DisCoveR: Process Mining for Knowledge-Intensive Processes with DCR Graphs

Tijs Slaats[1]

[1]Department of Computer Science, University of Copenhagen, Denmark

Abstract

Constraint-based notations aim to model processes by capturing their underlying rules instead of a limited number of potential process flows, leaving maximum flexibility for the actor to choose the best-suited order of execution for a particular process instance. Dynamic Condition Response (DCR) graphs are a constraint-based notation that has seen significant industrial adoption. In recent years there have been made significant inroads into the development of process mining algorithms and techniques for DCR Graphs. In this paper, accompanying the keynote of the same name delivered at the workshop Algorithms & Theories for the Analysis of Event Data, we discuss some of these recent advances in process mining with DCR Graphs and conclude by identifying a number of open challenges for DCR-based process mining.

Keywords

Process Mining, DCR Graphs

1. Introduction

Process modelling notations can be divided into the flow-based paradigm [1], which aims to model the control-flow of a process as tokens flowing through the activities, and the constraint-based paradigm [2, 3], which aims to model the control-flow of a process as declarative constraints between the activities controlling their temporal ordering. The latter paradigm has been argued to be well-suited to flexible knowledge processes executed by knowledge workers such as doctors and lawyers. Such processes leave the actors a significant amount of leeway in their decision making, and as a result the processes support so many variations that flow-based models turn into spaghetti diagrams.

One of the most prominent constraint-based process notations is Dynamic Condition Response (DCR) Graphs [2]. Compared to other constraint-based process notations, such as Declare [3] and DPIL [4], DCR Graphs stand out by their more widespread industrial adoption. This includes mature commercial modelling tools [5], a wealth of recorded industrial use cases [6, 7], and the integration of a DCR process engine in commercial case management and workflow tools used widely in local and central government institutions in Denmark [8].

One area where the research on DCR Graphs is still maturing is that of process mining [9, 10]. In process mining we use historical process executions, stored as event logs, to analyse processes.

ALGORITHMS & THEORIES FOR THE ANALYSIS OF EVENT DATA 2023

slaats@di.ku.dk (T. Slaats)

0000-0001-6244-6970 (T. Slaats)

Primary approaches in process mining include process discovery, where we construct models of a process based on logs, conformance checking, where we compare a log and model to each other, and process enhancement, where we improve models based on event logs.

While initially lagging behind on Declare [11, 12] in terms of the development of process mining techniques, research on DCR mining has seen significant leaps in progress in recent years. DisCoveR, a process discovery algorithm producing DCR Graphs, has been shown to be highly accurate and run-time efficient, not just when compared to other declarative miners, but also when compared to state-of-the-art imperative miners [13].

In the following section we discuss some of the recent advances in process mining with DCR Graphs, and finally we conclude and identify a number of open challenges for DCR-based process mining to mature.

2. Process Mining with DCR Graphs

We provide a short overview of the state-of-the-art in process mining with DCR Graphs and focus in particular on process discovery and conformance checking.

2.1. Process Discovery for DCR Graphs with DisCoveR

DisCoveR [13] is the current state-of-the-art discovery algorithm for mining DCR Graphs. In accordance with the constraint-based paradigm it starts from a model without any constraints and through several steps adds constraints that do not conflict with any traces in the log, resulting in a model that is perfectly fitting the training data.

The algorithm first abstracts the log into a number of data-structures representing different properties of and relations between the activities of the log. Notably some of these relations focus on finding trace-spanning correlations between activities instead of only their direct adjacency in a trace, which separates the miner from the directly-follows abstraction commonly used in flow-based discovery algorithms. The miner uses these abstractions to identify which DCR constraints can be added to the model.

Despite only identifying perfectly fitting constraints, the miner usually discovers a high number of potential constraints, which hampers the simplicity of resulting models. In the next step it therefore applies a number of optimization techniques to reduce the number of redundant constraints.

Finally, the miner applies a method that replays the log based on the currently found model in order to detect relevant constraints that were missed by the earlier steps.

Because both the abstraction of the log and the replay semantics can be optimized to be linear in the number of events in the log, the algorithm has been shown to be highly efficient. In addition one particularly efficient implementation of DisCoveR [14] makes use of a bitvector representation of both the abstractions and DCR Graph, which enables the algorithm to mine even large logs in seconds [13].

Through experimentation on public event logs the miner has also been shown to be highly accurate [15, 13], in particular allowing it to win the Process Discovery Contest at the 3rd

International Conference on Process Mining (ICPM 2021)[1].

2.2. Binary Process Discovery with the Rejection DCR Miner

Traditionally process discovery has been treated as an unary classification problem: the traces in the log (excluding potential noise) are treated as examples of desired behaviour, and the discovery algorithm mines a model matching this desired behaviour as closely as possible. As shown in earlier work [16, 17, 18] using only positive behaviour makes the mining task significantly more difficult, and in particular when mining for declarative notations it can be hard to determine which are the most meaningful constraints out of a large set of candidates that all fit the data. This can be alleviated by also using examples of behaviour that should not occur, which can identify which constraints in particular are the most useful for capturing the underlying rules of the process. Using such negative examples turns the discovery task from a unary into a binary classification problem. In recent work [19] we build on the earlier work on DisCoveR to create the Rejection DCR Miner and showed that it was able to mine models for labelled logs with a measurable increase in accuracy, and a highly significant improvement in terms of simplicity, reducing the number of mined constraints by at least one order of magnitude.

2.3. Conformance Checking and Trace Alignment

Because of the marking-based semantics of DCR Graphs, basic conformance checking through trace-replay is fairly straightforward to implement. However, trace replay is commonly seen as a relatively poor indicator of the conformance of a log [20, 21]. It does not distinguish between the number and types of deviations occurring in a trace, and simply records each trace as either satisfying the model or not. Trace alignment has been proposed as a more accurate measure of fitness and conformance [21]. It allows some deviations between the log and the model to be considered less serious than others by using a cost function parameterised by type of deviation and computing the optimal alignment between a trace and the model that minimizes this cost. In recent work [22] we have developed an algorithm for computing trace alignment for DCR Graphs which uses several optimizations to allow for the efficient computation of alignments on complex models.

3. Conclusion

While we showed in the previous section that process mining technologies for DCR Graphs have seen significant advances in the last few years, a number of challenges still remain [23]. We posit that solving some of these challenges will bring DCR mining to a level of maturity that is comparable to the state-of-the-art of constraint-based process mining.

First of all, while modelling tools for DCR Graphs have reached a high level of maturity and are leading in the field of declarative modelling, process mining tools are still an afterthought and at best available as plugins to the modelling tools which lack an intuitive user-friendly interface. We are therefore developing a toolset designed particularly for process mining users, comparable to RuM [24], which we expect to release in the near future.

[1] https://icpmconference.org/2021/process-discovery-contest/

In addition current DCR mining algorithms consider only basic control-flow and need to be extended to also consider time, data, and resource perspectives. They also do not yet make use of the various forms of hierarchy that have been proposed for DCR Graphs [25]. We expect that improving the algorithms to produce hierarchical and potentially also hybrid [26] models can significantly improve their simplicity. Research on mining DCR graphs with sub processes and time is currently underway [27], but does not yet fully encompass all aspects of multi-perspective processes and all potential forms of hierarchy.

Finally, process enhancement has not yet been studied for DCR Graphs. Recent developments create the foundation for improving DCR Graphs using event logs, e.g. trace alignment allows us to identify constraints that should be removed from a model in order to conform to a log, but more research is still needed.

References

[1] W. M. P. van der Aalst, C. Stahl, Modeling Business Processes - A Petri Net-Oriented Approach, Cooperative Information Systems series, MIT Press, 2011. URL: http://mitpress.mit.edu/books/modeling-business-processes.

[2] T. T. Hildebrandt, R. R. Mukkamala, Declarative event-based workflow as distributed dynamic condition response graphs, in: Proceedings Third Workshop on Programming Language Approaches to Concurrency and communication-cEntric Software, PLACES 2010, Paphos, Cyprus, 21st March 2010., 2010, pp. 59–73.

[3] M. Pesic, H. Schonenberg, W. M. P. van der Aalst, DECLARE: full support for loosely-structured processes, in: 11th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2007), 15-19 October 2007, Annapolis, Maryland, USA, 2007, pp. 287–300.

[4] S. Schönig, M. Zeising, The dpil framework: Tool support for agile and resource-aware business processes., BPM (Demos) 1418 (2015) 125–129.

[5] S. Debois, T. T. Hildebrandt, M. Marquard, T. Slaats, The dcr graphs process portal., in: BPM (Demos), 2016, pp. 7–11.

[6] T. T. Hildebrandt, R. R. Mukkamala, T. Slaats, Declarative modelling and safe distribution of healthcare workflows, in: Foundations of Health Informatics Engineering and Systems - First International Symposium, FHIES, 2011, pp. 39–56.

[7] S. Debois, T. Hildebrandt, M. Marquard, T. Slaats, Bridging the valley of death - a success story on danish funding schemes paving a path from technology readiness level 1 to 9, in: Accepted for 2nd International Workshop on Software Engineering Research and Industrial Practice (SER&IP 2015), 2015.

[8] T. T. Hildebrandt, A. A. Andaloussi, L. R. Christensen, S. Debois, N. P. Healy, H. A. López, M. Marquard, N. L. Møller, A. C. Petersen, T. Slaats, et al., Ecoknow: Engineering effective, co-created and compliant adaptive case management systems for knowledge workers, in: Proceedings of the International Conference on Software and System Processes, 2020, pp. 155–164.

[9] W. Van Der Aalst, Process mining: data science in action, volume 2, Springer, 2016.

[10] W. Van Der Aalst, et al., Process mining manifesto, in: Business Process Management Workshops: BPM 2011 International Workshops, Springer, 2011, pp. 169–194.
[11] C. D. Ciccio, M. Mecella, On the discovery of declarative control flows for artful processes, ACM Trans. Manage. Inf. Syst. 5 (2015) 24:1–24:37.
[12] F. M. Maggi, A. J. Mooij, W. M. P. van der Aalst, User-guided discovery of declarative process models, in: 2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM), 2011, pp. 192–199.
[13] C. O. Back, T. Slaats, T. T. Hildebrandt, M. Marquard, Discover: accurate and efficient discovery of declarative process models, International Journal on Software Tools for Technology Transfer 24 (2022) 563–587.
[14] Tijs Slaats, Discover, https://github.com/tslaats/DisCoveR, 2021.
[15] V. Nekrasaite, A. T. Parli, C. O. Back, T. Slaats, Discovering responsibilities with dynamic condition response graphs, in: Advanced Information Systems Engineering: 31st International Conference, CAiSE 2019, Rome, Italy, June 3–7, 2019, Proceedings 31, Springer, 2019, pp. 595–610.
[16] H. Ponce de León, L. Nardelli, J. Carmona, S. K. vanden Broucke, Incorporating negative information to process discovery of complex systems, Information Sciences 422 (2018) 480–496.
[17] E. Lamma, P. Mello, M. Montali, F. Riguzzi, S. Storari, Inducing declarative logic-based models from labeled traces, in: BPM, 2007, pp. 344–359.
[18] T. Slaats, S. Debois, C. O. Back, Weighing the pros and cons: Process discovery with negative examples, in: Business process management - 19th international conference, BPM 2021, Springer, 2021, pp. 47–64.
[19] T. Slaats, S. Debois, C. O. Back, A. K. F. Christfort, Foundations and practice of binary process discovery (2023).
[20] J. Carmona, B. van Dongen, A. Solti, M. Weidlich, Conformance checking, Springer. (2018).
[21] A. Adriansyah, Aligning observed and modeled behavior, Ph.D. thesis, Mathematics and Computer Science, 2014.
[22] A. K. F. Christfort, T. Slaats, Efficient optimal alignment between dynamic condition response graphs and traces, in: Accepted for the 21st International Conference on Business Process Management (BPM 2023), 2023.
[23] T. Slaats, Declarative and hybrid process discovery: Recent advances and open challenges, Journal on Data Semantics 9 (2020) 3–20.
[24] A. Alman, C. Di Ciccio, D. Haas, F. M. Maggi, A. Nolte, Rule mining with rum, in: 2020 2nd International Conference on Process Mining (ICPM), IEEE, 2020, pp. 121–128.
[25] S. Debois, T. Hildebrandt, T. Slaats, Hierarchical declarative modelling with refinement and sub-processes, in: Business Process Management: 12th International Conference, BPM 2014, Haifa, Israel, September 7-11, 2014. Proceedings 12, Springer, 2014, pp. 18–33.
[26] T. Slaats, D. M. Schunselaar, F. M. Maggi, H. A. Reijers, The semantics of hybrid process models, in: CoopIS 2016, Springer, 2016, pp. 531–551.
[27] P. Cosma, T. T. Hildebrandt, T. Slaats, Mining timed dcr graphs with subprocesses (2023).