Temporal Preference Models and their Deduction-based Analysis for Pervasive Applications

Radosław Klimek
AGH University of Science and Technology, al. A. Mickiewicza 30, 30-059 Krakow, Poland

Keywords: Preference Modeling, Context-aware Applications, Patterns, Temporal Logic, Deduction, Semantic Tableaux.

Abstract: This work concerns preference models and their formal analysis using a deductive approach, i.e. temporal logic for both specification and verification, and the semantic tableaux method for reasoning. The architecture of an automatic and deduction-based verification system for preference models is also proposed. It allows analysis of both desired properties of models and their semantic contradictions. Preference models are built from predefined patterns which enable automatic generation of logical specifications for preferences.

1 INTRODUCTION

Preference modeling enables customization of software behavior to a user's needs. Preference models are particularly important in *pervasive computing* and *ubiquitous computing* which are paradigms related to the behavior of software highly oriented towards users and their needs, assuming also the omnipresence of computing. Preference modeling is essential and crucial, and constitutes a type of bridge between a support-oriented user and a system which is itself able to provide support.

The model of preference might be constructed using fuzzy sets, classical logic and multi-valued logic. Classical logic, and particulary rule-based systems, are especially popular while non-classical logic and especially temporal logic, are rather less popular. However, temporal logic is a well established formalism for describing reactivity. At the same time, a typical pervasive application should be characterized by reactivity and flexibility in adapting to preference changes on the user's side. After building a preference model in temporal logic, one can analyze it using a deductive approach. The goal is to search, if possible, for contradictions in a model or to infer something about the correctness of the preference objectives. Temporal logic enables illustration the dynamic aspect of preferences over flows of time.

Motivations and Contributions

The general motivation is the lack of satisfactory and documented results of using temporal logic and the deduction-based approach for formal analysis of preference models. Another motivation factor is the lack of tools for automatic extraction of logical specifications for preference models.

The main contributions are the automation of the generation process of logical specifications for preference models, predefined preference patterns, and generation algorithm for preference patterns. Another contribution is the use of a non-standard method for deduction for preferences.

Related Works

Instead of discussion of related works, beyond the fundamental work by Öztürk et al. (Öztürk et al., 2005), let us pay attention on work by Fong et al. (Fong et al., 2011). It includes an excellent section on the state of the art of preference modeling. It is a highly in-deep and comprehensive review. This work is an attempt to extend the view of problems of preference modeling expressed in (Fong et al., 2011) by introducing temporal logic and the semantic tableaux method.

2 MODEL OF PREFERENCES

Preference models are discussed below. The proposed preference model is based on predefined patterns of rules which are shown in Fig. 1. These rules are expressed in temporal logic. Temporal logic and the deductive system are discussed in section 3. However, it is assumed that well-formed and syntactically correct

Klimek R..
Temporal Preference Models and their Deduction-based Analysis for Pervasive Applications.
DOI: 10.5220/0004314901310134
In *Proceedings of the 3rd International Conference on Pervasive Embedded Computing and Communication Systems* (PECCS-2013), pages 131-134
ISBN: 978-989-8565-43-3

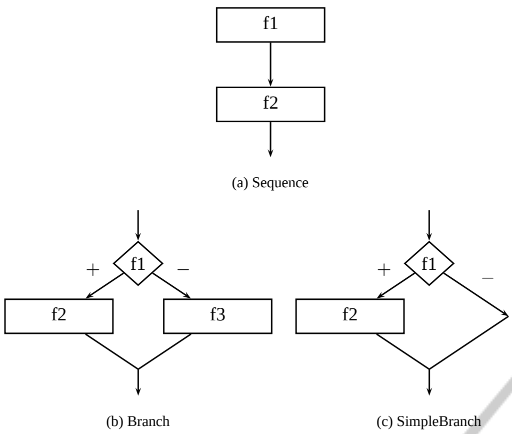

Figure 1: Patterns for preferences.

temporal logic formulas have already been defined, c.f. (Wolter and Wooldridge, 2011).

Preference models consist of some basic patterns. A *pattern* is a predefined solution for a special context where there are preference issues. They are generally indicated as $pat()$, where *pat* is a name of a given pattern, and their parameters are included in parentheses. The following three patterns are considered: *Branch*, *SimpleBranch* and *Sequence*. It is a kind of illustration of the if-then scheme. Patterns of behaviors and preferences can be nested. It follows from the scenario of multi-stage decision-making. A *basic set of patterns* Σ is a set of temporal logic formulas describing both liveliness and safety properties of a pattern. A set of three patterns, i.e. $\Sigma = \{Branch, SimpleBranch, Sequence\}$, will be considered. Let us define temporal properties for these patterns. Hence, set $Branch(f_1, f_2, f_3) = \{c(f_1) \Rightarrow \Diamond f_2 \wedge \neg\Diamond f_3, \neg c(f_1) \Rightarrow \neg\Diamond f_2 \wedge \Diamond f_3, \Box\neg(f_1 \wedge (f_2 \vee f_3))\}$ describes property of the Branch pattern and $SimpleBranch(f_1, f_2) = \{c(f_1) \Rightarrow \Diamond f_2, \neg c(f_1) \Rightarrow \neg\Diamond f_2, \Box\neg(f_1 \wedge f_2)\}$ the SimpleBranch pattern. Set $Sequence(f_1, f_2) = \{f_1 \Rightarrow \Diamond f_2, \Box\neg(f_1 \wedge f_2)\}$ defines the Sequence pattern. For the meanings of these patterns, refer to Fig. 1. Formulas f_1, f_2 etc. are atomic formulas for a pattern, and constitute some formal arguments for these pattern. $\Diamond f$ means that sometime (or eventually in the future) activity f is completed. In addition, $c(f)$ means that the logical condition associated with activity f has been evaluated and is satisfied.

The entire preference model can be written in the form of logical expressions, which is similar to some well-known regular expression. The goal is to write preference models in a concise and literal notation. A *logical expression* W_L is a structure created using the following rules:

- every elementary set $pat(a_i)$, where $i > 0$ and every a_i is an atomic formula, is a logical expression,
- every $pat(A_i)$, where $i > 0$ and every A_i is either
  - an atomic formula a_j, where $j > 0$, or
  - a set $pat(a_j)$, where $j > 0$ and a_j is an atomic formula, or
  - a logical expression $pat(A_j)$, where $j > 0$

  is also a logical expression.

Any logical expression may represent an arbitrary structure of patterns and examples of this are expression $Branch(a, SimpleBranch(f, g), c)$ and expression $Sequence(Branch(a, b, c), SimpleBranch(d, e))$. In the first case, the combination (and nesting) of two branchings is considered, i.e. the ordinary and the simple one. In this expression, a and f are the conditions. In the second case, the sequence of two branchings is considered.

An individual preference may belong to a set of preferences P, i.e. $P = \{p_1, p_2, ...p_n\}$, where p_i is a preference which is expressed as a single logical expression, i.e. $p_i = W_{Li}$.

3 DEDUCTION SYSTEM

The logical background is discussed further below. The important argument for a deductive approach is how natural it is for human beings and it is used commonly and intuitively in everyday life. *Temporal Logic* TL is a valuable formalism, e.g. (Wolter and Wooldridge, 2011), which has strong applications for the specification and verification of models. It exists in many variations, however, considerations in this paper are limited to the *Linear Temporal Logic* LTL, i.e. logic for which the time structure is considered as linear. Furthermore, considerations are limited to the *smallest*, or *minimal*, *temporal logic*, e.g. (Chellas, 1980), also known as temporal logic of class K. The following formulas may be considered as significant examples of minimal temporal logic: $quest \Rightarrow \Diamond answer$, $\Box(action \Rightarrow \Diamond react)$, $\Diamond liv$, $\Box\neg(bad)$ or $\Box\neg(ev1 \wedge (ev2 \vee ev3))$, etc.

Semantic tableaux is a decision-making procedure for checking satisfiability of a formula. The method is well known in classical logic but it can also be applied in modal and temporal logics (d'Agostino et al., 1999). The method is based on formula decompositions. At the end of the decomposition procedure, all branches of the received tree are searched for contradictions. When all branches of the tree have contradictions, it means that the inference tree is *closed*. If

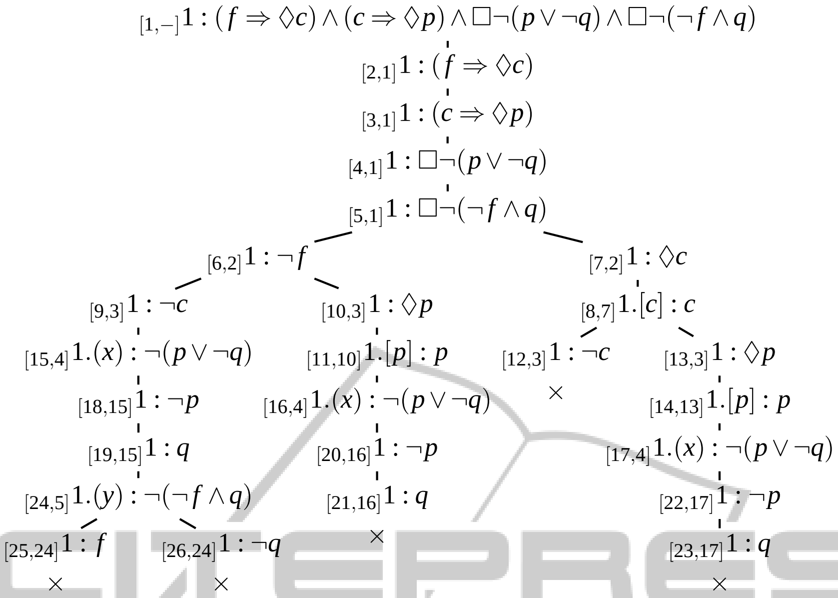

Figure 2: Truth tree of the inference process.

the negation of the initial formula is placed in the root, this leads to the statement that the initial formula is true. This method has some advantages over the traditional axiomatic approach. In the classical reasoning approach, starting from axioms, longer and more complicated formulas are generated and derived. Formulas become longer and longer step by step, and only one of them will lead to the verified formula. The method of semantic tableaux is characterized by the reverse strategy. The method provides, through so-called *open* branches of the semantic tree, information about the source of an error, if one is found, which is another and very important advantage of the method.

Let us consider the following example. The preference model is constructed from a natural text: *If he buys a Ferrari then he sometime smokes a Cuban cigar. If he smokes a Cuban cigar then he sometime drinks a Dom Perignon champagne. However, it is never so that he drinks a Dom Perignon champagne or he does not quit drinking alcohol. Moreover, it is not so that he does not buy a Ferrari and quit drinking alcohol.* The following simple sentences are extracted: “to buy a Ferrari” – f, ‘ ‘to smoke a Cuban cigar” – c, “to drink a Dom Perignon champagne” – p, and “to quit drinking alcohol” – q. Then, the following formulas of temporal logic are extracted: $(f \Rightarrow \Diamond c)$ and $(c \Rightarrow \Diamond p)$ and $\Box\neg(p \vee \neg q)$ and $\Box\neg(\neg f \wedge q)$. These formulas express the preference model. The first two formulas express the liveness aspect of the model, and the last two formulas express the safety aspect of the model. The preference model is analyzed using the method of sematic tableaux, c.f. Fig. 2. All branches contain contradictions ($\times$). This means there is no valuation for extracted propositions that satisfies the formula which is placed in the tree root. The preference model is semantically contradictory.

4 TOWARDS AUTOMATION

Building a logical model for preferences in the form of temporal logic formulas, i.e. $P = \{p_1, p_2, ...p_n\}$, enables examination two important aspects of the system:

1. semantic contradiction, or
2. correctness of the model due to some properties.

Analysis of semantic contradiction is shown in the previous section. In turn, formal verification of properties of the preference model leads to the analysis of formula $p_1 \wedge \ldots \wedge p_n \Rightarrow Q$, where Q is a desired property of the preference model $\{p_1, p_2, ...p_n\}$.

The system of automatic inference on preference models is proposed in Fig. 3. The Modeler module allows (instead of natural language) the preparation a preference model using preference patterns shown in Fig. 1. The output of the Modeler is preference models expressed as logical expressions W_L.

The next module is the Generator module and its inputs are logical expressions W_L and predefined set of preference patterns Σ. The output is a logical specification L understood as being a set of temporal logic

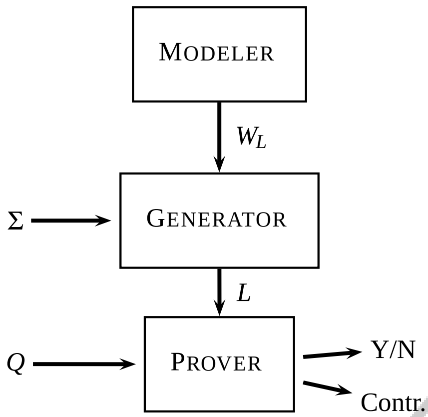

Figure 3: Deduction system.

formulas. The sketch of the generating algorithm is the following:

1. at the beginning, the logical specification is empty, i.e. $L = \emptyset$;
2. the most nested pattern or patterns are processed first, then the least nested patterns are processed one by one, i.e. patterns that are located more towards the outside;
3. if the currently analyzed pattern consists only of atomic formulas, the logical specification is extended, by summing sets and by formulas linked to the type of the analyzed pattern $pat()$, i.e. $L = L \cup pat()$;
4. if any argument is a pattern itself, then the logical disjunction of all its arguments, including nested arguments, is substituted in place of the pattern.

The above algorithm refers to similar ideas in work (Klimek, 2012). Let us supplement the algorithm with some examples. The example for the step 3: $Seq(p,q)$, gives $L = \{p \Rightarrow \Diamond q, \Box\neg(p \wedge q)\}$ and $Branch(a,b,c)$ gives $L = \{c(a) \Rightarrow \Diamond b \wedge \neg\Diamond c, \neg c(a) \Rightarrow \neg\Diamond b \wedge \Diamond c, \Box\neg(a \wedge (b \vee c))\}$. The example for the step 4: $Sequence(Branch(a,b,c),d)$ leads to $L = \{c(a) \Rightarrow \Diamond b \wedge \neg\Diamond c, \neg c(a) \Rightarrow \neg\Diamond b \wedge \Diamond c, \Box\neg(a \wedge (b \vee c)), (a \vee b \vee c) \Rightarrow \Diamond d, \Box\neg((a \vee b \vee c) \wedge d)\}$.

The last module is the Prover module that works using the semantic tableaux method. The inputs for the Prover are a logical specification L and a query Q which is a simple temporal logic formula which expresses the desired property of the preference model. (This formula can be prepared using a simple text editor.) The Prover provides examination for two cases:

1. correctness of the model due to some properties, i.e. the formal verification of the formula:

$$p_1 \wedge \ldots \wedge p_n \Rightarrow Q \quad (1)$$

or

2. semantic contradiction, i.e. the formal analysis of the formula:

$$p_1 \wedge \ldots \wedge p_n \quad (2)$$

In the case of correctness, the negation of the formula 1 is placed in the root and the Yes/No output is produced. In the case of contradiction, the formula 2 is placed in the root and the information about the semantic contradiction is produced.

5 CONCLUSIONS

The work presents a new approach to the formal analysis of preference models using temporal logic and the semantic tableaux method. Future work may include the implementation of the logical specification generation module and a deduction engine. The approach should results in a CASE software providing modeling preferences.

ACKNOWLEDGEMENTS

This work was supported by the AGH UST internal grant no. 11.11.120.859.

REFERENCES

Chellas, B. F. (1980). *Modal Logic*. Cambridge University Press.

d'Agostino, M., Gabbay, D., Hähnle, R., and Posegga, J. (1999). *Handbook of Tableau Methods*. Kluwer Academic Publishers.

Fong, J., Indulska, J., and Robinson, R. (2011). A preference modelling approach to support intelligibility in pervasive applications. In *8th IEEE Workshop on Context Modeling and Reasoning (CoMoRea 2011), Seattle, USA, 21–25 March 2011*, pages 409–414. IEEE.

Klimek, R. (2012). Proposal to improve the requirements process through formal verification using deductive approach. In Filipe, J. and Maciaszek, L., editors, *Proceedings of 7th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2012), 29-30 June, 2012, Wrocław, Poland [extended version is to be published in Springer–Verlag]*, pages 105–114. SciTePress.

Öztürk, M., Tsoukiàs, A., and Vincke, P. (2005). Preference modelling. In Figueira, J., Greco, S., and Ehrgott, M., editors, *Multiple Criteria Decision Analysis: State of the Art Surveys*, pages 27–72. Springer Verlag, Boston, Dordrecht, London.

Wolter, F. and Wooldridge, M. (2011). Temporal and dynamic logic. *Journal of Indian Council of Philosophical Research*, XXVII(1):249–276.