Accurate and Efficient Discovery of Process Models from Event Logs (Extended Abstract)

Adriano Augusto[1,2]

[1] University of Tartu, Estonia
[2] University of Melbourne, Australia
a.augusto@unimelb.edu.au

Keywords: automated process discovery · process mining · event log · business process management · optimization metaheuristic · BPMN

1 Introduction

Everyday, organizations deliver services and products to their customers by enacting their business processes, the quality and efficiency of which directly influence the customer experience. In competitive business environments, achieving a great customer experience is fundamental to be a successful company. For this reason, companies rely on programmatic business process management in order to discover, analyse, improve, automate, and monitor their business processes [8].

One of the core activities of business process management is *process discovery*. The goal of process discovery is to generate a graphical representation of a business process, namely business process model, which is then used for analysis and optimization purposes. Traditionally, process discovery has been a time-consuming activity performed either by interviewing relevant process stakeholders, or by observing process participants in action, or by analysing process reference documentation. However, with the diffusion of information systems and specialised software in organizational settings, a new form of process discovery is slowly emerging, which goes by the name of *automated process discovery* [15].

Automated process discovery allows business analysts to exploit process' execution data (recorded into so-called *event logs*) to automatically generate process models. Discovering high-quality process models is extremely important to reduce the time spent to enhance them and avoid mistakes during process analysis. In the past two decades, many research studies have addressed the problem of automated discovery of business process models from event logs [18, 11, 7, 3, 17]. Despite the richness of proposals, the state-of-the-art automated process discovery approaches (APDAs) struggle to *systematically* discover accurate process models. In general, the quality of a discovered process model depends on both the input log and the APDA that is applied. When the input event log is simple, some state-of-the-art APDAs can output accurate and simple process models. However, as the complexity of the input data increases, the quality of the discovered process models can worsen quickly. Given that real-life event logs tend to be highly complex (i.e. containing noise and incomplete event records), state-of-the-art APDAs turn to be unreliable. In addition, some of these APDAs do not scale up to real-life logs in terms of time performance.

Against this backdrop, in this thesis, we address the following three research questions.

- RQ1 - What are the state-of-the-art automated process discovery approaches, their strengths and limitations?
- RQ2 - How to strike a trade-off between the various quality dimensions in automated process discovery in an *effective* and *efficient* manner?
- RQ3 - How can the accuracy of an automated process discovery approach be efficiently optimized?

2 Summary of Contributions

In order to answer RQ1, we conducted a *Systematic Literature Review* (SLR) [4] through a scientific, rigorous and replicable approach as specified by Kitchenham [10]. We formulated a set of research questions to scope the search and developed a list of search strings that we ran on different academic databases. We applied inclusion criteria to select the studies retrieved through the search. Our SLR highlights a growing interest in the field of automated process discovery, and confirms the existence of a wide and heterogeneous number of proposals. Between 2012 and 2017 more than 80 studies proposing an automated process discovery approach have been published. We grouped these studies into 34 groups, each referring to a specific APDA, and we described their features, such as: the type of model they can discover, semantic they are able to represent, type of implementation and available artefacts and tools, and type of evaluation. Despite the variety of proposals, we could clearly identify two main streams: approaches that output procedural process models, and approaches that output declarative process models. Furthermore, while the latter ones only rely on declarative statements to represent a process, the former provide various language alternatives, although most of these methods output Petri nets. The predominance of Petri nets is driven by the semantic power of this language, and by the requirements of the methods used to assess the quality of the discovered process models (chiefly, fitness and precision). Finally, we designed a benchmark framework to enable researchers to empirically compare new APDAs against existing ones in a unified setting. Our benchmark shows that existing APDAs are affected by one (or more) of the following three limitations when used with their default input parameters: (i) they achieve limited accuracy; (ii) they are computationally inefficient to be used in practice; (iii) they discover syntactically incorrect process models. To analyse the importance and the impact of the APDAs input parameters, we also ran a hyper-parameter optimization evaluation. The latter highlighted that existing APDAs can achieve better results when their input parameters are finely tuned. However, such a tuning is expensive in terms of computational power and time, sometimes requiring up to 24 hours to discover a process model as opposed to seconds (or minutes) when choosing default parameters.

To address the limitations identified in our benchmark, and to answer RQ2, we proposed a novel APDA, namely *Split Miner* [5]. Split Miner operates over six steps: *DFG and loops discovery*; *concurrency discovery*; *filtering*; *splits discovery*; *joins discovery*; and *OR-joins minimization*. The algorithms that implement these six steps have been designed with the goal to strike a trade-off between the various quality dimensions in

automated process discovery in an *effective* and *efficient* manner. In other words, each of the six steps plays a fundamental role in the discovery of a highly fitting and precise process model, that is at the same time simple and deadlock-free. About the latter property, we formally proved that Split Miner guarantees to discover sound acyclic process models, and deadlock-free cyclic process models. Differently than other APDAs that guarantee soundness [11, 7], Split Miner does not achieve such a result by enforcing full structuredness on its discovered process models, making SM the very first[3] APDA that guarantees to discover sound (or deadlock-free) process models that are not (necessarily) fully structured. Furthermore, we showed that the time complexity of Split Miner is polynomial, guaranteeing efficiency in terms of execution times. Finally, in line with its formal properties, the results of the empirical evaluation of Split Miner highlighted that our approach performs better than the state-of-the-art APDAs. In particular, Split Miner achieved with statistical significance higher F-scores (of fitness and precision) than state-of-the-art APDAs. This is achieved by balancing fitness and precision, while maintaining a low complexity of the discovered process models. The empirical evaluation also highlighted that despite SM cannot guarantee soundness for cyclic process models, SM is still able to discover sound cyclic process models. Lastly, SM proved to be the fastest APDA, discovering process models in less than a second regardless of the size of the input event logs.

As in the majority of existing APDAs, Split Miner also requires input parameters. Such parameters can be optimized to achieve considerable improvements in the quality of the process models produced. As already noted, the quality improvement APDAs achieve via hyper-parameter optimization comes at the cost of longer execution times and higher computational requirements. This inefficiency is the direct consequence of two different problems: (i) the low scalability of the accuracy measures, especially precision; and (ii) a brute-force exploration of the solution space. By addressing both problems we can tackle RQ3.

Over the past decade, several measures of fitness and precision have been proposed and empirically evaluated in the literature [9]. Existing precision measures (and to a lesser extent also fitness measures) suffer from scalability issues when applied to models discovered from real-life event logs. In addition, Tax et al. [14] have shown that none of the existing precision measures fulfils a set of five intuitive properties, namely *precision axioms*. These axioms were then revised by Syring et al. [13], who also proposed a set of desirable properties of fitness measures, namely *fitness propositions*, showing that only the latest fitness and precision measures designed by Polyvyanyy et al. [12] can fulfil the proposed properties.

In this thesis, we designed a family of fitness and precision measures [2] based on the idea of comparing the k^{th}-order Markovian abstraction of a process model against that of an event log. We showed that the proposed measures fulfil all aforementioned properties for a suitable k dependent on the log.

While fulfilling the proposed properties is desirable, this does not guarantee that the proposed measures provide intuitive results in practice. To validate the intuitiveness of our measures, we compared the ranking they induce against those induced by existing fitness and precision measures using a collection of model-log pairs proposed by

[3] According to our literature review and benchmark.

van Dongen et al. [16] (extended to cover fitness in addition to precision). For $k \geq 4$, the proposed fitness measures induce rankings that coincide with alignment-based fitness [1] – a commonly used fitness measure. Meanwhile, for $k \geq 3$, the proposed precision measures induce rankings consistent with those of the anti-alignment precision measure [16], previously posited as a ground-truth for precision measures.

A second evaluation using real-life event logs showed that the execution times of the proposed fitness measures (for $k \leq 5$) are considerably lower than existing fitness measures, except when applied to process models that contain flower structures, in which case alignment-based fitness offers the best performance. Similarly, the execution times of the proposed precision measures (for $k \leq 5$) are considerably lower than existing precision measures, except for models that contain flower structures.

The comparison between the process model and the event log Markovian abstractions allows us to identify the behaviour to add to (remove from) the process model in order to improve its fitness (precision). Indeed, such information is encoded in the edges of the event log Markovian abstraction that cannot be found in the process model Markovian abstraction (and viceversa), and it is independent from the value of k and not prone to approximation by design.

By employing our Markovian accuracy measures to efficiently explore the solution-space of APDAs based on *directly-follows graphs* (DFGs), we designed an optimization framework powered by single-solution-based metaheuristics [6], that boosts the capability of DFG-based APDAs to discover process models with higher accuracy. The core idea of our framework is to perturb the intermediate representation of an event log used by the majority of the available APDAs, namely the Directly-follows Graph (DFG). Specifically, we consider perturbations that add or remove edges with the aim of improving fitness or precision, and allowing the underlying APDA to discover a process model from the perturbed DFG.

We instantiated our framework for three state-of-the-art APDAs: Split Miner, Fodina, and Inductive Miner, and using a benchmark of 20 real-life logs, we compared the accuracy gains yielded by four optimization metaheuristics relative to each other and relative to the hyper-parameter optimized APDA. The evaluation of our optimization framework highlights its effectiveness in optimizing DFG-based APDAs, allowing APDAs to explore their solution-space beyond the boundaries of hyper-parameter optimization and most of the times in a faster manner, ultimately discovering more accurate process models in less time, compared to the traditional hyper-parameters optimization.

In order to foster reproducibility and reuse, all the artefacts designed and developed for this thesis are publicly available as standalone open-source Java command-line applications.[4] Split Miner, our core contribution to the field of automated process discovery, has also been integrated into Apromore, an open-source business process analytics platform used by academics and practitioners alike.

References

1. A. Adriansyah, B. van Dongen, and W. van der Aalst. Conformance checking using cost-based fitness analysis. In *EDOC*. IEEE, 2011.

[4] Applications available at `https://apromore.org/platform/tools/`

2. A. Augusto, A. Armas Cervantes, R. Conforti, M. Dumas, M. La Rosa, and D. Reissner. Measuring fitness and precision of automatically discovered process models: A principled and scalable approach. Technical report, University of Melbourne, 2019.
3. A. Augusto, R. Conforti, M. Dumas, M. La Rosa, and G. Bruno. Automated discovery of structured process models: Discover structured vs. discover and structure. In *Conceptual Modeling: 35th International Conference, ER 2016, Gifu, Japan, November 14-17, 2016, Proceedings*, pages 313–329. Springer, 2016.
4. A. Augusto, R. Conforti, M. Dumas, M. La Rosa, F.M. Maggi, A. Marrella, M. Mecella, and A. Soo. Automated discovery of process models from event logs: Review and benchmark. *IEEE TKDE*, 31(4), 2019.
5. A. Augusto, R. Conforti, M. Dumas, M. La Rosa, and A. Polyvyanyy. Split miner: automated discovery of accurate and simple business process models from event logs. *KAIS*, 2018.
6. A. Augusto, M. Dumas, and M. La Rosa. Metaheuristic optimization for automated business process discovery. In *BPM*. Springer, 2019.
7. J. Buijs, B.F. van Dongen, and W.M.P. van der Aalst. Quality dimensions in process discovery: The importance of fitness, precision, generalization and simplicity. *International Journal of Cooperative Information Systems*, 23(01):1440001, 2014.
8. M. Dumas, M. La Rosa, J. Mendling, and H.A. Reijers. *Fundamentals of business process management (2nd Edition)*. Springer, 2018.
9. G. Janssenswillen, N. Donders, T. Jouck, and B. Depaire. A comparative study of existing quality measures for process discovery. *Information Systems*, 71:1–15, 2017.
10. B. Kitchenham. Procedures for performing systematic reviews. *Keele, UK, Keele University*, 33(2004):1–26, 2004.
11. S.J.J Leemans, D. Fahland, and W.M.P. van der Aalst. Discovering Block-Structured Process Models from Incomplete Event Logs. In *Application and Theory of Petri Nets and Concurrency: 35th International Conference, PETRI NETS 2014, Tunis, Tunisia, June 23-27, 2014. Proceedings*, pages 91–110. Springer International Publishing, 2014.
12. A. Polyvyanyy, A. Solti, M. Weidlich, C. Di Ciccio, and J. Mendling. Monotone precision and recall measures for comparing executions and specifications of dynamic systems. *CoRR*, abs/1812.07334, 2018.
13. A.F. Syring, N. Tax, and W.M.P. van der Aalst. Evaluating conformance measures in process mining using conformance propositions (extended version). *arXiv preprint arXiv:1909.02393*, 2019.
14. N. Tax, X. Lu, N. Sidorova, D. Fahland, and W.M.P. van der Aalst. The imprecisions of precision measures in process mining. *Information Processing Letters*, 135, 2018.
15. W.M.P. van der Aalst. *Process Mining - Data Science in Action*. Springer, 2016.
16. B.F. van Dongen, J. Carmona, and T. Chatain. A unified approach for measuring precision and generalization based on anti-alignments. In *BPM*. Springer, 2016.
17. S.K.L.M. vanden Broucke and J. De Weerdt. Fodina: a robust and flexible heuristic process discovery technique. *Decision Support Systems*, 2017.
18. A.J.M.M. Weijters and J.T.S. Ribeiro. Flexible heuristics miner (FHM). In *Computational Intelligence and Data Mining (CIDM), 2011 IEEE Symposium on*, pages 310–317. IEEE, 2011.