A Tool for Visualizing Costs of Process Variants through Directed Rooted Trees

Ignacio Velásquez[1,*], Marcos Sepúlveda[1]

[1]*Department of Computer Science, Pontificia Universidad Católica de Chile*

Abstract

Models depicting the behavior and performance of processes can be discovered through Process Mining. These models often facilitate visualizing alternate paths and rework through control-flow logic, but identifying all process variants in such models can become a complex endeavor. To address this, a tool that allows visualizing all process variants within a single model is proposed, in the form of Directed Rooted Trees whose leaves are the end states of variants. The generated trees are also decorated with cost information for quickly comparing the cost performance of variants. The tool is expected to be useful for performing exploratory analyses.

Keywords

Process Mining, Process Discovery, Performance Analysis, Cost Dimension, Directed Rooted Tree

1. Introduction

Process Mining (PM) is a discipline that allows the analysis of processes based on event logs containing records of their executions [1]. One of the tasks of PM is process discovery, i.e., obtaining process models depicting the behavior of the process that is observed in the event log [2]. These models can be enriched with process metrics like the duration or the costs of activities, which allows their analysis through a performance perspective [1]. By filtering the event log, it is possible to discover process models that allow comparing the performance of distinct variants of the process [3]. A variant is a subset of executions of a process that can be distinguished from others based on some characteristic, such as the occurring activities and their relative execution order [3]. Discovered process models are often created in a way that allows the visualization of alternate execution paths and rework cycles through control-flow logic [1]. As an example, Directly-Follows Graphs (DFG), which are commonly used for process visualization in commercial PM tools [1], connect activities based on their directly-follows relations. That is, if activity "B" occurs directly after activity "A" in any trace of the event log, an arc is drawn from the box representing "A" to the box representing "B". The limitation of the above is that, when considering complex processes with multiple execution variants, it becomes a complex endeavor to identify these variants within the same process model. As a minimal example, Figure 1 (a) shows a DFG that could be either the representation of the two traces "A" and "AAB" or of the two traces "AA" and "AB". However, it is not possible to discern between

BPM 2023 - Demos & Resources, September 11–15, 2023, Utrecht, The Netherlands

*Corresponding author.

iavelasquez@uc.cl (I. Velásquez); marcos.sepulveda@uc.cl (M. Sepúlveda)

the two without applying filters or analyzing the data within the event log.

To address the above, this work proposes a tool that allows the visualization of all process variants within a single process model. Specifically, the proposed process model is in the form of a transition system constructed as a Directed Rooted Tree (DRT). In transition systems, the events of the process are the transitions that imply the flow from one state of the system to another [1]. A DRT, or arborescence, is an acyclic graph that has exactly one path from its root to each vertex [4]. Based on this definition, every bifurcating behavior observed in the event log generates a separate set of transitions within the proposed DRT visualization. Thus, every leaf of the resulting model corresponds to the end state of a process variant. Using the same minimal example as above, Figure 1 (b) shows the DRT generated by the tool. This DRT allows observing that the actual traces of the event log are "AA" and "AB".

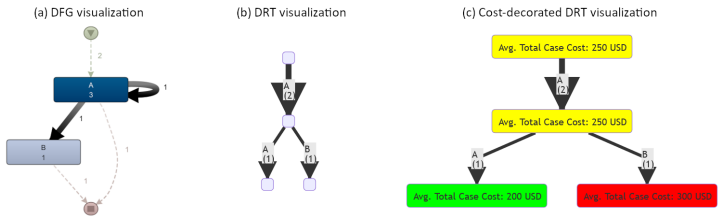

Figure 1: Distinct visualizations of event log consisting of the "AA" and "AB" traces,

Transition systems can be extended to include additional information [5, 6]. To provide additional value with the proposed tool, the use of cost-annotated event logs is supported [5]. By uploading such a log, the average total, accumulated, and remaining costs of cases flowing through every state can be observed, as well as the average, minimum, and maximum costs of every transition. States are also colored based on the average total cost of flowing cases. Using the decorated information, the visualization provided by the tool allows quickly comparing the cost performance of process variants, such as identifying the most expensive ones. Figure 1 (c) shows the DRT of the cost-annotated minimal example generated by the tool.

The DRT generated by the proposed tool possess similarities with context trees [7, 8]. Both DRT and context trees allow visualizing the bifurcating behavior of event logs. On one hand, the nodes in context trees indicate the activity executed in a given path, and its frequency, which has facilitated the use of these trees for trace clustering. On the other hand, while the proposed DRT maintain the same activity information in their transitions, the focus is on visualizing the performance behavior of variants through the annotated aggregation metrics.

The remainder of this work is structured as follows. Section 2 describes the main features of the tool, as well as its interface. In Section 3, the maturity of the tool is discussed. Section 4 provides links for the demonstration of the tool in the form of a tutorial document and video. Finally, the conclusions of this work are presented in Section 5.

2. Tool Features

The proposed tool is available at https://bit.ly/drt-variant-costs. It has been developed as a web application using the Angular framework [9]. For generating the DRT, the Mermaid

diagramming and charting tool [10] is utilized. Figure 2 shows the interface visualized when accessing the tool. This interface can be split into three areas: (1) the buttons enabling the distinct features of the tool, (2) the diagram area, which will contain the Directed Rooted Trees (DRT) generated by the tool, and (3) the format requirements for the event logs to load.

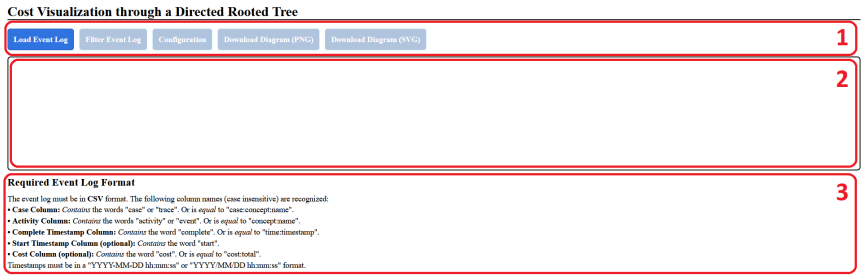

Figure 2: Interface of the proposed tool.

The buttons in area (1) enable the distinct features of the tool. The "Load Event Log" button visualizes a modal for loading an event log to the tool. Once loaded, the tool automatically generates the resulting DRT. The "Filter Event Log" button visualizes a modal containing three types of filters: “Activities to Include”, which filters the diagram to only visualize events of selected activities; “Start Activities”, which filters the diagram to only include cases that start with selected activities; and “End Activities”, which filters the diagram to only include cases that end with selected activities. The "Configuration" button visualizes a modal that allows changing the displayed currency, and the visualization of additional metrics in every state and transition of the DRT. Finally, the "Download Diagram (PNG)" and "Download Diagram (SVG)" buttons allow the download of a *.png or a *.svg file, respectively, containing the generated DRT.

By default, the states (nodes) of the transition system of the generated DRT contain the average case cost of executions flowing through them, whereas transitions (arcs) contain the activities executed between every state, and their frequency. States are colored using a traffic light color range (from green, to yellow, to red). The closer to green, the cheaper the average cost of executions flowing through the state. Conversely, the closer to red, the more expensive the average cost of flowing executions. The line width of transitions relates to the case frequency, i.e., a wider transition line corresponds to one with a higher occurrence frequency.

3. Tool Maturity

The tool is currently in a stable version. That is, no major bugs are known and all the features described in this work are implemented. Nevertheless, the maturity of the tool can be described in relation to its current limitations.

First, as any transition system, the generated DRT has problems expressing parallelism. Parallelism occurs when two activities of a case overlap [1]. As future work, it is desirable to devise a tool with similar features, but that generates models using a notation that supports parallelism, such as BPMN [11]. Such a notation could also be used to consider the existence of process variants in relation to case or activity attributes, rather than solely control-flow.

Second, only visualization of cost performance is currently supported. As future work, the possibility of extending the tool to also support waiting times between activities is considered. The visualization of any metric that can be aggregated is also foreseen.

Third, the tool has been designed for processing any event log that complies with the required format. However, loading event logs containing large numbers of cases (e.g., over 10,000) will result in long processing times. Moreover, the utility of the tool is reduced when loading event logs with excessive numbers of variants (e.g, over 100), as their comparison would be difficult without applying filters. To facilitate the analysis of processes with several variants, the trace clustering techniques used for context trees [7, 8] could be evaluated as future work. Regarding the performance of the tool, a revision of the source code is considered.

Fourth, design limitations, such as the currently required event log format, could be addressed through future work. Namely, through input boxes for users to indicate column names of the event log or the format of timestamps. The possibility of downloading the DRT in editable formats, such as JSON, or the notation of the used diagramming library, is also considered.

The tool has not been used for real case studies yet. However, it was tested with the road traffic fine management event log, which contains real-life records of an information system from an Italian city [12]. The DRT for this process was generated correctly.

4. Tool Demonstration

A tutorial document containing step-by-step instructions on how to utilize the tool has been constructed. This document can be found at https://bit.ly/drt-tutorial-document. A video summarizing this tutorial can be found at https://bit.ly/drt-video-tutorial.

In the tutorial, an example is used for illustrative purposes. The example is based on the blasting process of a Chilean copper mine. A model with the activities of this process is publicly available in Appendix 4 of [13]. Blasting is one of the main methods used in the mining industry to fragment hard rock minerals [14]. It is a complex process that requires coordination of various activities between the blasting crew and other employees [15]. A detailed description of this process is included in the tutorial document.

5. Conclusion

This work proposes a tool for the visualization of process variants within a single process model, in the form of a Directed Rooted Tree. Every leaf of the DRT is the end state of a variant, which facilitates their side-to-side comparison. The above is enhanced by the inclusion of cost metrics, which are supported by a color range for the identification of cheaper and more expensive process variants.

The proposed tool is expected to serve as a complementary visualization of discovered processes when performing exploratory analyses of event logs.

Acknowledgments

This work was funded by the Agencia Nacional de Investigación y Desarrollo de Chile. Grant

numbers ANID FONDECYT 1230697 and ANID-Subdirección de Capital Humano/Doctorado Nacional/2021-21210022.

References

[1] W. Van der Aalst, Process mining: data science in action, Springer, 2016.
[2] S. J. Leemans, E. Poppe, M. T. Wynn, Directly follows-based process mining: Exploration & a case study, in: 2019 International Conference on Process Mining (ICPM), IEEE, 2019, pp. 25–32.
[3] F. Taymouri, M. La Rosa, M. Dumas, F. M. Maggi, Business process variant analysis: Survey and classification, Knowledge-Based Systems 211 (2021) 106557.
[4] R. Afshar, M. T. Goodrich, Exact learning of multitrees and almost-trees using path queries, in: LATIN 2022: Theoretical Informatics: 15th Latin American Symposium, Springer, 2022, pp. 293–311.
[5] M. Wynn, W. Z. Low, A. ter Hofstede, W. Nauta, A framework for cost-aware process management: cost reporting and cost prediction, Journal of Universal Computer Science 20 (2014) 406–430.
[6] R. Cao, W. Ni, Q. Zeng, F. Lu, C. Liu, H. Duan, Remaining time prediction for business processes with concurrency based on log representation, China Communications 18 (2021) 76–91.
[7] H.-N. Bui, T.-T. Nguyen, T.-C. Nguyen, Q.-T. Ha, A new trace clustering algorithm based on context in process mining, in: Rough Sets: International Joint Conference, IJCRS 2018, Springer, 2018, pp. 644–657.
[8] H. Fang, W. Liu, W. Wang, S. Zhang, Discovery of process variants based on trace context tree, Connection Science 35 (2023) 2190499.
[9] A. Agius, A. Rickabaugh, A. Kushnir, A. Scott, A. Seguin, C. Lyding, D. Parker, D. Hunn, E. Twersky, J. Kuehlers, J. Elbourn, J. Janiuk, J. Perrott, K. Kostadinov, M. Scheidegger, M. Malerba, M. Gechev, P. Gschwendtner, P. Kozlowski, S. Cotin, Z. Arend, Angular the web development framework for building the future, 2016. URL: https://angular.io/.
[10] K. Sveidqvist, Mermaid diagramming and charting tool, 2014. URL: https://mermaid.js.org/.
[11] T. Allweyer, BPMN 2.0: introduction to the standard for business process modeling, BoD–Books on Demand, 2016.
[12] M. De Leoni, F. Mannhardt, Road traffic fine management process, Eindhoven University of Technology, Dataset 284 (2015).
[13] J. E. Acuña Maldonado, Modelo de Gestión de Operaciones de la mina Rajo Abierto de la División Andina de CODELCO, Thesis, Pontificia Universidad Católica de Valparaíso, 2008.
[14] J. M. Akande, A. E. Aladejare, A. I. Lawal, Evaluation of the environmental impacts of blasting in okorusu fluorspar mine, namibia, International Journal of Engineering and Technology 4 (2014) 101–108.
[15] T. S. Bajpayee, T. Rehak, G. L. Mowrey, D. Ingram, Blasting injuries in surface mining with emphasis on flyrock and blast area security, Journal of Safety Research 35 (2004) 47–57.