

# A Framework for Interactive Mining and Retrieval from Process Traces

Doctoral Consortium ICCBR 2016

L. Canensi

Department of Computer Science, Università di Torino, Italy  
canensi@di.unito.it

## 1 Research Summary

Processes are everywhere. We can find processes in hospital, companies, universities, institutions and so on. Therefore, it is no surprise that the usage of information systems and enterprise resource planning tools has been rapidly growing in organizations and companies worldwide, of all kinds and sizes. The sequences (traces) of actions that have been completed at an organization is usually stored in the so-called event log. Event logs constitute a very rich source of experiential knowledge, fundamental to support different tasks, like, e.g., mining a process model, or retrieving similar traces, in order to make predictions on the currently running process instance. In the Business Process Management (BPM) field, it is widely recognized that these activities are very important, and can be used to improve process execution and performance. However, all the works in the literature treat these activities separately, applying different theoretical and methodological approaches to each of them.

Instead, in my PhD thesis I propose a comprehensive approach, aiming at integrating the construction of the process model and its analysis.

In particular, my Thesis is composed of three steps:

- initial process model construction
- trace retrieval
- interactive process model abstraction and refinement.

The approach allows the user to take advantage of the integration of the three activities above; indeed, the initial process model we build is

---

Copyright © 2016 for this paper by its authors. Copying permitted for private and academic purposes.

In Proceedings of the ICCBR 2016 Workshops.  
Atlanta, Georgia, United States of America

used as an indexing structure to speed up trace retrieval; moreover, we exploit a unified methodological solution that allows to retrieve traces and specific paths in the model. We then exploit retrieval results as a basis to build a more abstract process model, in an interactive fashion.

Indeed, our innovative approach is able to "reconcile" apparently heterogeneous needs of business process management, supporting a user-friendly interaction with domain experts and to take advantage of all the available knowledge sources in a comprehensive way,

The following subsections describe in more detail the various steps of the thesis.

### 1.1 Initial process model construction

The first part of the work is related to the Process Mining (PM) field. PM is a research discipline that discovers, monitors, and improves real processes, by extracting knowledge from traces in the event logs, readily available from today systems [1]. Each trace consists of an ordered sequence of activities. The mined process model can be used to understand, adapt and modify the real process to increase performance and become a high quality process. There are three main classes of process mining techniques [2]:

- *Discovery*: the discovery of new process models based only on the event log
- *Conformance*: conformance verification of the recorded behavior with respect to a provided model
- *Enhancement*: extension of an existing process model using the information from the event log

Most of the phases of a process life cycle [3] can benefit of PM techniques: they can be adopted to analyse an existing model, to diagnose problem, and possibly to adapt/redesign/tune the process model itself.

All these considerations lead to define PM as a very important instrument for modern organizations that need to manage non-trivial operational processes.

My research deals with process discovery, the most relevant and widely used PM activity.

There are many different approaches to process discovery. Different algorithms have their own specificities: some focus on local relations between activities in the logs (Heuristic Miner [4]), while others focus on the whole log (Fuzzy Miner [5], Genetic Miner [6]). However, all of these

algorithms operate at a unique, system-defined level of abstraction. Instead, in many domains, it would be very important to have the ability to build/refine models, working at different levels of abstraction.

The contribution of my thesis to the process discovery area consists in a novel tool that allows the construction of a data structure (called log-tree), that can be used both as an initial model of the process (to be possibly abstracted in a further interactive session with the user), and as an index, to speed-up trace retrieval. The log-tree is a new representation formalism which has a well defined semantics (unambiguous) and maintain a direct connection between traces in the log and elements in the process model. Therefore, it is usable as an index of the traces and as a standard process model. In order to build an index, the algorithm guarantees that the log-tree only includes paths actually recorded as traces in the event log. In order to realize this objective, the algorithm: (1) makes an intensive use of all the available frequency information about the activities recorded in the event log; (2) properly forks the model into various branches, on the basis of the different execution contexts, implicitly represented by subsets of the traces in the event log.

## 1.2 Trace Retrieval

The second part of the thesis deals with trace retrieval. When the input trace is a currently running process instance, the retrieval of similar, already completed instances recorded as traces in the log, can enable the user to make predictions about the current instance completion, or can recommend suitable actions, resources or routing decisions to be adopted next; these goals are treated in the literature within the operational support research area [2]. Trace retrieval has been recently considered in the Case Base Reasoning (CBR) [7] literature. All approaches use traces as sources for retrieving and reusing user's experience. For instance, the work in [8] proposes trace-based reasoning, a CBR approach where cases are not explicitly stored in a library, but are implicitly recorded as "episodes" within traces. The paper in [9] extends that work, and defines a similarity measure to compare episodes extracted from traces. These work, however, do not aim at providing support in business process management, such as prediction for operational support, or pattern identification for abstraction. The goal of these tools is therefore usually very different from ours.

Moreover, current trace retrieval approaches typically take in input a fully specified trace. This is a severe limitation, because sometimes, the goal is to find races that fulfil partially specified patterns. We can

deal with this issue, by means of a powerful query language and query answering approach.

The log-tree is used as an index, allowing fast retrieval from the available event log. Thanks to its characteristics and methodological solutions, the tool implements operational support tasks in a flexible, efficient and user friendly way.

It is worth noting that our approach, besides retrieving traces from the log-tree, also allows to retrieve paths from a generic (more abstract) process model (a graph).

### 1.3 Interactive process model abstraction

The log-tree is already a process model, which guarantees a maximal precision (i.e., it does not represent any behavior that was not recorded in the traces). However, in some cases, it may be useful to have a more general process model, with consequent loss of precision, that abstracts from negligible details. So, the log-tree can be seen as a starting point to generate a more abstract process model in an interactive session of work, where the user is always allowed to inspect the current output, and possibly backtrack to the previous step. The ability of retrieving specific traces/paths in the model, corresponding to properties/situations of interest, is very useful in supporting the abstraction process, by suggesting portions of the model that could be merged. To the best of our knowledge, path retrieval and merging have never been described in the business process management literature, possibly with the exception of the fuzzy miner [5], which provides a functionality to cluster (merge) actions into macro actions. Moreover, no literature contribution provides a unifying framework, where a suite of different facilities are properly integrated as in our work, to support process mining at different level of abstraction, and efficient trace/path retrieval (also responding to abstract query patterns).

## References

1. "<http://www.win.tue.nl/ieeetfpm>." IEEE Taskforce on Process Mining: Process Mining Manifesto (last accessed on 4/11/2013).
2. W. M. P. van der Aalst, *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer Publishing Company, Incorporated, 1st ed., 2011.
3. W. Scacchi and P. Mi, "Process life cycle engineering: A knowledge-based approach and environment.," *Int. Syst. in Accounting, Finance and Management*, vol. 6, no. 2, pp. 83–107, 1997.
4. A. Weijters, W. V. der Aalst, and A. A. de Medeiros, *Process Mining with the Heuristic Miner Algorithm, WP 166*. Eindhoven University of Technology, Eindhoven, 2006.

5. C. W. Günther and W. M. P. Van Der Aalst, “Fuzzy Mining: Adaptive Process Simplification Based on Multi-perspective Metrics,” in *Proceedings of the 5th International Conference on Business Process Management, BPM’07*, (Brisbane, Australia), pp. 328–343, Springer-Verlag, 2007.
6. A. K. A. D. Medeiros and A. J. M. M. Weijters, “Genetic process mining,” in *Applications and Theory of Petri Nets 2005, volume 3536 of Lecture Notes in Computer Science*, pp. 48–69, Springer-Verlag, 2005.
7. A. Aamodt and E. Plaza, “Case-based reasoning: foundational issues, methodological variations and systems approaches,” *AI Communications*, vol. 7, pp. 39–59, 1994.
8. A. Cordier, M. Lefevre, P.-A. Champin, O. Georgeon, and A. Mille, “Trace-Based Reasoning — Modeling interaction traces for reasoning on experiences,” in *The 26th International FLAIRS Conference*, May 2013.
9. R. Zarka, A. Cordier, E. Egyed-Zsigmond, L. Lamontagne, and A. Mille, “Similarity Measures to Compare Episodes in Modeled Traces,” in *International Case-Based Reasoning Conference (ICCBR 2013)* (Springer, ed.), Lecture Notes in Computer Science, pp. 358–372, Springer Berlin Heidelberg, July 2013.