}_T$ is also conditionally acyclic.*

Proof. By definition, the nodes, pcp-arcs, pci-arcs and selectors are in $\mathcal{N}_T$, iff they are in $\mathcal{N}$. Therefore, the extended graphs of $\mathcal{N}_T$ are exactly the same as the extended graphs of $\mathcal{N}$. Since the all extended graphs of $\mathcal{N}$ are acyclic, the all extended graphs of $\mathcal{N}_T$ are also acyclic. So, $\mathcal{N}_T$ is conditionally acyclic. □



A	Pr(A)
a_1	0.8
a_2	0.2

B	$Pr(B)$
b_1	0.7
b_2	0.3

A	B	C	Pr(C|AB)
a_1	b_1	c_1	0.564
a_1	b_1	c_2	0.436
a_1	b_2	c_1	0.424
a_1	b_2	c_2	0.576
a_2	b_1	c_1	0.552
a_2	b_1	c_2	0.448
a_2	b_2	c_1	0.524
a_2	b_2	c_2	0.476

D	Pr(D)
d_1	0.3
d_2	0.7

Figure 9. The Opt-net corresponding to the Trans-net in Figure 8.

Lemma 3. *The probability of an outcome o to be optimal with respect to a conditionally acyclic PTCP-net $\mathcal{N}$ is the same as the probability of the outcome to be optimal with respect to the corresponding Trans-net $\mathcal{N}_T$.*

Proof. Let $\mathbb{N}$ and $\mathbb{N}_T$ be the set of $\mathcal{N}$'s and $\mathcal{N}_T$'s compatible TCP-nets correspondingly. Let o be optimal for every $N \in \mathbb{N}_1$, where $\mathbb{N}_1 \subseteq \mathbb{N}$. Let α be a pcp-arc. Let $\mathbb{N}_1 = \mathbb{N}_{1\alpha} \cup \mathbb{N}_{1\neg\alpha}$, where cp-arc α exists in every $N \in \mathbb{N}_{1\alpha}$ and cp-arc α does not exist for every $N \in \mathbb{N}_{1\neg\alpha}$. It means that each $N \in \mathbb{N}_{1\alpha}$ has a corresponding $N \in \mathbb{N}_{1\neg\alpha}$, where they differ only on the existence of α. Let N_α and $N_{\neg\alpha}$ be such correspondence, where α exists in N_α and does not exist in $N_{\neg\alpha}$.

On the other hand, N_α is also a compatible TCP-net of $\mathcal{N}_T$. There is no compatible TCP-net of $\mathcal{N}_T$ such that the cp-arc α does not exist, since α has probability 1 in $\mathcal{N}_T$. By Definition 14, we have: $Pr_{\mathcal{N}_T}(N_\alpha) = Pr_{\mathcal{N}}(N_\alpha) + Pr_{\mathcal{N}}(N_{\neg\alpha})$. With respect to $\mathcal{N}$, we get: $Pr(o) = \sum_{N\in\mathbb{N}_{1\alpha}} Pr_{\mathcal{N}}(N) + \sum_{N\in\mathbb{N}_{1\neg\alpha}} Pr_{\mathcal{N}}(N)$. With respect to $\mathcal{N}_T$, we get: $Pr(o) = \sum_{N\in\mathbb{N}_{1\alpha}} Pr_{\mathcal{N}_T}(N)$. Therefore, $Pr(o)$ is same with respect to $\mathcal{N}$ and $\mathcal{N}_T$. □

The above lemma ensures that the most probable optimal outcome in the Trans-net will give the most probable optimal outcome in the PTCP-net.

Definition 15. (Opt-net) The Opt-net $O_{\mathcal{N}_T}$ of a given Trans-net $\mathcal{N}_T$ is a BN with the following. Each variable X in $\mathcal{N}_T$ is also a variable X in $O_{\mathcal{N}_T}$. $D(X)$ in $O_{\mathcal{N}_T}$ are the values of X from $\mathcal{N}_T$ that rank first in at least one preference order with non-zero probability. All the pcp-arcs in $\mathcal{N}_T$ also exist in $O_{\mathcal{N}_T}$, which define the probabilistic dependencies over the variables in $O_{\mathcal{N}_T}$. $PT(X)$ in $O_{\mathcal{N}_T}$ is computed as: for each $x \in D(X)$ and $u \in D(Pa(X))$, $Pr(x|u) = \sum_{x \text{ ranks first in } \succ_X^u} Pr(\succ_X^u)$.

The purpose of an Opt-net is to represent the probability of each outcome that is optimal in at least one TCP-net. The Opt-net, corresponding to the Trans-net of Figure 8, is shown in Figure 9. The variables are A, B, C and D. The PT of each variable is determined from the corresponding PCPT.

Lemma 4. *An outcome o is optimal in at least one compatible TCP-net N of a conditionally acyclic PTCP-net $\mathcal{N}$ on V, if and only if o is an assignment with respect to the corresponding Opt-net $O_{\mathcal{N}_T}$.*

Proof. ($\Rightarrow$) Let $o = x_1 x_2 \cdots x_n$ be the optimal outcome of N. o is obtained using the forward sweep procedure (Boutilier et al., 2004; Brafman et al., 2006), that ensures that, for every $X_i \in V$, $X_i = x_i$ in o ranks first for the preference order over $D(X_i)$ of N given instantiation of $Pa(X_i)$. By Definition 15, the variable X_i also exists in $O_{\mathcal{N}_T}$ where $\{x_i\} \subseteq D(X_i)$ in $O_{\mathcal{N}_T}$. Therefore, o is an assignment of $O_{\mathcal{N}_T}$.

($\Leftarrow$) Let $o = x_1 x_2 \cdots x_n$ be an assignment of $O_{\mathcal{N}_T}$. We get that every $x_i \in o$ ranks first in a preference order over $D(X_i)$ of $\mathcal{N}_T$ with a non-zero probability, where $x_i \in D(X_i)$. By taking this preference order for every $x_i \in o$ from $\mathcal{N}_T$, we can find a compatible TCP-net of $\mathcal{N}_T$ in which o is optimal. □

Note that, for a given PTCP-net $\mathcal{N}$ and its Opt-net $O_{\mathcal{N}_T}$, there is a one-to-one correspondence between the assignments of $O_{\mathcal{N}_T}$ and the outcomes that are optimal in at least one compatible TCP-net. The probability of an outcome to be optimal in $\mathcal{N}$ is the joint probability of the corresponding assignment in $O_{\mathcal{N}_T}$. Thus the most probable optimal outcome in $\mathcal{N}$ can be found by finding the assignment with the highest joint probability in $O_{\mathcal{N}_T}$.

```
Algorithm 1. $FindMPOO(\mathcal{N}_T, V, (o, Pr(o)))$
Input: Conditionally acyclic Trans-net $\mathcal{N}_T$, set of variables $V$ on $\mathcal{N}_T$, partial assignment and its probability to be
  optimal $(o, Pr(o))$ on $V_{orig} - V$
Output: The most probable optimal outcome $MPOO$ and its probability $Pr(MPOO)$ to be optimal
  1. If $V$ is null, then return $(o, Pr(o))$
  2. Choose any variable $X$ such that there is no pcp-arc $\overrightarrow{(Y, X)}$ and no conditional pci-arc $(Y, X)$ in $\mathcal{N}_T$
  3. Set $V = V - \{X\}$
  4. For every $x_i \in D(X)$ such that $x_i$ ranks first in at least one preference order on $D(X)$
     (a) Set $o = ox_i$
     (b) Set $Pr(o) = Pr(o) \times \sum_{x_i \text{ ranks first in } \succ^u_X} Pr(\succ^u_X)$
     (c) Construct a sub Trans-net $\mathcal{N}_{Ti}$ as follows:
           i. Remove $X$ and its associated unconditional pci-arcs from $\mathcal{N}_T$
          ii. For every variable $Y$ such that $\overrightarrow{(X, Y)}$, revisit $Y$'s PCPT by restricting each row to $X = x_i$
         iii. For every conditional pci-arc $(Y, Z)$ such that $X \subseteq S(Y, Z)$, revisit $PCIT(Y, Z)$ by restricting each row to
              $X = x_i$
     (d) If $V$ is null then
             $(o', Pr(o')) = FindMPOO(\mathcal{N}_{Ti}, V, (o, Pr(o)))$
             If $Pr(o') > Pr(MPOO)$ then
                 $MPOO = o'$
                 $Pr(MPOO) = Pr(o')$
         Else
             $FindMPOO(\mathcal{N}_{Ti}, V, (o, Pr(o)))$
  5. End
```

Now, we aim to build an algorithm to find the most probable optimal outcome. We call it *Find Most Probable Optimal Outcome* (FindMPOO). FindMPOO is a recursive algorithm, and each recursive call accepts three parameters. The first parameter is a sub Trans-net $\mathcal{N}_T$, which is initially the given Trans-net $\mathcal{N}_{T_{orig}}$ of the PTCP-net. The second parameter is the set of variables V on $\mathcal{N}_T$, which is initially V_{orig}. The third parameter is a pair $(o, Pr(o))$, where o indicates a partial assignment on $V_{orig} - V$ and $Pr(o)$ is the probability of o to be optimal. o is initially *null* and $Pr(o)$ is initially 1. The algorithm returns the most probable optimal outcome in a global variable $MPOO$ and its probability to be optimal in another global variable $Pr(MPOO)$. $MPOO$ and $Pr(MPOO)$ are initially set to *null* and 0 correspondingly.

In step 1 of FindMPOO, the termination criterion is checked. If the variable set is *null*, an optimal outcome with its probability is returned. In step 2, a root variable is chosen (see Lemma 1). In step 3, the variable set is reduced by 1 that indicates the convergence of the algorithm. Steps 4(a)-(d) continue for each of the values of the chosen variable, which ranks first in at least one preference order. A value ranks first in a preference order indicates that there is at least one compatible TCP-net such that the value is in the optimal outcome. In step 4(a), the input assignment is updated by adding the current value. In step 4(b), the assignment's probability to be optimal is updated by multiplying the probability of the preference orders in which the value ranks first. This updated assignment and its probability to be optimal will be propagated as input in the next call of the algorithm. In step 4(c), a reduced Trans-net is formed that will also be one of the parameters in the next call of the algorithm. In step 4(d), if the variable set is *null*, it confirms that the algorithm finds a possible solution. The next call will return an outcome with its probability to be optimal. The outcome and the probability are stored in o' and $Pr(o')$ correspondingly. This outcome is compared with the so far most probable optimal outcome (MPOO). If the outcome dominates MPOO in terms of probability, it is made MPOO. After the complete execution of the algorithm, the most probable optimal outcome and its probability are stored in MPOO and Pr(MPOO) correspondingly.

Example 11. *Let us apply the FindMPOO algorithm to find the most probable optimal outcome and its probability for the Trans-net in Figure 8. The initial set of variables is* $V = \{A, B, C, D\}$*. The initial assignment and its probability are null and* 1*. For the initial call of FindMPOO, in step 2, the variable D is chosen. In step 3, V is set to* $\{A, B, C\}$*. In* $PCPT(D)$*,* d_1 *ranks first in* $d_1 \succ d_2$ *and* d_2 *ranks first in* $d_2 \succ d_1$*. So steps 4(a)-4(d) will repeat for* d_1 *and* d_2 *individually.*

Firstly, let us consider steps 4(a)-4(d) for d_1. In steps 4(a)-(b), o and Pr(o) are updated as d_1 and 0.3 *correspondingly. In step 4(c), a sub Trans-net $\mathcal{N}_{T1}$ is constructed by removing D from the original Trans-net and restricting $PCIT(A, B)$ to $D = d_1$. This sub Trans-net $\mathcal{N}_{T1}$ is shown in Figure 10 (ii). In step 4(d), V is not null and FindMPOO is again called with the parameters $\mathcal{N}_{T1}$, $\{A, B, C\}$ and $(d_1, 0.3)$ (let's say it is call 2). $(d_1, 0.3)$ is shown with an outgoing edge from D in Figure 10 (i). For this call 2, in step 2, we have two options to choose A or B. A is chosen that is depicted, in Figure 10 (i), by the outgoing edge from D directing to A. In step 3, V is set to $\{B, C\}$. In $PCPT(A)$, a_1 ranks first in $a_1 \succ a_2$ and a_2 ranks first in $a_2 \succ a_1$. So steps 4(a)-4(d) will repeat for a_1 and a_2 individually.*

Let us consider steps 4(a)-4(d) for a_1. In steps 4(a)-(b), o and Pr(o) are updated as a_1d_1 and 0.24 *correspondingly. In step 4(c), a sub Trans-net $\mathcal{N}_{T2}$ is constructed by removing A and unconditional pci-arc (A, B), and restricting $PCPT(C)$ to $A = a_1$. This sub Trans-net $\mathcal{N}_{T2}$ is shown in Figure 10 (iii). In step 4(d), V is not null and FindMPOO is again called with the parameters $\mathcal{N}_{T2}$, $\{B, C\}$ and $(a_1d_1, 0.24)$ (call 3). $(a_1d_1, 0.24)$ is shown with an outgoing edge from A in Figure 10 (i). For this call 3, in step 2, B is chosen that is depicted, in Figure 10 (i), by the outgoing edge from A directing to B. In step 3, V is set to $\{C\}$. In $PCPT(B)$, b_1 ranks first in $b_1 \succ b_2$ and b_2 ranks first in $b_2 \succ b_1$. So steps 4(a)-4(d) will repeat for b_1 and b_2 individually.*

Consider steps 4(a)-4(d) for b_1. In steps 4(a)-(b), o and Pr(o) are updated as $a_1b_1d_1$ and 0.168 *correspondingly. In step 4(c), a sub Trans-net $\mathcal{N}_{T3}$ is constructed by removing B and restricting $PCPT(C)$ to $B = b_1$. This sub Trans-net $\mathcal{N}_{T3}$ is shown in Figure 10 (iv). In step 4(d), V is not null and FindMPOO is again called with the parameters $\mathcal{N}_{T3}$, $\{C\}$ and $(a_1b_1d_1, 0.168)$ (call 4). $(a_1b_1d_1, 0.168)$ is shown with an outgoing edge from B in Figure 10 (i). For this call 4, in step 2, C is chosen that is depicted, in Figure 10 (i), by the outgoing edge from B directing to C. In step 3, V is set to null. In $PCPT(C)$, c_1 ranks first in $c_1 \succ c_2$ and c_2 ranks first in $c_2 \succ c_1$. So steps 4(a)-4(d) will repeat for c_1 and c_2 individually.*

Let us consider steps 4(a)-4(d) for c_1. In steps 4(a)-(b), o and Pr(o) are updated as $a_1b_1c_1d_1$ and 0.095 *correspondingly. In step 4(c), the sub Trans-net $\mathcal{N}_{T4}$ after removing C is null. In step 4(d), V is null that indicates that an outcome with its probability to be optimal is found. FindMPOO is called with the parameters $\mathcal{N}_{T4}$, null and $(a_1b_1c_1d_1, 0.095)$ (call 5). The return of call 5 (that is $(a_1b_1c_1d_1, 0.095)$) is stored in $(o', Pr(o'))$. It is depicted in Figure 10 (i) with an outgoing arc from C to null. Since $Pr(o') > Pr(MPOO)$, $a_1b_1c_1d_1$ and* 0.095 *are stored in $MPOO$ and $Pr(MPOO)$ correspondingly. The "bold" $(a_1b_1c_1d_1, 0.095)$ in Figure 10 (i) indicates that this outcome is the most probable optimal outcome so far with the algorithm.*

Now in call 4, consider the next iteration of steps 4(a)-4(b) for c_2. Similarly, this iteration will give us the outcome $a_1b_1c_2d_1$ with its probability to be optimal 0.073. *The probability of $a_1b_1c_2d_1$ is not greater than the probability of $a_1b_1c_1d_1$. It indicates that $a_1b_1c_1d_1$ is still the most probable optimal outcome so far and $(a_1b_1c_2d_1, 0.072)$ is not made "bold" in Figure 10 (i). The Figure 10 (i) for complete execution of the algorithm shows that $a_1b_1c_1d_2$ is the most probable optimal outcome with its probability* 0.221. □

We now provide the time complexity of FindMPOO.

Lemma 5. *Given a Trans-net $\mathcal{N}_T$ with n number of variables. If m is the number of values of a variable, FindMPOO requires $O(m^n n)$ in worst case, while it is $O(n)$ in best case.*

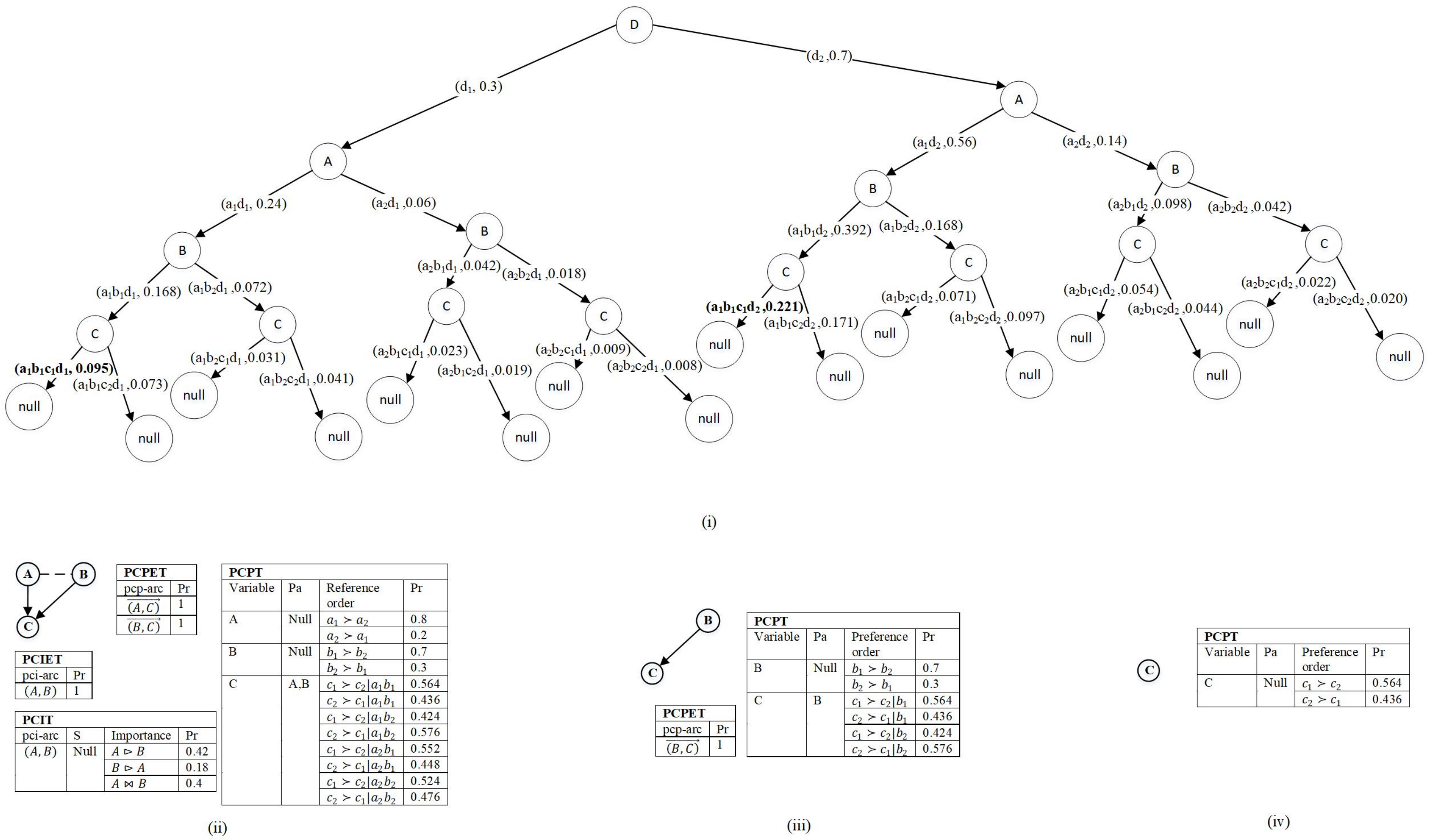
Proof. In worst case, number of possible outcomes is m^n. Every outcome requires $n - 1$ multiplications to find the probability to be optimal. To find the most probable outcome, $m^n - 1$ comparisons are needed. So, we get $m^n(n-1)+m^n-1$, which is $O(m^n n)$.

On the other hand, in best case, every variable can contain only a single preference order over its domain given the parent set. In this case, there will be one outcome. To compute its probability, we need $n-1$ multiplications, which is in $O(n)$. □

5. Application in Recommender Systems

In this section, we aim to extend *Collaborative Filtering* (CF) recommender system with users' qualitative and conditional preferences on the item attributes. We argue an intuitive model, *CF with Conditional Preferences* (CFCP), which will possibly overcome the inherent limitations of CF. However, an experimental validation is left for a future research. CFCP is a basic CF extended to two knowledge sources: the items have a finite set of attributes and the users express conditional preferences over the attributes. The PTCP-net is used to aggregate the users' preferences. Given the CFCP model, we need to perform two tasks: (1) how to estimate the rating of an item to a user and (2) how to tackle new user and new item.

In a conventional CF, to estimate the rating of an item to a user, the ratings of the similar users on the same item are used. Some techniques (such as correlation-based (Resnick et al., 1994) and cosine-based (Breese et al., 1998)) are



PCPET	
pcp-arc	Pr
$\overrightarrow{(A,C)}$	1
$\overrightarrow{(B,C)}$	1

PCIET	
pci-arc	Pr
(A,B)	1

PCIT			
pci-arc	S	Importance	Pr
(A,B)	Null	$A \rhd B$	0.42
		$B \rhd A$	0.18
		$A \bowtie B$	0.4

PCPT			
Variable	Pa	Reference order	Pr
A	Null	$a_1 \succ a_2$	0.8
		$a_2 \succ a_1$	0.2
B	Null	$b_1 \succ b_2$	0.7
		$b_2 \succ b_1$	0.3
C	A,B	$c_1 \succ c_2 \mid a_1b_1$	0.564
		$c_2 \succ c_1 \mid a_1b_1$	0.436
		$c_1 \succ c_2 \mid a_1b_2$	0.424
		$c_2 \succ c_1 \mid a_1b_2$	0.576
		$c_1 \succ c_2 \mid a_2b_1$	0.552
		$c_2 \succ c_1 \mid a_2b_1$	0.448
		$c_1 \succ c_2 \mid a_2b_2$	0.524
		$c_2 \succ c_1 \mid a_2b_2$	0.476

PCPET	
pcp-arc	Pr
$\overrightarrow{(B,C)}$	1

PCPT			
Variable	Pa	Preference order	Pr
B	Null	$b_1 \succ b_2$	0.7
		$b_2 \succ b_1$	0.3
C	B	$c_1 \succ c_2 \mid b_1$	0.564
		$c_2 \succ c_1 \mid b_1$	0.436
		$c_1 \succ c_2 \mid b_2$	0.424
		$c_2 \succ c_1 \mid b_2$	0.576

PCPT			
Variable	Pa	Preference order	Pr
C	Null	$c_1 \succ c_2$	0.564
		$c_2 \succ c_1$	0.436

Figure 10. An illustration of the Algorithm 1 (FindMPOO).

used to measure the similarity between two users. For our convenience, we call this "quantitative similarity" because it is measured from the users' quantitative ratings. Then, the similar users' ratings are aggregated using the measured similarities to estimate the rating. In CFCP, we introduce "qualitative similarity" that indicates the similarity between two users in terms of their conditional preferences (i.e., it is a score that indicates the number of common preference and relative importance statements between two users). We use both quantitative and qualitative similarities to calculate the actual similarity and this actual similarity is used to compute the rating.

When there is a new user (i.e., the user has not rated any item yet and so the quantitative similarity with other users cannot be calculated) in the system, we consider only qualitative similarity to estimate the rating of the items. In this case, if the user's preferences are not given, we can utilize the most probable compatible TCP-net that most likely represents the preferences of an unknown user. After the ratings are estimated, the items with the highest estimated rating are recommended to the user. We tackle a new item (i.e., the item has not been rated by any users) as follows. We can find the users such that their preferences are encoded with any of the TCP-nets in which the optimal outcome is same to the values of the item attributes. If the values of the item attributes are not given, we can utilize the most probable optimal outcome that most likely represents the values of an unknown item. Then, the new items are recommended to those users.

Formally, the CFCP is formulated with the following components.

1. There is a set of items S. The set of qualitative attributes for the items is $\{X_1, X_2, \cdots, X_k\}$. The domains of the attributes are $D(X_1), D(X_2), \cdots, D(X_k)$.

2. The rating of an item by a user is an integer ranging from 1 to r.

3. There is a set of users C. Each user expresses conditional preference statements over the domains of the attributes. Each user might have conditional relative importance statements between two attributes. The users' choices are aggregated with a PTCP-net.

4. u be a utility function that measures the usefulness of item $s \in S$ to user $c \in C$, i.e., $u : C \times S \Rightarrow R$, where R is a totally ordered set (e.g., non-negative integers or real numbers within a certain range). Then, for each user $c \in C$, we want to choose such item $s' \in S$ that maximizes the user's utility. More formally: $\forall c \in C, s'_c = \arg\max_{s \in S} u(c, s)$.

We address the following tasks in CFCP.

Estimating unknown utility: The value of the unknown utility $u(c, s)$ for user c and item s is computed as: $u(c, s) = k \sum_{c' \in \hat{C}} sim(c, c')u(c', s)$. $\hat{C}$ denotes the set of users with non-zero $sim(c, c')$. k is a normalizing factor selected as $k = 1/ \sum_{c' \in \hat{C}} |sim(c, c')|$. $sim(c, c')$ indicates a similarity measure between the users c and c'. $sim(c, c')$ is computed as: $sim_{quantitative}(c, c') + sim_{qualitative}(c, c')$. $sim_{quantitative}(c, c')$ indicates quantitative similarity and $sim_{qualitative}(c, c')$ indicates qualitative similarity. The-state-of-the-arts approaches, such as correlation-based (Resnick et al., 1994) and cosine-based (Breese et al., 1998), are used to compute the quantitative similarity. In the near future, we plan to find heuristics to measure qualitative similarity between two users. For example, qualitative similarity is a score that indicates the number of common preference and relative importance statements between two users. There can be a number of scenarios. (1) For both of the users, conditional preferences over the all attributes are given. (2) One of the users is given with partial preferences. (3) Both of the users have partial preferences.

Tackling new user and new item: A new user will fall in one of the two categories. (1) The user expresses preferences over the all attributes or a subset of attributes. In this case, we can find the qualitative similarities with the other users. For estimating the rating, only these qualitative similarities are used. (2) The user expresses no preferences over the attributes. From the PTCP-net, we find the most probable TCP-net that encodes a set of preferences that most likely map onto an unknown user. Therefore, the new user is assumed to have this set of preferences. For estimating the rating, again, only the qualitative similarities are used. Now, we describe the strategy to tackle a new item. A new item will fall in one of the two categories. (1) The values are known for the all attributes (or a subset of attributes). This item is recommended to a user, where the item's values are equal to (or subset of) the optimal outcome of the TCP-net that represents the user's preferences. (2) The attribute values are unknown. In this case, we find the most probable optimal outcome for the PTCP-net. The item is recommended to a user, where the most probable optimal outcome is equal to the optimal outcome of the TCP-net that represents the user's preferences.

6. Conclusion and Future Work

We have proposed the PTCP-net that is a probabilistic extension of the TCP-net. The extension consists of adding the probability over preference orders and relative importance statements for two reasons. First, in a context of multi-users, users' preference and relative importance statements may vary from one user to another. In this case, the probability is used for the purpose of aggregation. We show that a PTCP-net represents a probability distribution over a set of TCP-nets, where each has the same set of variables and their domains but differ in preference and relative importance statements. Second, a single user might have uncertain preference and relative importance statements, which a PTCP-net can represent.

Two types of queries have been evaluated: finding the most probable compatible TCP-net and finding the most probable optimal outcome. In a context of multiple users, the most probable TCP-net represents the preference and relative importance statements that an unknown user most likely expresses. The most probable optimal outcome is most likely the outcome to be recommended to an unknown user.

We propose an extension of the collaborative filtering type recommender system in which a PTCP-net is used to aggregate users' preferences. We intuitively argue that this model will overcome the underlying issues of recommender system, however experimental validation is left for a future research.

In the near future, we are planning to work on dominance query, i.e., finding the probability that an outcome is preferred to another outcome. Note that this will be done after we explore the dominance testing with respect to TCP-nets (Brafman et al., 2006). Moreover, we plan to extend the PTCP-net model with hard constraints using the Constraint Satisfaction Problem framework (Dechter, 2003) as was done for TCP-nets (Zhang et al., 2015). This is particularly important given that, in a system of multiple unknown users, each user can have a di_erent set of constraints.

Acknowledgment

This research was funded by Natural Sciences and Engineering Research Council of Canada (RGPIN-2016-05673).

References

Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-theart and possible extensions. *IEEE Transactions on Knowledge and Data Engineering, 17*(6), 734–749. https://doi.org/10.1109/TKDE.2005.99

Craig, B., Ronen, I. B., Carmel, D., Holger, H. H., & David, P. (2004). CP-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements. *J. Artif. Intell. Res. (JAIR), 21,* 135–191. https://doi.org/10.1613/jair.1234

Cristina, C., Judy, G., Nicholas, M., Francesca, R., & Venable, K. B. (2013). Updates and uncertainty in CP-nets. In Proceedings of 26th Australasian Joint Conference on Artificial Intelligence, pages 301–312. Springer. https://doi.org/10.1007/978-3-319-03680-9 32

Cristina, C., Umberto, G., Judy, G., Nicholas, M., Francesca, R., & Venable, K. B. (2015). Reasoning with PCP-nets in a multi-agent context. In Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems, AAMAS '15, pages 969–977, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 978-1-4503-3413-6.

Damien, B., Bruno, Z., Helene, F., & Jerome, M. (2013). Probabilistic conditional preference networks. In Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence, pages 72–81.

Francesca, R., Kristen, B. V., & Toby, W. (2004). mCP nets: representing and reasoning with preferences of multiple agents. *In Proceedings of Nineteenth National Conference on Artificial Intelligence (AAAI-2004), 4,* 729–734.

Gabriella, P., Alexis, T., & Paolo, V. (2016). Preferences in artificial intelligence. *Annals of Mathematics and Artificial Intelligence, 77*(3), 361–401. https://doi.org/10.1007/s10472-015-9475-5

Johan, K. (2011). Most probable explanations in bayesian networks: Complexity and tractability. Int. J. Approx. *Reasoning, 52*(9), 1452–1469, December. ISSN 0888-613X. https://doi.org/10.1016/j.ijar.2011.08.003

John, S. B., David, H., & Carl, K. (1998). Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, pages 43–52. Morgan Kaufmann Publishers Inc.

Joseph, A. K., Bradley, N. M., David, M., Jonathan, L., Herlocker, Lee, R. G., & John, R. (1997). Grouplens: applying collaborative filtering to usenet news. *Communications of the ACM, 40*(3), 77–87. https://doi.org/10.1145/245108.245126

Judea, P. (1988). Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann.

Juntao, L., Zhijun, Y., Yi, X., Wenyu, L., & Caihua, Wu. (2013). Learning conditional preference network from noisy samples using hypothesis testing. *Knowledge-Based Systems, 40,* 7–16. https://doi.org/10.1016/j.knosys.2012.11.006

Nahla, B. A., Didier, D., H´ela Gouider, & Henri, P. (2016). Graphical models for preference representation: An overview. In Proceedings of 10th International Conference on Scalable Uncertainty Management (SUM), pages 96–111. https://doi.org/10.1007/978-3-319-45856-4 7

Paul, R., Neophytos, I., Mitesh, S., Peter, B., & John, R. (1994). Grouplens: an open architecture for collaborative filtering of netnews. In Proceedings of the 1994 ACM conference on Computer supported cooperative work, pages 175–186. ACM. https://doi.org/10.1145/245108.245126

Rina, D. (2003). Constraint processing. *Morgan Kaufmann.* https://doi.org/10.1016/B978-1-55860-890-0.X5000-2

Robin, B., Alexander, F., & Mehmet, H. G¨oker. (2011). Recommender systems: An overview. *AI Magazine, 32*(3), 13–18. https://doi.org/10.1609/aimag.v32i3.2361.

Ronald, R. Y. (2001). Fusion of multi-agent preference orderings. *Fuzzy Sets and Systems, 117*(1), 1–12. https://doi.org/10.1016/S0165-0114(98)00418-7

Ronen, I. B., & Carmel, D. (2002). TCP-nets for preference-based product configuration. In Proceedings of the Forth Workshop on Configuration (in ECAI-02), pages 101–106.

Ronen, I. B., Carmel, D., & Solomon, E. S. (2006). On graphical modeling of preference and importance. *Journal of Artificial Intelligence Research, 25,* 389–424. https://doi.org/10.1613/jair.1895

Rong, J., Luo, S., & Cheng, X. Z. (2002). Preference-based graphic models for collaborative filtering. In Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence, pages 329–336. Morgan Kaufmann Publishers Inc.

Shu, Zh., Malek, M., & Samira, S. (2015). Integrating TCP-nets and CSPs: The constrained TCP-net (CTCPnet)model. In Proceedings of International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, pages 201–211. Springer. https://doi.org/10.1007/978-3-319-19066-220

Sleh, El F., Malek, M., Nahla, Ben, A., & Eisa, A. (2017). Representing and reasoning with constrained PCPnets. In Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC), pages 1007–1012. IEEE. https://doi.org/10.1109/SMC.2017.8122742

Sultan, A., & Malek, M. (2017). Probabilistic TCP-net. In Proceedings of 30th Canadian Conference on Artificial Intelligence, pages 293–304. Springer. https://doi.org/10.1007/978-3-319-57351-9 34

Copyrights