

Received July 23, 2021, accepted September 4, 2021, date of publication September 22, 2021, date of current version September 29, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3114637

Conditional Preference Networks for Cloud Service Selection and Ranking With Many Irrelevant Attributes

ABDULAZIZ ALASHAIKH¹ AND EISA ALANAZI²

¹Department of Computer and Network Engineering, University of Jeddah, Jeddah 23218, Saudi Arabia

²Department of Computer Science, College of Computers and Information Systems, Umm Al-Qura University, Makkah 21421, Saudi Arabia

Corresponding author: Abdulaziz Alashaikh (asalashaikh@uj.edu.sa)

This work was supported by the Deanship of Scientific Research (DSR), University of Jeddah, Saudi Arabia, under Grant UJ-20-050-DR.

ABSTRACT Cloud service selection and ranking are two different but related important tasks in the realm of cloud computing. Given the rapid growth in the number of available services offered by cloud providers with different quality of service, these tasks are becoming more and more challenging to be managed by the end customers. Furthermore, many services in the cloud are defined over a large set of attributes where the potential customer is interested on only a small subset of them. The other attributes are viewed as irrelevant to the customer selection and ranking strategies. In this work, we show how to employ conditional preferential dependencies in the domain of cloud service selection and ranking. In particular, we use Conditional Preference Networks (CP-nets) as a compact model to represent and reason with customer's preferences and criteria interdependencies. We show how to select the best service and how to rank, in non-increasing order, a set of services given possibly incomplete information in the customer's CP-net. Our experimental results prove the feasibility of CP-nets as a mechanism for selecting and ranking hundreds of services defined on hundreds of attributes with complex interdependencies.

INDEX TERMS Conditional preferences, CP-nets, cloud service selection, ranking, decision making.

I. INTRODUCTION


Cloud computing has emerged as the standard paradigm for many organizations when looking to expand or establish IT services or infrastructure [1]. This has given a rise to a market where a large number of cloud services are offered to potential customers with different attributes and levels of quality of service (QoS). Hence, the task of identifying the best service or ranking the available services are becoming more challenging and drawing a lot of attention recently [2]–[8].

Typically, services are perceived as a mapping of values to a set of criteria or attributes¹ that describe technical and non-technical service requirements such as capacity, memory, QoS parameters, and other attributes defined by the potential customer or provider. One of the noted issues in the

literature is the lack of formal approaches to tackle criteria interdependencies [3], [8]–[10]. It is commonly assumed that the preference relation of one attribute is independent from all other attributes when selecting a service. However, it is natural for the preference of one criterion to depend on the values of other criteria. Another issue is the fact that services are usually defined over a large number of attributes where only a small subset of them are relevant to the customer preferences.

In decision theory, a possible remedy for the above two issues is by utilizing the notion of *preferential dependency* [11] which has been discussed and applied in many contexts [12]–[14] and employed within the model of Conditional Preference Networks (CP-nets) [15]. CP-nets allow customers to identify their relevant attributes and express the preferences on smaller and more manageable set of solutions or services.

Informally, a CP-net is a collection of qualitative preference statements having the form: $x : y \succ y'$ which means the preference of variable Y with possible values y and y' is

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Wang .

¹We use the words attributes, variables, and criteria interchangeably throughout the paper.

conditioned upon the value of another variable X whereas the symbol \succ denotes the preference relation. Specifically, when $X = x$, y is preferred to y' . The preference holds only when x is true. This way, the preference of Y is independent from all other attributes given the values of its dependent variables (*i.e.*, X). In this work, we show that representing/modeling conditional preferential dependencies via CP-nets is an efficient/suitable approach for the cloud service selection problem even when there is a large (*i.e.*, hundreds) number of attributes describing every service. Furthermore, we consider the case when customers supply partial preferences (*i.e.*, incomplete CP-nets) and the system is asked to return a ranked list on the available services, in a non-increasing order, based on the given information. Such results confirm our hypothesis that CP-nets are a valid option in representing and reasoning about decision making issues related to the cloud.

Recently, there have been some attempts to tackle complex criteria interactions within the problem of cloud service selection [3], [5], [9], [16]. However, they are either of a quantitative nature where an importance weight for every criterion is computed based on pairwise comparisons to measure the dependencies [2], [9] or inefficient in the sense that they assume the existence of a graph that is exponential in the number of attributes [5]. Moreover, almost all the proposed approaches in the literature were tested against random or real instances with small number of attributes (most of the time up to 20 criteria). It is practically important for the proposed approach to be scalable to hundreds of attributes and services.

To the best of our knowledge, there exists no purely qualitative approach to the problem of service selection and ranking when the CP-net is assumed to be incomplete with hundreds of attributes in the domain.

The paper is organized as follows: Next section presents an example to motivate utilizing CP-nets with the domain of cloud service selection and ranking. Related approaches are discussed in Section III. Section IV presents the background information on CP-nets. Section V presents the proposed approach to select best service and rank available services given the customer's CP-net. Experiment results showing scalability of the approach is presented in Section VI. Finally, conclusion remarks and future work is discussed in Section VII.

II. MOTIVATION EXAMPLE

Consider a customer, Emma, who wants to select among all the available services, the one that fits her preferences the most. At first, Emma always prefers to have a service from providers with high Reputation (*Rep*), and high Availability (*A*), Security (*Sec*), Reliability (*Rel*), and Successability (*Suc*). She categorizes the possible values for each of the above attributes or criteria into a set of three values high (*h*), medium (*m*), or low (*l*) for her convenience. Furthermore, Emma's preference on Replicability (*Rec*) depends on the attribute Cost (*C*) of the service. If the cost is high, she prefers a high replicability to medium or low. As for the hosted region

of the service, she always prefers Europe (*EU*) to North America (*NA*) or the region of Middle East and North Africa (*MENA*). Finally, the cost value depends to the reputation of the provider and the availability of the service. Emma is not willing to invest in a high cost service if the reputation is low and the availability is medium or low.

Thus, among all the services attributes, *Rep*, *A*, *Sec*, *Rel*, *Suc*, *Rec*, *Reg* and *C* are the relevant ones to her preferences. For the preference of Reputation, we can conclude that $h \succ m \succ l$, for instance, where \succ is the preference relation (formal definitions will be given in Section IV) and $x \succ x'$ means " x is preferred to x' for the attribute under consideration". Emma's knowledge shows some interdependencies within criteria or attributes. This is evident in her preference over *Rec* criterion where she prefers $Rec = h$ (*i.e.*, high Replicability) if the service cost was high (*i.e.*, $C = h$).

Figure 1 shows the CP-net of Emma based on the above eight attributes. Each node in the graph represents a relevant criterion or an attribute, whereas edges represent dependencies. Each node is also annotated with a Conditional Preference Table (CPT) that shows preference relations of this attribute given assignments of its dependent variables (also known as parents). For instance, Emma's preference on the cost depends on the values of Reputation and Availability (referred to as the *parent* attributes of Cost). Specifically, if the provider has low reputation and medium or low availability, she is not interested in investing high cost of money in the service. The CPT of the Cost shows for some values of the parents (*i.e.*, Reputation and Availability), a preference relation on the values of the Cost.

It is important to notice that the CP-net shown in Figure 1 has some incomplete preference information. For example, the preference of Replicability when Cost is medium or low is not given and the network shows Emma's preference only when the Cost is high. Such missing knowledge (or incomplete preference) is natural in everyday's decisions and needs to be handled properly within cloud service selection and ranking approaches. It is rarely the case where the potential customer is able to give complete preferences on different attributes depending on different parents.

Technically speaking, a service is a mapping from every attribute in the problem to an element of its *domain*. For instance, in the above CP-net, all attributes have three possible values or domain (h, m, l) except for Region where *EU*, *NA*, and *MENA* are the possible values. Any such mapping is considered a potential service and in our case we have 3^8 possible services.

This albeit is a standard assumption in many works, it is unrealistic in practical problems as the ones we are discussing here. Usually, the customer has an access to a large set of services and she wishes to select or rank them based on her intuition. Thus, in this work, we always assume a subset of the services are available and our goal is to rank them based on the customer's CP-net.

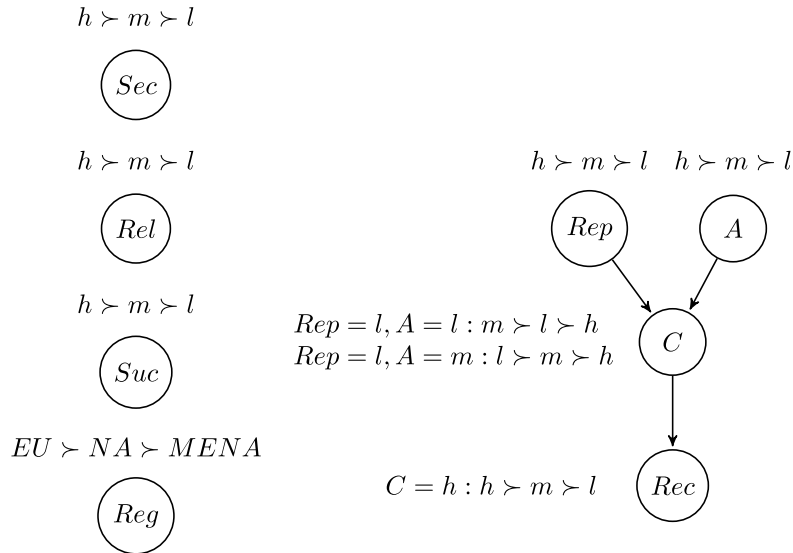


FIGURE 1. Emma's network showing different interdependencies among criteria.

III. RELATED WORK

The cloud service selection and ranking problem has attracted a lot of attention in the last recent years [9], [10], [17]. The proposed approaches can be roughly classified into either quantitative or qualitative multi criteria decision making or multiobjective optimization approaches.

It is natural to view the problem as a multi criteria decision making (MCDM) problem. Indeed, many existing works in the literature adopt this view and utilize different MCDM techniques including TOPSIS [18] and AHP [7], [19], [20] to tackle the problem. For instance, an MCDM was proposed in [18] utilizing TOPSIS technique to rank available cloud services to the customer and experimentally compared it to AHP and fuzzy AHP techniques. These approaches usually involve the criteria importance weight that is assumed to be computed and then select the most appropriate decision among possible alternatives.

The closest work to ours is probably the work of Sun *et al.* [9] where an MCDM approach was proposed taking into account the criteria interdependencies. The authors in this work compute the importance value for every criterion via pairwise comparisons to measure the interaction degree between every pair of criteria. The work supports three types of interactions: conflicting, supporting, and independent. The strength of interactions is determined from historical data where the problem was solved using non-linear constraint optimization techniques. Chung and Seo [21] presented an analytical network process (ANP) method to select best IaaS provider. In [3], a hybrid MCDM approach based on ANP was proposed where cloud providers were clustered into groups and criteria interactions were identified based on representative criteria in each cluster.

There exists several attempts on utilizing CP-nets in different tasks related to cloud computing [2], [5], [6], [22], [23] including virtual machine placement [23] and service

composition [2]. Alashaikh and Alanazi [23] tackled the optimization problem of virtualization in cloud data centers where the virtual machine owner has a simple CP-net representing customer's preferences over different attributes related to the virtual machine.

Fattah *et al.* [2] considered the problem of finding the optimal set of requests for an IaaS provider, *i.e.*, a problem known as service composition. It was assumed that there exists several CP-nets where every customer is associated with a CP-net and they are compared against the provider's CP-net. The customers with similar preferences to the provider preferences are then recommended.

Another interesting work is [5], [16] where authors proposed an economical model based on a temporal variant of CP-nets for long term IaaS provider profit. In principle, it is assumed that the preferences change over time and the induced graph of the IaaS provider, which is exponential in the number of attributes, is given (see Definition 7 in the following section). Then, the model tries to match the incoming requests to the provider's preferences.

IV. CONDITIONAL PREFERENCE NETWORKS (CP-NETS)

We assume a problem domain that can be factored into a finite set of attributes $\mathcal{V} = \{X_1, X_2, \dots, X_n\}$. Each variable $X_i \in \mathcal{V}$ has a set of possible values (its domain) $dom(X_i) = \{x_1^i, x_2^i, \dots, x_m^i\}$. For a subset $Y = \{X_1, X_2, \dots, X_j\}$ of \mathcal{V} , we use $dom(Y)$ to denote the elements of the Cartesian product $dom(X_1) \times dom(X_2) \times \dots \times dom(X_j)$. An assignment is an arbitrary element of $dom(Y)$. For an assignment s of $Y \subseteq \mathcal{V}$, we use $s[X]$ to refer to the values of $X \subseteq Y$ in s . For the special case when $Y = \mathcal{V}$, the elements of $dom(Y)$ is known as solutions and we refer to them as \mathcal{S} . Indeed, there are $|dom(X_1) \times dom(X_2) \times \dots \times dom(X_n)|$ possible solutions in this case. We use xy to denote concatenating the assignment $y \in dom(Y)$ to another assignment $x \in dom(X)$ for two

mutually exclusive sets X and Y of \mathcal{V} . Such problem domains are known as *combinatorial domains*.

Definition 1: A preference \succ on a set \mathcal{S} is an *irreflexive, antisymmetric, and transitive binary relation*.

\succ is also known as a strict partial order on \mathcal{S} . For any two elements $s, s' \in \mathcal{S}$, we write $s \succ s'$ to refer to the fact that s is strictly better (or preferred to) s' , and $s \not\succ s'$ to denote that s is not strictly preferred to s' . Also, s and s' are said to be *incomparable* if $s \not\succ s'$ and $s' \not\succ s$ holds otherwise they are *comparable*. Finally, \succ is a *total order* if any two elements in \mathcal{S} are comparable. Given that the size of \mathcal{S} is exponential in the number of variables $|\mathcal{V}| = n$, it is practically infeasible and cognitively expensive to work directly on \mathcal{S} . Fortunately, one can reduce the preference effort to smaller variables domains when certain structural independencies hold.

Definition 2 ([15]): X is said to be *preferentially independent from another set Y* iff

$$xy \succeq x'y \iff xy' \succeq x'y',$$

for all $x, x' \in \text{dom}(X)$ and $y, y' \in \text{dom}(Y)$.

In essence, X is preferentially independent from another set Y in case the preference given to the set X does not change no matter what Y is. That is, knowing Y 's values does not add anything new to the preference relation defined on the set X . There is also a more fine-grained conditional version of preferential dependency stated as follows

Definition 3 ([15]): Consider X, Y and Z to be disjoint sets of \mathcal{V} where $X \cup Y \cup Z = \mathcal{V}$. We say that X is *independent from Y conditioned on another set Z* iff

$$xyz \succeq x'yz \iff xy'z \succeq x'y'z,$$

for all $x, x' \in \text{dom}(X)$, $y, y' \in \text{dom}(Y)$ and $z \in \text{dom}(Z)$.

Thus, the notion of (conditional) preferential independency allow us to compactly articulate and represent the preference of variable X_i regardless of how large the domain is. We are only required to supply locally how different elements of $\text{dom}(X_i)$ stand with different assignments of the dependent variables Y while ignoring the assignments of all other variables in $\mathcal{V} \setminus \{Y \cup X_i\}$ which are independent of X_i . Then, an aggregation operation over variables relations is needed to compare the elements of \mathcal{S} .

In what follows, we show that such dependencies comes handy given a set of conditional preference statements and we outline the basics of CP-nets as presented in the literature [15].

Definition 4: Given an assignment $y \in \text{dom}(Y)$ for an arbitrary subset $Y \subseteq \mathcal{V} \setminus X_i$, a *conditional preference statement for X_i with respect to y* , denoted as \succ_y^i , is a preference relation over $\text{dom}(X_i)$.

We abuse the notation and write \succ^i in the context of y to refer the conditional statement \succ_y^i , which is also written as $y \succ^i$.

Example 1: Consider $\mathcal{V} = \{X_1, X_2\}$ with their identical domain $\text{dom}(X_i) = \{h, m, l\}$ shorthand for high, medium, and low. A conditional statement $h : l \succ^1 h$ is interpreted as “if X_2 's value is high then I prefer low to high as a value for X_1 ”

Note that the statement is a partial order as, for example, there is no information how $X_1 = h$ stands with respect to $X_1 = m$ when $X_2 = h$.

Definition 5: A *Conditional Preference Table for a variable X_i (written as $\text{CPT}(X_i)$) with respect to other variables $Y \subseteq \mathcal{V} \setminus X_i$ is a collection of conditional preference statements $\succ_{y_1}^i, \succ_{y_2}^i, \dots, \succ_{y_r}^i$ where $y_j \in \text{dom}(Y)$ for all $j \in \{1, 2, \dots, r\}$.*

Example 2: Consider again $\mathcal{V} = \{X_1, X_2\}$ and $\text{dom}(X_i) = \{h, m, l\}$. A possible $\text{CPT}(X_1)$ with respect to X_2 is

$$\begin{aligned} h &: h \succ^1 m \succ^1 l \\ m &: h \succ^1 l \succ^1 m \\ l &: m \succ^1 l \succ^1 h \end{aligned}$$

$\text{CPT}(X_i)$ is said to be *complete* if \succ_y^i is total and $\text{CPT}(X_i)$ shows a statement for every $y \in \text{dom}(Y)$, otherwise it is incomplete. The CPT in Example 2 is a complete one. It however will be incomplete if \succ^1 was partial in any context of $y \in \text{dom}(X_2)$ or if one or more statements are missing. Therefore, the incompleteness of $\text{CPT}(X_i)$ is either because there exists no statements for some elements of $\text{dom}(Y)$ or because, for some statements $y \succ^i$, the preference \succ^i is a partial order over $\text{dom}(X_i)$, or both. By $\text{Pa}(X_i)$ we refer to the set of variables Y whose assignments are the contexts in $\text{CPT}(X_i)$.

Definition 6: A *CP-net over a set of variables $\mathcal{V} = \{X_1, X_2, \dots, X_n\}$ is a directed graph whose nodes are \mathcal{V} and an edge (X_j, X_i) exists iff $X_j \in \text{Pa}(X_i)$ and each vertex $X_i \in \mathcal{V}$ is associated with $\text{CPT}(X_i)$ whose contexts are $\text{dom}(\text{Pa}(X_i))$.*

The CP-net graph shows the preferential dependencies that hold in the problem domain. That is, X_i is independent from all other variables given its parents $\text{Pa}(X_i)$. When developing an application, the CP-net can be constructed as follows:

- For every variable $X_i \in \mathcal{V}$, the user is asked to choose “parent variables” $\text{Pa}(X_i) \subseteq \mathcal{V} \setminus X_i$ that affect the preference relation of X_i .
- For some assignments $u \in \text{dom}(\text{Pa}(X_i))$, the user articulates preference statement \succ_u^i over $\text{dom}(X_i)$.

In case the user supplied a total order statement for every $u \in \text{dom}(\text{Pa}(X_i))$ then we have a complete CP-net, otherwise it is incomplete.

Example 3: Figure 2 shows a CP-net over three variables $\mathcal{V} = \{X_1, X_2, X_3\}$ with $\text{dom}(X_1) = \{x_1, \bar{x}_1\}$, $\text{dom}(X_2) = \{x_2, \bar{x}_2\}$, $\text{dom}(X_3) = \{x_3, \bar{x}_3\}$. Each variable is annotated with its CPT. For the variable X_1 , the user prefers x_1 to \bar{x}_1 unconditionally. For X_3 , the preference depends on the values of X_2 , i.e., $\text{Pa}(X_3) = \{X_2\}$. For instance, in the context of \bar{x}_2, \bar{x}_3 is preferred over x_3 . This CP-net is a complete CP-net since the user supplied a total order for every parent assignment of any $X_i \in \mathcal{V}$. However, if we removed any statement from $\text{CPT}(X_i)$ it becomes an incomplete one. In the binary case (i.e., $|\text{dom}(X_i)| = 2$) having a partial order on $\text{dom}(X_i)$ is equivalent to complete removal of \succ^i .

A CP-net N defined over \mathcal{V} induces a partial order \succ (sometimes we refer to it as \succ_N) on the set of possible

solutions \mathcal{S} . We also can describe the relation in \succ in terms of flips. A flip with respect to a solution s results in another new solution s' that is different from s in exactly one variable value X_i . In complete CP-nets, a flip from s to s' can be either an improving flip (the new value $s'[X_i]$ is preferred to the old value $s[X_i]$) or worsening flip (the new value $s'[X_i]$ is worse compared to the old one $s[X_i]$). However, for incomplete CP-nets, a flip can be neither improving or worsening as there could be no relation to judge the preference of $s[X_i]$ compared to $s'[X_i]$.

Formally, Let $u \in \text{dom}(\text{Pa}(X_i))$ be the parent assignments for a variable $X_i \in \mathcal{V}$. Let $\succ_u^i = x_1^i \succ \dots \succ x_m^i$ be the preference order of X_i where u is the context. Then, going from x_j^i to x_k^i is an improving flip with respect to X_i whenever $k < j \leq m$. Indeed, if \succ_u^i is a partial order, there could be no relation between x_j^i and x_k^i .

Example 4: In Figure 2, $(\bar{x}_1\bar{x}_2x_3, \bar{x}_1x_2x_3)$ with respect to the variable X_2 because $x_2 \succ \bar{x}_2$ according to $\text{CPT}(X_2)$. Furthermore, $(x_1x_2\bar{x}_3, x_1x_2x_3)$ is an improving flip with respect to X_3 as given $X_2 = x_2$, the user prefers x_3 to \bar{x}_3 .

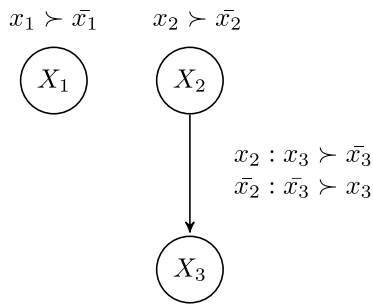


FIGURE 2. A CP-net.

Given a CP-net N and two solutions $s, s' \in \mathcal{S}$, a main challenge is to determine whether s dominates s' in the induced relation \succ (written as $s \succ s'$). To answer this in positive, we simply look for a sequence of improving flips $(\lambda_1, \lambda_2, \dots, \lambda_r)$ where $s' = \lambda_1$ and $s = \lambda_r$ and each $(\lambda_j, \lambda_{j+1})$ is an improving flip for all $j \in \{1, \dots, r - 1\}$. Therefore, s dominates s' if there exists a sequence of improving flips from s' to s otherwise s does not dominate s' , i.e., $s \not\succeq s'$. Note that $s \not\succeq s'$ gives no clue how s' stands with respect to s as both can be incomparable (there exists no sequence of improving flips from one to another). Incomparability simply means the presented information in N does not suffice to determine which one is better. In contrast to incomplete CP-nets, pairs of solutions (s, s') where s differs from s' in exactly one variable value are always comparable in complete CP-nets. The induced relation \succ can be captured by what is known as the induced graph of N .

Definition 7 (Induced Graph [15]): The induced graph for a CP-net N is defined as a directed graph G where the set of vertices is \mathcal{S} and an edge from s to s' exists iff (s', s) form an improving flip.

Example 5: Figure 3 shows the induced graph for the CP-net in Example 3. Each node in the graph represent a solution s where edges represent improving flips.

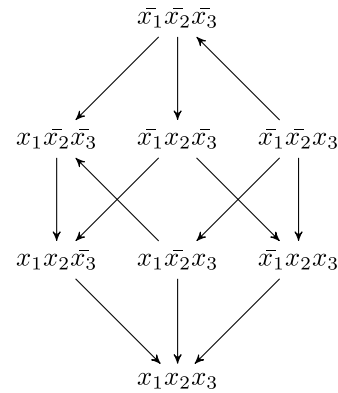


FIGURE 3. The full relation induced by the CP-net in Example 2.

The induced graph G is sufficient to answer any dominance query in the following sense: $s \succ s'$ if there is a path from s' to s in the transitive closure of G . Notice that N is exponentially smaller than G . This succinct representation of CP-net makes it appealing for domains with large number of attributes. The dominance task can be answered directly from N without constructing the induced graph and there exists efficient algorithms to tackle dominance tasks for restricted classes of CP-nets. The exact complexity of answering dominance queries depends on the CP-net structure and representation of its CPTs.

Another related task is to consistently order a subset of solutions $\mathcal{S}' \subseteq \mathcal{S}$ with respect to N . That is, obtain a total order \succ^* on \mathcal{S}' that is consistent with the information presented in N . Assume $X_1 \succ X_2 \succ \dots \succ X_n$ to be a topological order over \mathcal{V} that is consistent with N and assume we are interested in ordering two solutions s and s' that have same values for all variables X_1, \dots, X_r but they differ on the value of X_{r+1} . Assume also that, given the values of X_1, \dots, X_r in s and s' , $\text{CPT}(X_{r+1})$ shows $s[X_{r+1}] \succ s'[X_{r+1}]$ then s should be ordered before s' in \succ^* . Ordering task is a weak version of dominance. If $s \succ s'$ then $s' \not\succeq s$ and subsequently s is ordered before s' . However, if we know that s can be ordered before s' , this does not necessarily mean $s \succ s'$.

V. PROPOSED METHOD

We assume services are defined over a set of attributes $\mathcal{V} = \{X_1, \dots, X_n\}$ and a service is simply an element of $\mathcal{S} = \text{dom}(X_1) \times \text{dom}(X_2) \times \dots \times \text{dom}(X_n)$. In many real-world applications, however, it is expected that the customer has no complete knowledge in all attributes that define a particular service. Hence, it is often the case that preferences are defined on a subset of the attributes $\mathcal{V}' \subseteq \mathcal{V}$ that the customer is aware of or cares about. All other attributes $\mathcal{V} \setminus \mathcal{V}'$ can be viewed as irrelevant to the service selection strategy of that particular customer. This can be, for example, when some attributes preference relations require expertise that the current

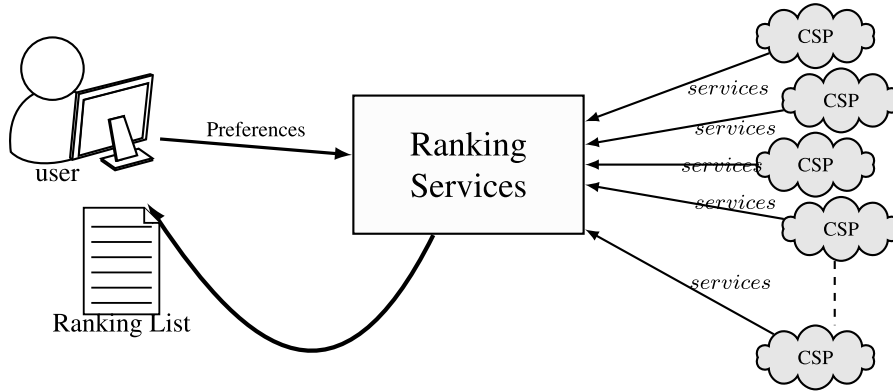


FIGURE 4. The customer first articulates the preferences on different attributes of interest. The available services are then ranked from most preferred to least preferred according to the customer’s CP-net. The ranked list is submitted back to choose the most suitable service meeting the customer needs.

customer does not have. As a result, the customer selects a subset of attributes \mathcal{V} that are of interest and is asked to rank a set of available services S' based on the provided preferences. For simplicity, we assume $S' = \text{dom}(\mathcal{V})$.²

In this section, we propose a CP-net based approach to capture criteria interdependencies in cloud service selection. We show how to i) choose the best service and ii) how to rank a set of available services based on the provided CP-net from most to least preferred service. Figure 4 outlines the approach. Initially, the customer is supplied with all variables \mathcal{V} . In principle, we assume \mathcal{V} is a common knowledge *i.e.*, can be viewed from the website of the cloud provider. Then, for each criteria X_i of the customer interest, the cloud customer is asked to identify a set of *parent* criteria $Pa(X_i)$ that influence the preference of X_i . For instance, assume the customer has the following statement “if the service has high availability rate then a provider with high reputation is preferred to medium one”. Then, the availability (A) is a parent of reputation (R). and the statement $h : h \succ m$ holds in $\text{CPT}(R)$. After articulating the preferences, a ranking over the available services is carried and the ranked list of services is submitted to the customer.

A. FINDING BEST SERVICE

For a complete CP-net N , finding the unique best service s^* is easy, all we need is to traverse N in a topological order while assigning for every X_i , the best value in \succ^i given its parents values. This is known as the sweep-forwarding algorithm in [15] which is also outlined in Algorithm 1. However, it is unlikely that the customer would be able to express a total preference relation for every $X_i \in \mathcal{V}$ for all $y \in \text{dom}(Pa(X_i))$. It is natural to miss or abstain from articulating preferences for some contexts y due to a gap in the knowledge. For incomplete CP-nets, it is no more guaranteed to have a unique best solution and there could be exponentially many non-dominated solutions.

²This is not a restriction as we can always construct the set $S' = \{s[\mathcal{V}'] \mid s \in \mathcal{S} \text{ and } s \text{ is provided to the customer}\}$

Algorithm 1 Sweep-Forward Procedure as Discussed in [15]

```

Input : A complete acyclic CP-net  $N$ 
Output: the best solution  $s^*$  in  $N$ 
1 Let  $\pi$  be a topological order consistent with  $N$ 
2  $s^* = \emptyset$ 
3 for  $i = 1$  to  $n$  do
4   Let  $x_1^i \succ x_2^i \succ \dots \succ x_m^i$  be the statement of
   CPT( $X_i$ ) given  $Pa(X_i)$ 
5    $s^* = s^* \cup x_1^i$ 
6 end
    
```

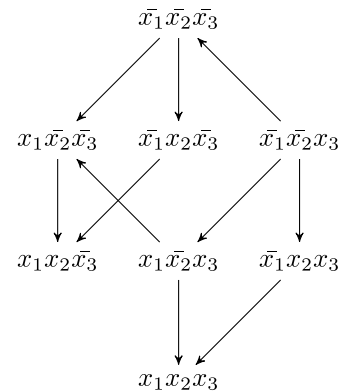


FIGURE 5. The full relation induced by the incomplete CP-net in Example 6.

Example 6: Consider the CP-net in Figure 2 where the statement $x_2 : x_3 \succ \bar{x}_3$ is removed from $\text{CPT}(X_3)$. The induced graph of this incomplete CP-net is shown in Figure 5. There are two non-dominated services or solutions, namely $x_1 x_2 \bar{x}_3$ and $x_1 x_2 x_3$. For all the others, they are dominated by one or more solutions.

However, it is possible to extend the sweep-forwarding algorithm to the case of incomplete ones as discussed in [15]. Simply, for every variable X_i where the statement \succ_y^i is

completely or partially missing: If \succ_y^i is completely missing (i.e., there is no preference relation on $dom(X_i)$ for the context y) choose an arbitrary element of $dom(X_i)$. Otherwise, choose any element of $dom(X_i)$ that is not dominated in \succ_y^i . This way, the final constructed solution is a non-dominated solution.

B. RANKING AVAILABLE SERVICES

A more interesting problem deals with ranking the available services S' from most preferred to least preferred according to the customer's CP-net.

Definition 8: A ranking \succ^* over a set S' is a consistent ranking w.r.t a CP-net N if the following holds: $\forall s, s' \in S'$ if $s \succ s'$ then $s \succ^* s'$

For complete CP-nets, Boutilier *et al.* [15] discussed how to construct a ranking \succ^* over S' . For the incomplete CP-nets, it is not obvious how to extract \succ^* from the CP-net. It turned out we can extend any incomplete CP-net to a complete one while preserving the dominance relations.

Definition 9: A complete CP-net N is an extension of an incomplete N' if N and N' share the exact same graph and, for every CPT(X_i) in N' showing a possible partial statement \succ_u^i , there exists a total order statement \succ_u^i in the CPT(X_i) of N that agrees with it.

Essentially, an extension for the incomplete CP-net N' is simply "filling the gaps" in its CPTs. For example, the complete CP-net in Figure 2 is an extension of the incomplete CP-net in Example 6.

Proposition 1: For any complete CP-net N that extends an incomplete one N' , if $s \succ_{N'} s'$ then $s \succ_N s'$.

Proof: The proof is by contradiction. Assume $s \succ_{N'} s'$ and $s \not\succ_N s'$ hold for any two solutions s and s' . As $s \succ_{N'} s'$ then there is a sequence of improving flips $s' = \lambda_1, \dots, \lambda_k = s$ in N' but not in N . This means there exists at least one variable X_i where two of its values have been flipped (say from x_i to x'_i) because of the statement \succ_u^i in CPT(X_i). However, there was no improving flip from x_i to x'_i in N . This means $x_i \succ_u^i x'_i$ in N' but $x_i \not\succ_u^i x'_i$ in N which contradicts with N being an extension of N' . \square

The above result states that it is always possible to rank a set of services according to any extension of the incomplete CP-net N . The ranking \succ^* would indeed be a consistent ranking with the information expressed in the network. Obviously, there always exists at least one extension for any incomplete CP-net.

VI. EXPERIMENT STUDY

To test our proposed method, we randomly generate a number of customer's CP-nets with varying size, $n = 10 \sim 200$, and multiple in-degree bounds, $c = 0, 1, 2$. This bound on the indegree (i.e., the number of parents for a given attribute) is not a limitation as the class of all directed acyclic graphs with indegree at most 2 is still expressive to represent complex interactions within criteria. When the indegree is zero or one, we end up with a special class of CP-nets known as separable ($c = 0$) or trees ($c = 1$). Separable CP-nets

express situations when there is no interdependencies and every attribute is independent from all other attributes, while tree CP-nets assert that there exists at most one parent for any given attribute in the network. Such two classes were shown to be of practical interest and it is conjectured that humans tend to base their preferences based on a small number of parents. For simplicity, the attributes are assumed to be binary (i.e., has two possible values).

For the offered cloud services³, we consider that there are $2n$ candidate services, e.g., for CP-net of size $n = 10$ there is 20 candidate services to choose from or rank. Then we apply the two algorithms presented above. Each experiment (i.e., combination of n , c , and an algorithm) is repeated 20 times and results are averaged accordingly. The experiments were conducted using basic built-in Matlab functions, running in a 2015 MacBook Pro with 2.5 GHz Intel Core i7 and 16 GB DDR3 RAM.

Figures 6 and 7 plot the average time to find the best service and to rank all services, respectively, in CP-nets with varying size and using the different in-degree bounds. The figures show that our approach is promising in terms of scalability. It can handle large set of attributes and return solutions within a fraction of the second. Also the figures show a time difference between different in-degree bounds as more complex structures (i.e., more conditional statements given by the user) would require longer time to reach a solution even when compared to a smaller CP-net (e.g., the case of $n = 170$ with $c = 0$ took less average time $n = 140$ with $c = 2$).

While our approach is purely qualitative one, in many real-world applications, a numerical value expressing criteria importance is needed. This is the case, for example, when adopting a fuzzy decision support system that assumes importance weights on criteria are given. Recent Advancements in CP-nets allow us to quantify the importance of every attribute in the domain given complex criteria interactions [4], [24].

The CP-net structure, inherently, encapsulates variable importance within the network, which can be used to assess the contribution of these variables in assessing a given solution. The basic idea of computing the weight of importance is by considering the node location in the network. Initially, all nodes have equal weights of one, then starting from leaf nodes (the bottom of the graph) and traversing the nodes in a reverse topological order, the weight of each parent is increased by an amount equal to the sum of the weights of its children [24]. For example, in Emma's CP-net, the weight of the bottom leaf node, $w(Rec)$ is 1, then its parent has a weight $w(C) = 2$, and lastly both $w(Rep) = w(A) = 3$. Emma did not need to estimate or express the weight of importance but rather conditional preference statements.

Figure 8 plots the computation time of the weights for the generated CP-nets and for the three in-degree bounds.

³We use cloud service instead of cloud service provider as a single provider may offer multiple services with different configurations and attributes.

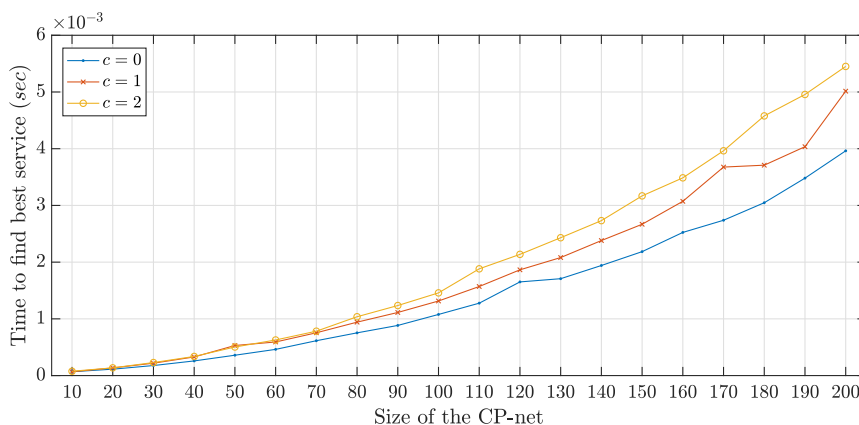


FIGURE 6. Average time to find the best service in CP-nets with varying size.

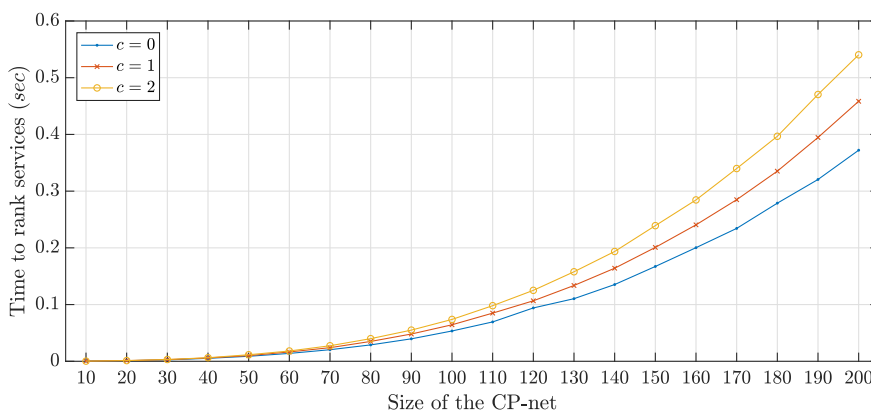


FIGURE 7. Average time to rank all candidate services in CP-nets with varying size.

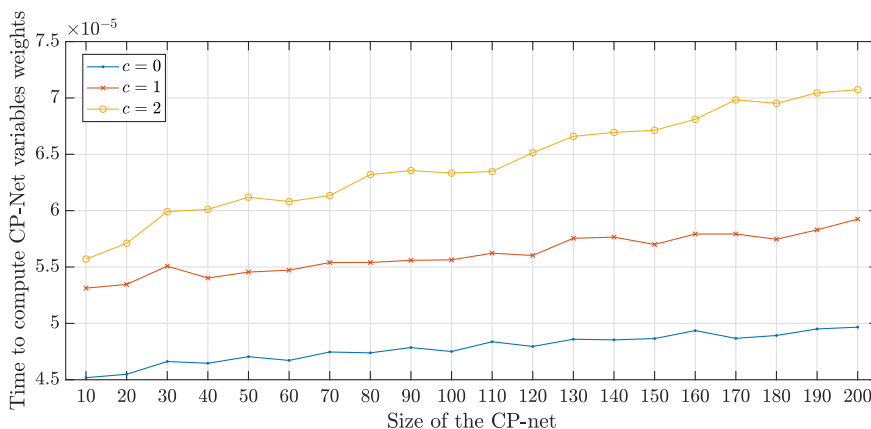


FIGURE 8. Time needed to compute attributes importance weight for different size of CP-nets.

The results indicate linear computation time increase as the size of the CP-nets (*i.e.*, number of attributes) increases. Also as expected, the larger in-degree bound, c , the more complex the network and hence more computational time is incurred. Overall, CP-nets exhibits scalable and efficient performance in modeling user attributes and selecting and ranking solutions.

VII. CONCLUSION AND FUTURE WORK

We have showed that CP-net, a prominent preference model in the AI community, is an adequate model for customer preferences in cloud service selection and ranking. We further demonstrated even in the case of incomplete CP-nets, it is still possible to reason about best available service and rank an arbitrary set of services. Our experimental results prove

that CP-nets, and preferential dependencies in general, are indeed valuable tools to capture preferences in domains with large number of attributes. Indeed, our experimental results demonstrated it is possible to compute attributes' importance weights in efficient way from a purely comparative statements and complex interactions as represented by the CP-net graph.

In the near future, we plan to extend this work into learning relevant and irrelevant attributes in an interactive manner, using preference languages that are more expressive than CP-nets, and test the usage of CP-net from real world data.

ACKNOWLEDGMENT

The authors would like to acknowledge the technical support of the University of Jeddah.

REFERENCES

- [1] L. Wang, G. Von Laszewski, A. Younge, X. He, M. Kunze, J. Tao, and C. Fu, "Cloud computing: A perspective study," *New Generat. Comput.*, vol. 28, no. 2, pp. 137–146, 2010.
- [2] S. M. M. Fattah, A. Bouguettaya, and S. Mistry, "A CP-Net based qualitative composition approach for an iaas provider," in *Proc. Int. Conf. Web Inf. Syst. Eng.* Springer, 2018, pp. 151–166.
- [3] A. Al-Faifi, B. Song, M. M. Hassan, A. Alamri, and A. Gumaei, "A hybrid multi criteria decision method for cloud service selection from smart data," *Future Gener. Comput. Syst.*, vol. 93, pp. 43–57, Apr. 2019.
- [4] K. Laing, P. A. Thwaites, and J. P. Gosling, "Rank pruning for dominance queries in CP-nets," *J. Artif. Intell. Res.*, vol. 64, pp. 55–107, Jan. 2019.
- [5] S. Mistry, A. Bouguettaya, H. Dong, and A. Erradi, "Qualitative economic model for long-term iaas composition," in *Proc. Int. Conf. Service-Oriented Comput.* Springer, 2016, pp. 317–332.
- [6] S. Mistry, A. Bouguettaya, H. Dong, and A. Erradi, "Probabilistic qualitative preference matching in long-term iaas composition," in *Proc. Int. Conf. Service-Oriented Comput.* Springer, 2017, pp. 256–271.
- [7] M. Sun, T. Zang, X. Xu, and R. Wang, "Consumer-centered cloud services selection using AHP," in *Proc. Int. Conf. Service Sci. (ICSS)*, Apr. 2013, pp. 1–6.
- [8] Z. U. Rehman, F. K. Hussain, and O. K. Hussain, "Towards multi-criteria cloud service selection," in *Proc. 5th Int. Conf. Innov. Mobile Internet Services Ubiquitous Comput.*, Jun. 2011, pp. 44–48.
- [9] L. Sun, H. Dong, O. K. Hussain, F. K. Hussain, and A. X. Liu, "A framework of cloud service selection with criteria interactions," *Future Gener. Comput. Syst.*, vol. 94, pp. 749–764, May 2019.
- [10] L. Sun, H. Dong, F. K. Hussain, O. K. Hussain, and E. Chang, "Cloud service selection: State-of-the-art and future research directions," *J. Netw. Comput. Appl.*, vol. 45, pp. 134–150, Oct. 2014.
- [11] R. L. Keeney and H. Raiffa, *Decisions With Multiple Objectives: Preferences and Value Trade-Offs*. Cambridge, U.K.: Cambridge Univ. Press, 1993.
- [12] M. Bichler and H. Werthner, "A classification framework of multidimensional, multi-unit procurement negotiations," in *Proc. 11th Int. Workshop Database Expert Syst. Appl.*, 2000, pp. 1003–1009.
- [13] J. Lang and L. Xia, "Sequential composition of voting rules in multi-issue domains," *Math. Social Sci.*, vol. 57, no. 3, pp. 304–324, May 2009.
- [14] A. Permalu, K. Rantasila, M. Porthin, V. Hinkka, J. Eckhardt, and J. Leonardi, "Multi-criteria evaluation method for freight logistics innovations," *IET Intell. Transp. Syst.*, vol. 9, no. 6, pp. 662–669, Aug. 2015.
- [15] C. Boutilier, R. I. Brafman, C. Domshlak, H. H. Hoos, and D. Poole, "CP-Nets: A tool for representing and reasoning with Conditional ceteris paribus preference statements," *J. Artif. Intell. Res.*, vol. 21, pp. 135–191, Feb. 2004.
- [16] S. Mistry, A. Bouguettaya, and H. Dong, "Long-term qualitative IaaS composition," in *Economic Models for Managing Cloud Services*. Springer, 2018, pp. 77–110.
- [17] A. V. Dastjerdi and R. Buyya, "A taxonomy of QoS management and service selection methodologies for cloud computing," *Cloud Comput., Methodol., Syst., Appl.*, vol. 15, pp. 16–76, Dec. 2011.
- [18] R. R. Kumar and C. Kumar, "A multi criteria decision making method for cloud service selection and ranking," *Int. J. Ambient Comput. Intell.*, vol. 9, no. 3, pp. 1–14, Jul. 2018.
- [19] M. Whaiduzzaman, A. Gani, N. B. Anuar, M. Shiraz, M. N. Haque, and I. T. Haque, "Cloud service selection using multicriteria decision analysis," *Sci. World J.*, vol. 2014, pp. 1–10, Feb. 2014.
- [20] Z. U. Rehman, O. K. Hussain, and F. K. Hussain, "IaaS cloud selection using MCDM methods," in *Proc. IEEE 9th Int. Conf. e-Business Eng.*, Sep. 2012, pp. 246–251.
- [21] B. D. Chung and S. Kwang-Kyu, "A cloud service selection model based on analytic network process," *Indian J. Sci. Technol.*, vol. 8, no. 18, Aug. 2015, Art. no. IPL0186.
- [22] H. Wang, J. Zhang, W. Sun, H. Song, G. Guo, and X. Zhou, "WCP-Nets: A weighted extension to CP-nets for web service selection," in *Proc. Int. Conf. Service-Oriented Comput.* Springer, 2012, pp. 298–312.
- [23] A. S. Alashaikh and E. A. Alanazi, "Incorporating ceteris paribus preferences in multiobjective virtual machine placement," *IEEE Access*, vol. 7, pp. 59984–59998, 2019.
- [24] M. Li, Q. B. Vo, and R. Kowalczyk, "Efficient heuristic approach to dominance testing in CP-nets," in *Proc. 10th Int. Conf. Auto. Agents Multiagent Syst.*, vol. 1, 2011, pp. 353–360.



ABDULAZIZ ALASHAIKH received the B.Sc. degree in electrical and computer engineering from King Abdulaziz University, in 2004, and the M.S. degree in telecommunications and networking and the Ph.D. degree in information science from the University of Pittsburgh, in 2011 and 2017, respectively. He is currently an Assistant Professor with the Department of Computer and Network Engineering, University of Jeddah. His research interests include multi-layer network design, survivability and resilience in networks, and multiobjective optimization.



EISA ALANAZI received the B.Sc. degree in information systems from King Saud University, in 2007, and the M.Sc. and Ph.D. degrees from the University of Regina, Canada, in 2011 and 2017, respectively. He is currently an Associate Professor with the Department of Computer Science, College of Computers and Information Systems, Umm Al-Qura University, Saudi Arabia. His research interests include preference learning and reasoning.