

Perfect Process Models?! A Process-Discovery Technique Optimizing Over Quality Measures

Patrizia Schalk¹

¹University of Augsburg, Universitätsstraße 6a, 86135, Augsburg, Germany

Abstract

The holy grail in process mining is a process model that scores perfectly in fitness, precision, generalization and simplicity. This is why there are not only many process discovery algorithms, but also many ways to calculate a numerical representation for these four quality dimensions. The thesis described in this extended abstract aims to answer the following question: Can we automate finding a process model that is optimal with respect to selectable but fixed measures of fitness, precision, generalization and simplicity?

Keywords

Process Mining, Conformance Checking, Fitness, Precision, Generalization, Simplicity, Optimization

Process mining is about finding and analyzing a descriptive model for an existing and running business process. Most process mining techniques assume the existence of a so-called event log: A collection of recorded behavior of the business process in question. Process discovery algorithms take such an event log as input and return a formal model for the analysis and enhancement of the business process [1]. Naturally, each process discovery algorithm makes decisions during its execution that influence the model it returns. In turn, different process discovery algorithms may return different process models for the same event log as input. This poses the question of which process model is the best for the business process.

To answer this question, four quality dimensions proved themselves useful for the process mining community: Fitness, precision, generalization and simplicity [2]. Fitness (or recall) evaluates how much of the behavior in the event log is featured in the process model, while precision measures how much of the behavior of the model is featured in the event log. Generalization reviews how well the process model is able to replay behavior of the business process that is not part of the event log. Simplicity (or complexity) judges how easy the process model is to understand. For each of these dimensions, there are several techniques that compute a value between 0 and 1 which indicates how good a process model performs for the specific quality dimension. A value close to 1 means that the process model performs (almost) perfectly. While there are many quality metrics for fitness [3, 4, 5, 6, 7, 8, 9] and precision [5, 6, 7, 8, 9, 10, 11, 12, 13], there are much fewer metrics defined for generalization [8, 9, 10, 11] and simplicity or complexity [14], as they are harder to formalize. Each of the aforementioned quality metrics perform their calculations differently, and thus sometimes return different values for the same pair of model and event log.


ICPM Doctoral Consortium and Demo Track 2023

✉ patrizia.schalk@informatik.uni-augsburg.de (P. Schalk)

ORCID 0009-0001-7757-6628 (P. Schalk)



© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

Research regarding these quality metrics is still active, because already existing metrics tend to lack desired features [15] or don't correctly measure what they were created for [16]. Janssenswillen et al. [17] found a high correlation between fitness and precision, which raises the desire for more independent metrics. The generalization metric, on the other hand, is severely under-developed [18], since generalization is hard to define formally with having just the event log as the representation for the system behavior. Simplicity has a similar problem, as it aims to "prefer an easier model over a more complex one" [1] – a description that is highly subjective and dependent on the specific use-case. Nonetheless, these imperfect quality metrics are widely used in practice to evaluate mined process models. It is common practice to improve a process model based on these metrics to get a model that features better quality scores. In these cases, the outcome is a well-performing process model with respect to the chosen quality metrics. However, this procedure is not only time-intensive, but also may not yield the best possible result. Therefore, in this thesis, we investigate whether it is possible to automatically discover a process model that is optimal for a chosen set of quality metrics. The idea of this algorithm is that the user specifies an arbitrary but fixed metric for each of the dimensions fitness, precision, generalization and simplicity, as well as an event log and a weighting function. The weighting function indicates how important each quality dimension is compared to the others. As output, the process discovery algorithm returns a process model in the form of a Petri net that is optimal regarding the specified quality metrics and the weighting function.

With this process discovery algorithm, a user can specify any computable and deterministic quality metric to get a "perfect" process model with respect to their choices. The results of this process discovery technique become predictable and depend only on the chosen quality metrics. However, the aim of this process discovery algorithm is not to create actually perfect process models, since this is not possible with imperfect metrics. Instead, when optimizing over a quality metric, we can get a better understanding of which structures are rewarded or punished by the metric. This makes it easier to further investigate existing quality metrics and tailor them towards metrics that satisfactorily fulfill their purposes. For example, this can be achieved by creating an artificial system-model, as well as an event log for it, and checking to which extent the optimizing strategy is able to rediscover the original model (i.e. the rediscovery problem).

There are already some algorithms discovering process models that are optimal for some quality dimensions. The inductive miner [19] by design creates a process model with perfect fitness and keeps this process model simple by not allowing duplicate transition labels. The ILP miner [20] optimizes over fitness and precision and keeps the process model simple by not allowing duplicate or silent transitions. Because of the optimal results, these two discovery algorithms are widely used and researchers still try to improve them. However, they don't offer to consider metrics for generalization and don't allow choosing the quality metrics whose scores should be optimized.

At first sight, the Evolutionary Tree Miner [21] seems strongly related to our ideas. It as well optimizes over all four quality dimensions. However, it does not give a user the freedom to choose any quality metric, but fixes the alignment-based fitness [5] and the escaping edge precision [13]. As for simplicity, the authors punish duplicate labels and events that occur in the event log but not in the process model. For generalization, the authors detect less frequently used parts of the model and give them a low generalization score. The generalization metric

is nondeterministic for models that can replay a trace in more than one way. It is unclear which result this metric should yield for unfitting traces. Furthermore, we can easily “trick” this generalization metric by adding a chain of silent transitions in front of the original start place of a workflow net. These silent transitions are then visited in every trace. This would improve the generalization score without adding more behavior to the process model, making the metric unfit for our purposes. Finally, the Evolutionary Tree Miner is a genetic algorithm and therefore nondeterministic.

Our goal is to create a process discovery algorithm that is deterministic and able to optimize over a set of quality measures defined by the user. To do so, our initial idea is to investigate common optimization techniques and to formulate optimization problems for existing quality metrics. We plan to start with the fitness metric using alignments [5], the precision and generalization metrics using anti alignments [10] and the Cardoso metric for simplicity [22], as these metrics are widely used and accepted. Later, we will extend our research to other quality metrics, as well as metrics that don’t focus just on the control flow. Our approach is as follows: First, we formulate an optimization problem for each of the chosen metrics and prove its correctness by a formal proof. Afterward, we take two of the metrics and implement an algorithm that optimizes the aforementioned optimization problems for these two metrics. For this step, we use metrics that are orthogonal to each other, as optimizing over fitness and precision would lead to the trace model and optimizing over generalization and simplicity would lead to the flower model. Finally, we take metrics for all four quality dimensions and evaluate the results by performing a case study that investigates if the output process model is as useful in practice as the scores for the metrics suggest. We don’t expect this to be the case, as it is already known that each quality metric has severe weaknesses [15] and that some don’t value what they are constructed for at all [16]. As the output of our process discovery technique, we choose free-choice workflow nets, since they can easily be transformed into BPMN [23] and could therefore make the application handy for a broader field. Furthermore, many hard problems are easier to solve on free-choice Petri nets, which makes them a very handy tool. As non free-choice constructs are often undesired in process models, we accept this restriction for our output.

We have already identified some challenges for our approach: First, most simplicity measures are defined for BPMN or EPC, but we aim to produce free-choice workflow nets. However, translating existing simplicity- or complexity measures [14] to measures for free-choice Petri nets is either straight-forward or a possible translation was already proposed [24]. Tough, we still need to find a way to normalize the complexity metrics if we want to use them in the target function. Second, optimization is undefined if the target function is nondeterministic. But some quality metrics, like the token-based replay fitness [9], are nondeterministic. We plan to either find deterministic versions of these quality metrics or to focus on Petri net types where the metrics are deterministic. Third, depending on the runtime for calculating the quality metric, the runtime for our process discovery might be high. This means that our approach might not be easily scalable and not applicable for large event logs. Even though we plan to investigate the runtime and to find ways to make the algorithm more scalable, the runtime is not our biggest concern. This is because this process discovery algorithm removes the necessity of computing the score of the quality metrics for the model afterward. Therefore, we can accept a higher runtime if it is not worse than the runtime needed to compute the quality scores.

References

- [1] W. M. P. van der Aalst, *Process Mining - Data Science in Action*, Second Edition, Springer, 2016. doi:10.1007/978-3-662-49851-4.
- [2] J. Carmona, B. F. van Dongen, A. Solti, M. Weidlich, *Conformance Checking - Relating Processes and Models*, Springer, 2018. doi:10.1007/978-3-319-99414-7.
- [3] A. Weijters, W. Aalst, van der, A. Alves De Medeiros, *Process mining with the Heuristic-sMiner algorithm*, BETA publicatie : working papers, Technische Universiteit Eindhoven, 2006.
- [4] W. M. P. van der Aalst, T. Weijters, L. Maruster, *Workflow mining: Discovering process models from event logs*, *IEEE Trans. Knowl. Data Eng.* 16 (2004) 1128–1142. doi:10.1109/TKDE.2004.47.
- [5] W. M. P. van der Aalst, A. Adriansyah, B. F. van Dongen, *Replaying history on process models for conformance checking and performance analysis*, *WIREs Data Mining Knowl. Discov.* 2 (2012) 182–192. doi:10.1002/widm.1045.
- [6] S. Goedertier, D. Martens, J. Vanthienen, B. Baesens, *Robust process discovery with artificial negative events*, *J. Mach. Learn. Res.* 10 (2009) 1305–1340. doi:10.5555/1577069.1577113.
- [7] S. J. J. Leemans, D. Fahland, W. M. P. van der Aalst, *Scalable process discovery and conformance checking*, *Softw. Syst. Model.* 17 (2018) 599–631. doi:10.1007/s10270-016-0545-x.
- [8] A. Solti, C. Di Ciccio, J. Mendling, A. Polyvyanny, M. Weidlich, *Behavioural quotients for precision and recall in process mining*, 2018.
- [9] A. Rozinat, W. M. P. van der Aalst, *Conformance checking of processes based on monitoring real behavior*, *Inf. Syst.* 33 (2008) 64–95. doi:10.1016/j.is.2007.07.001.
- [10] B. F. van Dongen, J. Carmona, T. Chatain, *A unified approach for measuring precision and generalization based on anti-alignments*, in: *Business Process Management - 14th International Conference, BPM 2016, Rio de Janeiro, Brazil, September 18-22, 2016. Proceedings*, volume 9850 of *Lecture Notes in Computer Science*, Springer, 2016, pp. 39–56. doi:10.1007/978-3-319-45348-4_3.
- [11] S. K. L. M. vanden Broucke, J. D. Weerd, J. Vanthienen, B. Baesens, *Determining process model precision and generalization with weighted artificial negative events*, *IEEE Trans. Knowl. Data Eng.* 26 (2014) 1877–1889. doi:10.1109/TKDE.2013.130.
- [12] G. Janssenswillen, N. Donders, T. Jouck, B. Depaire, *A comparative study of existing quality measures for process discovery*, *Inf. Syst.* 71 (2017) 1–15. doi:10.1016/j.is.2017.06.002.
- [13] J. Munoz-Gama, J. Carmona, *A fresh look at precision in process conformance*, in: *Business Process Management - 8th International Conference, BPM 2010, Hoboken, NJ, USA, September 13-16, 2010. Proceedings*, volume 6336 of *Lecture Notes in Computer Science*, Springer, 2010, pp. 211–226. doi:10.1007/978-3-642-15618-2_16.
- [14] J. Lieben, T. Jouck, B. Depaire, M. Jans, *An improved way for measuring simplicity during process discovery*, in: *Enterprise and Organizational Modeling and Simulation - 14th International Workshop, EOMAS 2018, Held at CAiSE 2018, Tallinn, Estonia, June 11-12, 2018, Selected Papers*, volume 332 of *Lecture Notes in Business Information Processing*,

- Springer, 2018, pp. 49–62. doi:10.1007/978-3-030-00787-4_4.
- [15] A. F. Syring, N. Tax, W. M. P. van der Aalst, Evaluating conformance measures in process mining using conformance propositions, *Trans. Petri Nets Other Model. Concurr.* 14 (2019) 192–221. doi:10.1007/978-3-662-60651-3_8.
 - [16] N. Tax, X. Lu, N. Sidorova, D. Fahland, W. M. P. van der Aalst, The imprecisions of precision measures in process mining, *Inf. Process. Lett.* 135 (2018) 1–8. doi:10.1016/j.ip1.2018.01.013.
 - [17] G. Janssenswillen, N. Donders, T. Jouck, B. Depaire, A comparative study of existing quality measures for process discovery, *Inf. Syst.* 71 (2017) 1–15. doi:10.1016/j.is.2017.06.002.
 - [18] A. Polyvyanyy, A. Moffat, L. García-Bañuelos, Bootstrapping generalization of process models discovered from event data, in: *Advanced Information Systems Engineering - 34th International Conference, CAiSE 2022, Leuven, Belgium, June 6-10, 2022, Proceedings*, volume 13295 of *Lecture Notes in Computer Science*, Springer, 2022, pp. 36–54. doi:10.1007/978-3-031-07472-1_3.
 - [19] S. J. J. Leemans, D. Fahland, W. M. P. van der Aalst, Discovering block-structured process models from event logs - A constructive approach, in: *Application and Theory of Petri Nets and Concurrency - 34th International Conference, PETRI NETS 2013, Milan, Italy, June 24-28, 2013. Proceedings*, volume 7927 of *Lecture Notes in Computer Science*, Springer, 2013, pp. 311–329. doi:10.1007/978-3-642-38697-8_17.
 - [20] J. M. E. M. van der Werf, B. F. van Dongen, C. A. J. Hurkens, A. Serebrenik, Process discovery using integer linear programming, in: *Applications and Theory of Petri Nets, 29th International Conference, PETRI NETS 2008, Xi'an, China, June 23-27, 2008. Proceedings*, volume 5062 of *Lecture Notes in Computer Science*, Springer, 2008, pp. 368–387. doi:10.1007/978-3-540-68746-7_24.
 - [21] J. C. A. M. Buijs, B. F. van Dongen, W. M. P. van der Aalst, On the role of fitness, precision, generalization and simplicity in process discovery, in: *On the Move to Meaningful Internet Systems: OTM 2012, Confederated International Conferences: CoopIS, DOA-SVI, and ODBASE 2012, Rome, Italy, September 10-14, 2012. Proceedings, Part I*, volume 7565 of *Lecture Notes in Computer Science*, Springer, 2012, pp. 305–322. doi:10.1007/978-3-642-33606-5_19.
 - [22] J. Cardoso, Process control-flow complexity metric: An empirical validation, in: *2006 IEEE International Conference on Services Computing (SCC 2006)*, 18-22 September 2006, Chicago, Illinois, USA, IEEE Computer Society, 2006, pp. 167–173. doi:10.1109/SCC.2006.82.
 - [23] C. Favre, D. Fahland, H. Völzer, The relationship between workflow graphs and free-choice workflow nets, *Inf. Syst.* 47 (2015) 197–219. doi:10.1016/j.is.2013.12.004.
 - [24] K. B. Lassen, W. M. P. van der Aalst, Complexity metrics for workflow nets, *Inf. Softw. Technol.* 51 (2009) 610–626. doi:10.1016/j.infsof.2008.08.005.