Hybrid Process Modeling and Mining

Anti Alman[1]

[1]University of Tartu, Tartu, Estonia

Abstract

One of the cornerstones of business process management and process mining is the ability to describe any business process systematically as a process model. The notations used for these descriptions can, in general, be divided into two paradigms: procedural and declarative. The procedural paradigm describes processes in terms of explicit process control flow, whereas the declarative paradigm describes processes in terms of process rules while leaving the process control flow implicit. This difference has significant implications for describing knowledge-intensive processes. A knowledge-intensive process may have thousands of case variants, all of which may correspond to valid process executions. While procedural models are generally considered more understandable, describing a knowledge-intensive process using the procedural paradigm often results in an unreadable spaghetti model. The declarative paradigm avoids this issue by leaving the control flow of the process implicit. However, this has a tendency to hide the overall structure of the process, and therefore again leading to understandability issues. During this project we plan to address these issues by first analysing the existing notations suitable for process modeling, and then using the results of this analysis for developing a novel hybrid process modeling approach and a set of corresponding process mining techniques.

Keywords

Process modeling, Process mining, Knowledge-intensive processes, Hybrid modeling

1. Introduction and Motivation

In the field of Business Process Management (BPM) there are two types of languages for describing a process model [1]. The first type is procedural languages (sometimes referred to as imperative languages), which aim to describe end-to-end processes and allow only for activities that are explicitly triggered through control-flow. Some examples of procedural languages are BPMN [2], Petri Nets [3] and YAWL [4]. The second type is declarative languages, which aim to describe the process as a set of rules that the process should follow, while allowing for any control flow that does not conflict with these rules. Some examples of declarative languages are Declare [5], DCR Graphs [6] and OCBC models [7].

Both procedural and declarative languages have their strengths and weaknesses [8, 9, 10]. Procedural languages are considered to be especially suitable for predictable processes that have very few deviations from the main flow of the process (sometimes referred to as strict or structures processes). Conversely, the declarative approach is considered to be especially suitable for less predictable processes where many deviations from the main flow of the process

Proceedings of the Demonstration & Resources Track, Best BPM Dissertation Award, and Doctoral Consortium at BPM 2021 co-located with the 19th International Conference on Business Process Management, BPM 2021, Rome, Italy, September 6–10, 2021

anti.alman@ut.ee (A. Alman)

are possible (sometimes referred to as flexible or unstructured processes). An example of the latter can be seen in knowledge-intensive processes [11].

This doctoral project aims to develop a hybrid approach for process modeling and mining that would combine the strengths while also addressing the weaknesses of both procedural and declarative languages. More specifically, the current direction of research is to use aspects of the procedural languages to represent the overall structure of the process and to use aspects of the declarative languages to represent parts or rules of the process where the order of activities is less relevant than the business rules associated with these activities.

2. Related Work

Developing a hybrid process modeling and mining approach has been previously suggested and there are indications that practitioners would be open to such an approach [10]. There has also been some research in that direction both in the context of process modeling and process mining [12]. Some examples are:

- BPMN-D [13], which is a conservative extension of the standard Business Process Model and Notation (BPMN) for declarative process modeling.
- Pockets of flexibility [14, 15], which is a concept for handling flexibility in workflow specifications.
- Hybrid Miner [16], which discovers a hybrid model where each subprocess is either a Declare model or a Petri net.
- Fusion Miner [17], which first uses Heuristics Miner to discover a procedural process model and then Declare Miner to refine the model by replacing some of the flow connections of the procedural model with declarative constraints.

Overall, the area of hybrid modeling and mining is still relatively unexplored and there are ample opportunities for research and development of novel approaches.

3. Example

Consider a possible workflow for the treatment of a patient consisting of the following sequential steps: admittance, diagnosis, treatment, discharge, follow-up care. Such steps would be best described with a procedural language. However, each of these steps is likely to contain a high variability of different activities depending on the conditions being treated, status of the patient, analysis results, etc. Such high variability would be better described as a set of business rules, which can be confined to one of the main steps but could also apply to multiple steps.

For example, during diagnosis there may be some analysis that that will always be performed together (rule confined to one main step) but may also not be performed at all (irrelevant to this case). The decision of which analysis to perform can depend on the complaints of the patient during admittance (rule applying to multiple steps). If there is a specific complaint during admittance then this could require some events to occur during follow-up care, meaning that there could possibly be hundreds of events between when a business rule is triggered and when it is met (rule applying to multiple steps).

Table 1
Envisioned phases and the current status of the doctoral project.

Phase	Summary	Status
Systematic literature review	Identification and analysis existing notations that are suitable for process modeling. The focus is mainly on declarative and hybrid notations.	In Progress
Development of process mining approaches	Development of hybrid process mining approaches for the main process mining branches.	In Progress
Requirements formalization	Setting the scope for the new hybrid notation based on the systematic literature review.	Pending
Notation development and evaluation	Creating a prototype of the new hybrid notion and identifying further improvement opportunities via a user evaluation.	Pending
Process modeling and mining case study	Implementing a process discovery approach based on the new hybrid notation and validating it via a case study based on a real-life data set.	Pending

These are just some examples of the complexities that should be taken into account when modeling processes which have both strict and flexible aspects. Suitable notations already exist for both ends of the flexibility spectrum in isolation (e.g. BPMN and DECLARE). However, existing notations tend to be more focused on one end of the spectrum while often handling the other end of the spectrum in an ad hock manner. As a result, when working with processes having both strict and flexible aspects (such as the example above) one is often forced to prioritize precise representation of flexible aspects at the expense of strict aspects or vice versa.

4. Research Plan and Current Results

This doctoral project is currently at an early stage and therefore the current research plan is not fully fixed and some adjustments are likely to occur. We envision this doctoral project consisting of five phases starting from an extensive review of related works and culminating with the evaluation of the developed hybrid process modeling and mining approaches. An overview of the project phases and the current status is presented in Table 1. Furthermore, a short description of the first two phases together with a brief overview of the current status and results is given in the following sections.

4.1. Systematic Literature Review

The Systematic Literature Review (SLR) will be conducted based on the guidelines presented in [18]. The contribution of this SLR will be twofold. First, we will identify existing notations and languages that are suitable for modeling business processes. Second, we will analyse the characteristics of the identified languages and create a corresponding classification framework. Given that the procedural language BPMN is specified as an Object Management Group (OMG) standard [2], the SLR will focus on declarative and hybrid languages. However, we have not designed the SLR to specifically exclude other procedural languages.

The resulting classification framework will be used in the third phase of the project to formalize the requirements of the new hybrid notation and to select the possible base languages that we believe can be either successfully combined or that have some characteristics we believe can be utilized for creating the new hybrid notation. The SLR is currently in progress. The search for primary studies is completed, and we are in the study selection phase of the SLR.

4.2. Development of Process Mining Approaches

The purpose of this phase is to develop process mining approaches that would support the adoption of the hybrid process modeling notation developed during this project. Initially the focus will be on approaches that are not language specific. Later, during phases three and four, the focus will shift towards supporting the new hybrid notation. Ideally, approaches for discovery, conformance checking, and monitoring will be developed. However, this list is not yet final and will depend on the results of the SLR and the requirements formalization.

Currently we have been considering the scenario in which a single process execution is governed by multiple process specifications. For example, consider a single patient having multiple medical conditions, all of which must be treated simultaneously. This would necessitate following multiple clinical guidelines (procedural models) simultaneously, which additionally must be overlaid with basic medical knowledge (declarative models). The interplay of these models may lead to *conflicts* (situations where violating at least one model becomes inevitable). For example, different guidelines recommending to administer drugs, which on their own are safe, but may have serious side-effect when combined.

Inspired by the above-mentioned scenario, we have developed a prototype of a monitoring approach for hybrid process specifications. This approach does not focus on a specific language, but instead uses standard formal semantics for expressing procedural and declarative models. Petri Nets and LTL_f formulas [19] (respectively) were chosen at this stage, because both can be translated into finite-state automata [20, 21] in a way that allows to compute the cross-product of these automata. The cross-product will represent a conjunction of the behaviours specified by the individual models, therefore allowing for monitoring multiple process models simultaneously. Additionally, the cross-product can be augmented with violation costs such that recommendations on the next course of action in case of conflicts become possible.

Acknowledgments

The work of A. Alman was supported by the Estonian Research Council (project PRG1226) and ERDF via the IT Academy Program.

References

[1] W. M. P. van der Aalst, Process Mining - Discovery, Conformance and Enhancement of Business Processes, Springer, 2011.

[2] B. P. Model, Notation (BPMN) version 2.0, OMG Specification, Object Management Group (2011).

[3] W. M. P. van der Aalst, C. Stahl, Modeling Business Processes - A Petri Net-Oriented Approach, Cooperative Information Systems series, MIT Press, 2011.
[4] A. H. M. ter Hofstede, W. M. P. van der Aalst, M. Adams, N. Russell (Eds.), Modern Business Process Automation - YAWL and its Support Environment, Springer, 2010.
[5] M. Pesic, H. Schonenberg, W. M. P. van der Aalst, DECLARE: full support for loosely-structured processes, in: EDOC, IEEE Computer Society, 2007, pp. 287–300.
[6] T. T. Hildebrandt, R. R. Mukkamala, Declarative event-based workflow as distributed dynamic condition response graphs, in: PLACES, volume 69 of *EPTCS*, 2010, pp. 59–73.
[7] A. Artale, A. Kovtunova, M. Montali, W. M. P. van der Aalst, Modeling and reasoning over declarative data-aware processes with object-centric behavioral constraints, in: BPM, volume 11675 of *Lecture Notes in Computer Science*, Springer, 2019, pp. 139–156.
[8] S. Zugal, J. Pinggera, B. Weber, The impact of testcases on the maintainability of declarative process models, in: BMMDS/EMMSAD, volume 81 of *Lecture Notes in Business Information Processing*, Springer, 2011, pp. 163–177.
[9] P. Pichler, B. Weber, S. Zugal, J. Pinggera, J. Mendling, H. A. Reijers, Imperative versus declarative process modeling languages: An empirical investigation, in: Business Process Management Workshops (1), volume 99 of *Lecture Notes in Business Information Processing*, Springer, 2011, pp. 383–394.
[10] H. A. Reijers, T. Slaats, C. Stahl, Declarative modeling-an academic dream or the future for bpm?, in: BPM, volume 8094 of *Lecture Notes in Computer Science*, Springer, 2013, pp. 307–322.
[11] C. D. Ciccio, A. Marrella, A. Russo, Knowledge-intensive processes: Characteristics, requirements and analysis of contemporary approaches, J. Data Semant. 4 (2015) 29–57.
[12] T. Slaats, Declarative and hybrid process discovery: Recent advances and open challenges, J. Data Semant. 9 (2020) 3–20.
[13] G. D. Giacomo, M. Dumas, F. M. Maggi, M. Montali, Declarative process modeling in BPMN, in: CAiSE, volume 9097 of *Lecture Notes in Computer Science*, Springer, 2015, pp. 84–100.
[14] S. W. Sadiq, W. Sadiq, M. E. Orlowska, Pockets of flexibility in workflow specification, in: ER, volume 2224 of *Lecture Notes in Computer Science*, Springer, 2001, pp. 513–526.
[15] S. W. Sadiq, M. E. Orlowska, W. Sadiq, Specification and validation of process constraints for flexible workflows, Inf. Syst. 30 (2005) 349–378.
[16] F. M. Maggi, T. Slaats, H. A. Reijers, The automated discovery of hybrid processes, in: BPM, volume 8659 of *Lecture Notes in Computer Science*, Springer, 2014, pp. 392–399.
[17] J. D. Smedt, J. D. Weerdt, J. Vanthienen, Fusion miner: Process discovery for mixed-paradigm models, Decis. Support Syst. 77 (2015) 123–136.
[18] B. Kitchenham, S. Charters, Guidelines for performing systematic literature reviews in software engineering, 2007.
[19] A. Bauer, M. Leucker, C. Schallhart, Comparing LTL semantics for runtime verification, J. Log. Comput. 20 (2010) 651–674.
[20] M. Notomi, T. Murata, Hierarchical reachability graph of bounded petri nets for concurrent-software analysis, IEEE Trans. Software Eng. 20 (1994) 325–336.
[21] D. Giannakopoulou, K. Havelund, Automata-based verification of temporal properties on running programs, in: ASE, IEEE Computer Society, 2001, pp. 412–416.