Letter to the Editor

Pattern-Based Analysis of Computer-Interpretable Guidelines: Don't Forget the Context

In the paper by Grando et al. [1], the authors use formal methods to prove whether a computer-interpretable guideline modeling language satisfies a set of control-flow workflow patterns [2]. They contrast the formal proof with an earlier informal analysis [3] in which we had compared different guideline modeling languages according to workflow patterns that they supported. Grando et al. find differences between their analysis and our analysis on the support of certain patterns by the PROforma guideline modeling language and conclude that "As tools for comparing languages these formal techniques make misinterpretation of the semantics of target patterns less likely, add much-needed rigor and give greater confidence in the results, but they cannot guarantee that all language features have been fully understood by the investigator".

However, while formal proof certainly could provide additional rigor, naturally, it still depends on correct understanding of the semantics of the modeling language and of the workflow patterns. When errors in understanding of these semantics occur, then formal methods are no better than informal ones. In fact, the formal proof strategy presented in Grando's paper includes an initial informal step in which the modeler tries to specify a workflow W in language L, which could potentially provide the behavior described by pattern P. If the modeler finds such workflow, then the formal proof continues from it; if he does not, then he exhaustively tries all possible combination of components from L that can provide the behavior described by pattern P to conclude that the pattern in not satisfied. This informal step was exactly the strategy that we had used in our study [3]. However, the quality of this informal proof step depends on the correct understanding of the modeler regarding the semantics of the guideline modeling language and of the workflow patterns. Clearly Grando et al. are experts in PROforma semantics, which allowed them in an earlier paper [4] to find an error in our analysis, where we did not realize a way for PROforma to support pattern 10 (Arbitrary Cycles). But, Grando et al. misunderstood the context condition for pattern 5 (Simple Merge), concluding erroneously that our analysis was incorrect and that PROforma does not satisfy pattern 5, where in fact it does. Specifically, this context condition, which is an assumption of the pattern, states that the place at which the merge occurs (i.e., place p1 in figures 4-a) is safe and can never contain more than one token (see http://www.workflowpatterns.com/patterns/control/basic/wcp5.php). The exclusive Simple Merge assumes that it is preceded by Exclusive Choice, therefore it is always exclusive. The pattern should not be initialized with one token in place I1 and one token in place I2, as done in the analysis of Grando et al; instead the pattern should be initialized with a token either in place I1 or in place I2 (and not both). In Figure 4-a, only one or another of the exclusive paths may execute (AC or BC). The occurrence of a situation, where task A as well as task B can both be enabled and the execution of each of which is followed by execution of task C, resulting in traces ACBC or BCAC, assumes that tasks A and B are preceded by a pattern distinct from Exclusive Choice (according to the Exclusive Choice pattern either task A or task B could have initially be enabled). As the Simple Merge pattern, by definition, is always preceded

by the Exclusive Choice pattern, it cannot result in the traces ACBC or BCAC. In order for tasks A and B execute as ACBC or BCAC there must be an environment which cannot guarantee safe conditions (so multiple tokens may reside in one place), therefore the Multi-merge pattern (pattern 8) (which assumes a weaker context condition) needs to be used. It would allow for ACBC and BCAC if both B and C are activated.

The above discussion shows the importance of using *explicit context conditions* (see also Section 3.2 of [5]). Moreover, we would like to stress that the workflow patterns were developed not just to analyze the formal expressiveness of a language. Most languages are Turing complete and can therefore model any behavior. Any programming language is Turing complete, but this does not make it a good language for modeling clinical computer-interpretable guidelines. Therefore, the patterns focus on the notion of suitability rather than expressiveness [2, 5]. The existence of some workaround to realize a pattern does not imply that the pattern is supported well by the language. Unfortunately, the notion of suitability is difficult to formalize. Therefore, [5] lists explicit evaluation criteria per pattern.

References

[1] M.A. Grando, Glasspool D., Fox J. A formal approach to the analysis of clinical computer-interpretable guideline modeling languages. Artificial Intelligence in Medicine 2011.

[2] W.M.P. van der Aalst, ter Hofstede A.H.M., Kiepuszewski B., Barros A.P. Workflow Patterns. Distributed and Parallel Databases 2003;14(3):5-51.

[3] N. Mulyar, van der Aalst W.M.P., Peleg M. A Pattern-based Analysis of Clinical Computer-Interpretable Guideline Modeling Languages. J Am Med Inform Assoc 2007;14(6):781-7.

[4] M.A. Grando, Glasspool D., Fox J. Petri Nets as a Formalism for Comparing Expressiveness of Workflow-Based Clinical Guideline Languages. In: Danilo Ardagna MM, Jian Tang, editor. Business Process Management Workshops, BPM 2008 International Workshops. Milano, Italy: Springer; 2008. p. 348-360.

[5] N. Russell, ter Hofstede A.H.M., van der Aalst W.M.P., Mulyar N. Workflow Control-Flow Patterns: A Revised View; 2006. Report No.: BPM Center Report BPM-06-22.

Mor Peleg*
Department of Information Systems,
University of Haifa, 31905 Haifa, Israel

Nataliya Mulyar
Rijkswaterstaat, Department of Infrastructure,
P.O. Box 20.000, NL-3502 LA, Utrecht, The Netherlands

Wil M.P. van der Aalst
Architecture of Information Systems,
Eindhoven University of Technology,
P.O. Box 513, NL-5600 MB, Eindhoven, The Netherlands

*Corresponding author. Tel.: +972-4-828-8509;
fax: +972-4-828-8522.
E-mail address:morpeleg@is.haifa.ac.il