



Do You Behave Always the Same?

A Process Mining Approach

Di Federico, Gemma; Burattin, Andrea

Published in:
Proceedings of Process Mining Workshop

Link to article, DOI:
[10.1007/978-3-031-27815-0_1](https://doi.org/10.1007/978-3-031-27815-0_1)

Publication date:
2023

Document Version
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

Citation (APA):
Di Federico, G., & Burattin, A. (2023). Do You Behave Always the Same? A Process Mining Approach. In *Proceedings of Process Mining Workshop: ICPM 2022 International Workshops* (Vol. 468, pp. 5-17). Springer. https://doi.org/10.1007/978-3-031-27815-0_1

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.



Do You Behave Always the Same?

A Process Mining Approach

Gemma Di Federico^(✉) and Andrea Burattin

Technical University of Denmark, Kgs. Lyngby, Denmark
gdf@dtu.dk

Abstract. Human behavior could be represented in the form of a process. Existing process modeling notations, however, are not able to faithfully represent these very flexible and unstructured processes. Additional non-process aware perspectives should be considered in the representation. Control-flow and data dimensions should be combined to build a robust model which can be used for analysis purposes. The work in this paper proposes a new hybrid model in which these dimensions are combined. An enriched conformance checking approach is described, based on the alignment of imperative and declarative process models, which also supports data dimensions from a statistical viewpoint.

1 Introduction

A process is a series of activities that are executed with the aim of achieving a specific goal. The notion of process can be used to describe most of the behaviors we adopt in our daily life. Whenever we deal with an ordered series of activities, that are performed repetitively, we can leverage the notion of *process* [10]. A process model is a formalization of a process, it abstracts activities and dependencies in a conceptual model. A process modeling language offers the set of rules and structural components to represent a process in form of a model. An example of a process is the procedure to get medications from a prescription, as well as the process that a person follows in order to get ready for work in the morning. In the former example, the procedure is strict and follows a well-defined and ordered set of activities; in the latter example, the process is flexible and can vary based on daily preferences, meaning that the process does not necessarily enforce a static structure. To some extent, existing process modeling languages are able to represent processes related to human behavior, however, several important aspects cannot be expressed by those languages. Dealing with human processes is challenging [8] since human beings are not forced to follow a strict procedure while executing activities, which results in high variability of the process, and the model. What is more, human behavior can be influenced by external factors, such as the environment. Modeling languages have structural limitations which restrict the expressiveness of the models they can represent. Among these is the fact that a process model primarily focuses on the control flow perspective. Consider a process executed in an environment with a temperature of 18°,

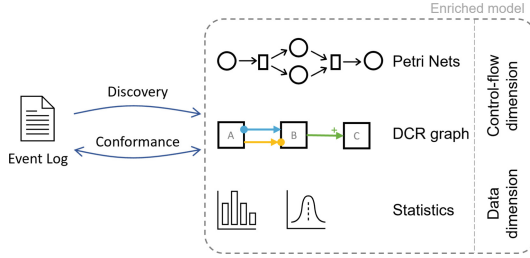


Fig. 1. Approach overview

in which a person is drinking 5 glasses of water per day. If the ambient temperature rises, the frequency of the activity “drinking” is expected to increase as well. Let’s now consider a new instance of the same process, in which a person is drinking 5 glasses of water but the temperature is 32° . Just considering a control-flow perspective, the two instances are perfectly compliant. However, combining the drinking activity with both its frequency and the environment temperature, leads to a more detailed representation of the behavior. Additionally, most of the imperative languages only allow the design of uncountable loops, while this aspect could be relevant when representing human behavior. Declarative languages only specify the workflow through the use of constraints, i.e., only the essential characteristics are described. Hence, the model over-generalizes the process, often allowing too many different behaviors. As process models are conceptual models, they actually are abstractions of reality, focusing only on the aspects that are deemed relevant. The reality can be captured by observing the actual process, resulting in a set of events collected in an event log. When trying to workshops establish a relation between a process model and the reality, in which both refer to the same process execution, it can be easily noted how far from each other they can be. Even if numerous process modeling languages exist, the control-flow and the constraints discovery (both referring to imperative and declarative processes) are not always sufficient to capture all the characteristics of some kind of process. Other dimensions must be considered and included in the analysis. Among the tasks investigated in Process Mining [6], *conformance checking* [4] assumes process models to be *prescriptive* (a.k.a. *normative*) and thus it tries to establish the extent to which executions are compliant with the reference model. Therefore, if conformance checking tasks are needed, the model should be as robust and realistic as possible.

The work presented in this paper aims at improving conformance checking techniques by extending them in such a way that the control-flow is used alongside other dimensions. As depicted in Fig. 1, we suggest a hybrid approach in which process and data dimensions are combined, and we implement an enriched conformance checking approach based on the alignment of imperative and declarative process models, which also supports data dimensions from a statistical viewpoint.

The paper is structured as follows. Section 2 presents related work and motivate the paper. In Sect. 3 the solution is presented. Evaluation and discussion is in Sect. 4. Section 5 concludes the paper and presents future work.

2 Background

2.1 State of the Art

The difference between a business process and a human-related process lies in the rigidity of the structure: human processes can be extremely flexible, involving additional perspectives [8] on top of the control-flow.

Although it is possible to define the set of activities that compose human behavior, we cannot define (or restrict) with certainty their order of execution. The reason is that activities are typically combined based on the specific cases, i.e. they are heavily case-dependent [18], and the behavior changes according to known or unknown factors, in a conscious or unconscious way [13]. Even though they share many characteristics with knowledge intensive processes [7], they have a lower degree of uncertainty. Traditional process modeling languages manifest significant limitations when applied to such unstructured processes, usually resulting in describing all possible variants [9] in form of complex and chaotic process models. A process model representing human behavior must abstract the underlying process, allowing for variability, but without over-generalizing.

In order to combine rigid and flexible parts of the models, and thus take advantage of both imperative and declarative process modeling languages [16], hybrid approaches have emerged. Hybrid models combine several process dimensions to improve the understandability of a process model and to provide a clearer semantic of the model itself. According to Andaloussi et al. [2] three process artifacts are usually combined in hybrid approaches, and are static, dynamic or interactive artifacts. Schunselaar et al. [17] propose a hybrid language which combines imperative and declarative constructs. The approach firstly derives an imperative process model (a process tree) and then the less structured parts are replaced with declarative models to improve the overall model precision. López et al., in [12], combine texts with the Dynamic Condition Response (DCR) language. The declarative model is discovered directly from text, and then a dynamic mapping between model components and text is provided. The approach aims to improve the understandability of declarative models. An interactive artifact is proposed in [14] where authors combine the static representation of a process model (DCR graph) and its execution through a simulation. The work presents a tool in which the user can interact directly with the process model. Hybrid approaches focus on the combination of a graphical representation of the process model, together with either another static component (e.g. a process model in a different notation, alongside or hierarchically integrated) or a dynamic or interactive artifact such as a simulation. Although they improve the representation of a process model, the control-flow only is not expressive enough.

Felli et al. [11] recognized the importance of enriching a process model with other perspectives, by proposing a framework to compute conformance metrics and data-aware alignments using Data Petri Nets. However, they consider, in a combined way, the control-flow and the data that the process manipulates, without considering non-process aware perspectives. In the work presented in this paper, the data dimension refers to all those attributes of the activities that

are not directly captured by process discovery algorithms, hence not represented in a process model. Without considering these additional perspectives, a process model would be too general, always leading to a successful alignment between the model and new process instances. As a result, if a new process instance varies in activities frequency or duration, it will always fit the model. In this respect, conformance checking fails in its principle.

2.2 Problem Description

Behaviour modelling is a demanding task [8]. In view of the fact that human beings have their own minds and their own interests, their behavior cannot be entirely defined *ex-ante*. There are logical reasons behind the execution of an ordered series of activities, but the way in which these activities are coordinated is not necessarily a single and unique pattern. This makes the control-flow of behaviors highly *variable*. Additionally, a considerable part of human behavior is composed of *repeatable activities*. Human beings perform a *semi-defined* set of activities every day, but part of it is repeated several times throughout the day [3]. Whenever an activity is executed, it may be part of a different set of circumstances, a.k.a. *context*.

Moreover, the *duration* of the activities is also a key factor that allows us to distinguish situations. An activity, based on its duration, can have different meanings. E.g. the *sleeping* activity executed for 8 h can be interpreted differently from the same activity executed only for 2 h. Both scenarios are represented by the

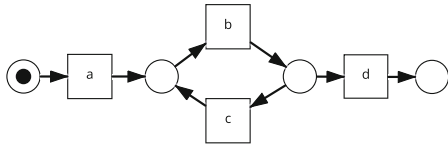


Fig. 2. WF-net derived from $L = [\langle a, b, c, b, d \rangle^2, \langle a, b, c, b, c, b, d \rangle^{10}]$

same process model, but the duration is not directly captured and encoded in the model. As a consequence, the two situations cannot be distinguished. This case can be observed in Fig. 2, in which a simple WF-Net is derived from the traces in L . From the model we cannot distinguish whether activity a was performed for one minute or for one hour. The last aspect we focus on is the *frequency* of activities. As for the duration, the frequency of occurrence of an activity can affect the meaning of the process. Although process modeling languages are capable of representing the repetitions of activities (such as loops), information on the recurrence of the frequency is not included. Loops and repetitions are therefore uncountable. For instance, from the model in Fig. 2 we can't differentiate if the loop between the activities b and c is executed one time or ten times. A trace $t = \langle a, b, c, b, c, b, c, b, c, b, d \rangle$ can perfectly be executed in the model, even though previous examples from the log show only fewer repetitions.

To tackle the above-mentioned issues, we implemented an enriched conformance checking approach, in which we provide information on the process based on different points of view, i.e. control-flow dimensions (both declarative and imperative) along with data dimensions. The work presented in this paper aims to answer the following research question:

RQ: *Does a hybrid process model help in describing human behavior, with the goal of understanding whether such behavior has changed or whether it is consistent with previous observations?*

3 Approach

A process model by itself is not always capable to faithfully capture human behavior. As introduced in the above section, several types of hybrid approaches have been developed, but they all focus only on the process dimension. Especially when dealing with human behavior, typical process representations are not enough. We therefore analyzed human behavior processes and investigated whenever the process representation does not relate to the real process. The conformance checking approach presented in this paper consists of an integrated solution that combines discovery and conformance of both a process and a data dimension. As introduced in Fig. 1, our discovery produces process models as well as a list of statistics for the activities in the event log. The models represent the control-flow perspective, while the statistics the data perspective. In this first version of the approach, the statistics focuses on three data aspects which allow to capture other dimensions of the process, and are the duration of activities, their frequency and the absolute time. The conformance checking produces an enriched fitness value that is based on the verification between each trace in the event log and the enriched discovered model. The enriched fitness value is the composition of the six fitness measures, and it is calculated according to the procedure described in the next subsections. It is important to highlight the importance of the enriched fitness value obtained by the application of the approach presented in this paper. In fact, the value does not refer only to a control-flow perspective, but takes into consideration other dimensions that are not strictly process related.

3.1 Control-Flow and Data Discovery

Control-Flow Representation and Discovery. The main challenge in behavioral modeling is to observe the process from different points of view. The first viewpoint is the control-flow perspective, which can be represented using imperative or declarative languages. Although a declarative language allows to abstract from the problem of variability, as it represents the process in form of constraints, an imperative language has a clearer and more structured representation. The two language categories have different characteristics and, based on the usage, the most appropriate one can be chosen. However, to allow the discovery and the conformance, only languages with a clear execution semantic are considered in the presented approach. The main purpose beyond this paper is that a process model representing human behavior is visually clear and representative of the process. As argued before, imperative and declarative languages have pros and cons in this task. Therefore, to avoid to restrict the final user through a specific representational direction, we decided to include both

language families in the proposed approach. In particular, the process discovery includes the *Heuristic* [19] and *Inductive Miner* [15], which produce Petri Nets, and the *DisCoveR* [15] algorithm which produces a DCR Graph.

Data Representation and Discovery. The data dimension focuses on the derivation of relevant statistics under the frequency of activities, their duration, and their occurrence time point of view. As introduced in Sect. 2.2, the *frequency* of activities is a relevant feature to discover repetitions of activities inside the process. To compute the frequency, the occurrence of each unique activity identifier is counted in each trace of the event log. Then, the frequencies are aggregated to the entire event log, and basic statistics are calculated for each activity. The statistics are the mean, the standard deviation, the median, the minimum frequency and the maximum frequency. The values computed enrich the discovered process from a frequency perspective, allowing to have information on the occurrence of each activity identifier.

The second element modelling the data perspective is the *duration*, used to investigate the duration of each activity over time. A different duration in the execution of an activity can completely change the meaning with respect to the process. The duration of the activities is calculated based on the mean duration of each unique activity identifier in each trace. Given an activity identifier, mean, median, min and max duration are calculated for each trace. The values are then aggregated to obtain more accurate results which describe the entire event log.

Always remaining in the time dimension, the *absolute time* when activities happen is another relevant factor in behavioral modeling. Even if conceptually activities are not executed at the same precise time, the absolute time is a powerful tool for identifying delays in the execution of activities. This dimension is treated by considering the histogram of how often each activity has been observed within a certain time interval (e.g., hours of the day).

3.2 Control-Flow and Data Conformance

Once the enriched model is derived, conformance checking algorithms can be used to relate the process model with instance of the process collected in an event log. The conformance checking tries to align both the control-flow and the data perspectives, producing an enriched fitness value as output.

Conformance of the Control-Flow Dimension. The enriched model is represented both in form of Petri Nets and a DCR Graph. According to these languages, the conformance checking algorithms included are the alignment [1] for the Petri Nets, and a rule checker [5] for the DCR Graph. An alignment algorithm establishes a link between a trace in the event log and a valid execution sequence of the process model. For each trace in the event log, a fitness value is obtained. The rule checker verifies if a trace violates the constraints of the graph. For each trace, a fitness value is obtained.

Conformance of the Data. While for the control-flow perspective there are conformance checking techniques available, for the data part it was necessary to investigate the most suitable ways to compare the reference data with the actual instance. For each component of the data dimension, we implement a comparison function. To verify if the frequency statistics in the enriched model conform to the event log, we will assume that activities are normally distributed. The normal distribution is used to show that values close to the mean are more frequent in occurrence than values far from the mean. Assuming that the mean value is our reference value for the frequency, by means of the computed probability density function we can interpret the likelihood that the mean frequency value (for each activity identifier) in the trace, is close to the reference. Then, we consider the likelihood as the fitness value for the frequency dimension. What is more, the frequency value under analysis has to be in the range from the minimum number of occurrences up to the maximum number of occurrences (defined in the model), otherwise a zero fitness value is returned.

The same approach explained for frequencies is used for the duration of activities. Activity durations are assumed to be normally distributed and hence the same strategy is used.

Concerning the absolute time, the approach used in the previous two cases cannot be used, primarily because the absolute time is not cumulative. E.g., we may have the same activity repeated multiple time within the same trace and therefore it might not be possible to aggregate the time of those activities. We decided to use the histogram of the frequencies of each activity over time intervals. To compute the conformance we normalize the frequencies in the interval 0–1 (where 1 indicates the most frequent time interval and 0 the least frequent time interval) and the conformance of new absolute time is then computed as the normalized frequency for the time interval the new absolute time belongs to.

The final fitness value is an aggregation of six values, that are the results of the application of conformance checking algorithms together with the results of the conformance of the statistics. Let's call \oplus the aggregation function for the individual measures, the overall conformance becomes:

$$\oplus \left(\underbrace{\text{Conf. Inductive, Conf. Heuristics, Rule check DCR}}_{\text{Control-flow dimension}}, \underbrace{\text{Stats on freq., Stats on duration, Stats on abs. time}}_{\text{Data dimension}} \right)$$

Examples of possible aggregations functions (i.e., \oplus) could be the (weighted) average, the maximum, and the minimum. The (weighted) average would be useful when all dimensions should be considered, the minimum would be a bit more restrictive as it'd require all measures to be high in order to return a high value itself. The fitness value shows how the discovered hybrid model reflects the behavior in the event log, both under a control-flow and a data dimension. By means of the enriched conformance checking approach presented in this paper, we have a powerful tool to explain and identify variations and discrepancies even under a non-process aware dimension.

4 Evaluation

The approach presented in this work aims to demonstrate that behavioral modeling cannot be represented solely by the control-flow: additional perspectives not referring to the control-flow must be considered. The evaluation conducted is based on trying different scenarios and verifying how the control-flow and the data perspectives respond to the identification of the variations. We identified three different scenarios, and we built (via simulation) a total of 8 synthetic event logs¹, with 1000 traces each. Each scenario contains a “normal” situation (the reference event log) and “anomalous situations” (the event logs used for verifying the conformance). Each scenario aims at identifying the advantages and limitations of both process and data perspectives.

4.1 Scenarios and Logs Description

Scenario 1 (S1) *Description* - The first scenario describes the night routine of a person. The idea is that a person sleeps all night but wakes up between zero and two times to go to the bathroom. *Variations* - The first variation describes a situation in which a person goes to the bathroom very frequently during the night, from four to ten times. In the second variation the person goes to the toilet a normal number of times but stays in the bathroom for a long period of time. *Objective* - The main objective of S1 is to highlight the importance of the data perspective. In fact, the variation is in the frequency and the duration, perspectives that are usually not represented on top of process models.

Scenario 2 (S2) *Description* - The second hypothetical scenario focuses on repetitive activities. The log synthesizes a day where a person eats lunch, leaves the apartment and then comes back for eating dinner, and relaxes on the couch until sleeping time. In a normal scenario, the person has lunch between 11:30 and 13:00, and dinner between 18:00 and 20:00. Both having lunch and having dinner are referred to as the activity of *eating*. *Variations* - Eating lunch or dinner outside the predefined ranges is considered an anomalous behavior. In the first variation, the person has lunch around 14:00 and dinner on time, or has lunch on time and delayed dinner between 21:30 and 23:00. The second variation skips one or both of the meals. *Objective* - The objective of S2 is to verify the behavior of the modeling languages with repetitive activities, both in terms of execution time and actual occurrence. We should be able to identify if a person is skipping meals, or if they are having delayed meals.

Scenario 3 (S3) *Description* - The last scenario describes a hypothetical morning routine: the person wakes up and has breakfast. Right after they go to the bathroom and then get dressed, ready to go out. *Variations* - In the variation the person does not follow the normal control-flow of the activities but mixes the execution of them. The process always starts with waking up but then the get dressed activity can be skipped and executed later. After that, the breakfast, bathroom, and get dressed activities can be executed in any order.

¹ All the event logs can be found at <https://doi.org/10.5281/zenodo.6632042>.

Table 1. Fitness values for control-flow perspective

Scenario		CCHeu	CCInd	CCDCR	Avg
S1 Norm vs	S1 Freq	1.00	1.00	1.00	1.00
	S1 Duration	1.00	1.00	1.00	1.00
S2 Norm vs	S2 Absence	0.75	0.75	0.00	0.50
	S2 Delay	1.00	1.00	1.00	1.00
S3 Norm vs	S3 Shuffle	0.76	0.76	0.00	0.50

Table 2. Fitness values for data perspective

Scenario		CCFreq	CCDur	CCTime	Avg
S1 Norm vs	S1 Freq	0.33	0.52	0.73	0.53
	S1 Duration	0.76	0.46	0.90	0.71
S2 Norm vs	S2 Absence	0.75	0.67	0.90	0.77
	S2 Delay	1.00	0.70	0.48	0.72
S3 Norm vs	S3 Shuffle	1.00	0.67	0.82	0.83

In the end, the person goes out. *Objective* - The purpose of S3 is to focus solely on the control-flow. In this scenario we introduce variability in the execution of activities, starting from a structured and linear situation.

4.2 Log Evaluation

The approach presented in this paper is implemented as Java and Python applications². We constructed a Python script to orchestrate the execution of all algorithms and return a final conformance value. For each scenario, the base event log is used to derive the reference model. Conformance checking is then applied on the reference model together with each variation log. The results are stored in a CSV file. The created logs aim at demonstrating that there are cases in which the control-flow cannot explain the process by itself and cases in which the statistics alone do not give a clear overview of the problem. In particular, scenario S1 focuses entirely on the data perspective, showing how frequency and duration affect the analysis. S3 highlights the importance of the control-flow perspective, while S2 combines both of them with missing activities on one hand and the delay on the other hand.

4.3 Results and Discussion

The results of the application of the approach are presented below. The values obtained are referred as: CCHeu for the alignment between the log and the Petri

² The implementation can be found at <https://doi.org/10.5281/zenodo.6631366>.

Table 3. Fitness values for control-flow and data perspective

Scenario		$\oplus = \min$	$\oplus = \text{avg}$
S1 Norm vs	S1 Freq	0.33	0.76
	S1 Duration	0.46	0.85
S2 Norm vs	S2 Absence	0.00	0.64
	S2 Delay	0.48	0.86
S3 Norm vs	S3 Shuffle	0.00	0.67

Net obtained by the Heuristic Miner, CCInd for the Inductive, and CCDCR for the rule checker of DCR. Similarly for the other measures: conformance on the frequency is CCFreq, on duration is CCDur, and on absolute time is CCTime.

To highlight the importance of the two dimensions, the results are firstly presented separately. Table 1 shows the fitness values obtained in each conformance evaluation under a control-flow perspective. Only in two cases the conformance is not perfect, that is the case of S2 Absence and S3 Shuffle. In the first one, since one activity can be skipped, the fitness value for both the Petri Nets is lowered. The fitness of the conformance with the DCR graph is zero because the constraints between *eating* and *leave* activities, and between *eating* and *relax* activities are violated when the execution of the *eating* activity is missing. In the second case instead, the order of the activities is violated. To sum up, perfect fitness values can be observed in 3 cases, while 0.5 is the average for the remaining two cases. The conclusion that can be drawn from this table is that by analyzing the processes only from the control-flow perspective, no anomaly is identified in the form of variation of frequency, duration or absolute time.

Table 2 shows all the fitness values obtained in each conformance evaluation under a data perspective. The conformance between the model from S1 and the log with frequency variation returns a fitness value of 0.33, as expected. Discrepancies also emerge in the same scenario, but in the duration variation, under the duration statistic. A significant divergence between the reference model and the actual data is observed in Scenario S2, in the delay variation, under a time perspective. In fact, the conformance of the absolute time statistic returns a low fitness value, while all the other values are optimal. By computing the average fitness for each scenario/variation, highlights the discrepancies between the data perspective and the control-flow perspective. The average values in Table 2 are much lower than the average values in Table 1.

To obtain more consistent results, all the individual values of conformance must be combined. Table 3 compares the two perspectives together, returning aggregated values in form of average and minimum for each scenario/variation. The table reveals the gap between the fitness of the control-flow dimension and the fitness of the data dimension. In almost all the scenarios, the minimum fitness value obtained (over all the perspectives) is close to zero. The total average in the table is the arithmetic mean. According to the situation at hand, other aggregation functions might also be used (e.g. by using a weighted mean, thus

providing different weights for different aspects). In the case of this experiment, none of the logs evaluated returned a perfect fitness value as instead observed in Table 1, where the focus was only on the control-flow.

Based on the results shown in Table 3 we can conclude that, while individual dimensions might show perfect fitness by themselves, even when the logs should not be explainable by the model (cf. both Table 1 and Table 2); a hybrid approach is instead always able to discriminate non-compliant behavior (observable by having no entries with value 1 in Table 3), even when different aggregation functions are used. Therefore, the research question stated in Sect. 2 can be positively answered.

4.4 Limitations

Although the evaluation pointed out promising results, there are several limitations. The first aspect to consider regards the statistics: the statistics on the duration assume a normal probability distribution. Remaining on the perspective of the accuracy of time, the histogram used in the absolute time statistics is calculated by aggregating the executions per hour. Hence, if an activity is delayed but still within the same hour (with respect to the reference model), the fitness is not affected. Finally, choosing a proper aggregation function might not be trivial. In fact, the enriched conformance checking proposed should include a tuning function capable of balancing all the dimensions.

5 Conclusions and Future Work

In order to deal with human behavioral, and in particular, in order to understand whether the behavior is compliant with a normative model, new conformance checking techniques are needed. The control-flow is not enough and it does not provide all information needed for the application of conformance checking techniques when dealing with human behavior. The process must be analyzed from different point of view: the control-flow perspective and the data perspective. The method proposed in this paper produces an enriched fitness value that balances control-flow alignment and data statistics. The control-flow alignments investigates whether the order of the activities is compliant with expectations, whereas the statistics focus on the activity frequency, activity duration, and absolute time. By creating synthetic event logs, we have demonstrated that the application of this methodology allows the identification of variations and discrepancies between a reference model and an event log where the typical conformance techniques were failing. In a previous work (see [8]), we identified all the requirements that a process modeling language must fulfill in order to represent human behavior. These requirements have been used to identify the two perspectives to include in the hybrid model. To reply the research question introduced in Sect. 2.2, taking advantage from the evaluation conducted in this paper, especially from the results in Table 3, it emerged that to properly verify the conformance of a process representing human behavior, a hybrid process model is

needed. The first step as a future work, is to refine the statistics, such as the duration, and evaluate other perspectives to be included. After that, we would like to combine the two dimensions together from a semantic point of view.

Acknowledgements. We would like to thank Anton Freyr Arnarsson and Yinggang Mi who helped with the formulation of some of the ideas of this paper.

References

1. van der Aalst, W., Adriansyah, A., van Dongen, B.: Replaying history on process models for conformance checking and performance analysis. *Wiley Interdiscip. Rev.: Data Min. Knowl. Discov.* **2**(2), 182–192 (2012)
2. Andaloussi, A.A., Burattin, A., Slaats, T., Kindler, E., Weber, B.: On the declarative paradigm in hybrid business process representations: a conceptual framework and a systematic literature study. *Inf. Syst.* **91**, 101505 (2020)
3. Banovic, N., Buzali, T., Chevalier, F., Mankoff, J., Dey, A.K.: Modeling and understanding human routine behavior. In: *Proceedings of the CHI Conference*, pp. 248–260 (2016)
4. Carmona, J., van Dongen, B., Solti, A., Weidlich, M.: *Conformance Checking*. Springer, Cham (2018)
5. Debois, S., Hildebrandt, T.T., Laursen, P.H., Ulrik, K.R.: Declarative process mining for DCR graphs. In: *Proceedings of SAC*, pp. 759–764 (2017)
6. van Der Aalst, W.: *Process Mining: Data Science in Action*, 2nd edn. Springer, Heidelberg (2016). <https://doi.org/10.1007/978-3-662-49851-4>
7. Di Ciccio, C., Marrella, A., Russo, A.: Knowledge-intensive processes: characteristics, requirements and analysis of contemporary approaches. *J. Data Semant.* **4**(1), 29–57 (2015)
8. Di Federico, G., Burattin, A., Montali, M.: Human behavior as a process model: which language to use? In: *Italian Forum on BPM*, pp. 18–25. *CEUR-WS* (2021)
9. Diamantini, C., Genga, L., Potena, D.: Behavioral process mining for unstructured processes. *J. Intell. Inf. Syst.* **47**(1), 5–32 (2016). <https://doi.org/10.1007/s10844-016-0394-7>
10. Dumas, M., La Rosa, M., Mendling, J., Reijers, H.A.: *Introduction to business process management*. In: *Fundamentals of Business Process Management*, pp. 1–33. Springer, Heidelberg (2018). https://doi.org/10.1007/978-3-662-56509-4_1
11. Felli, P., Gianola, A., Montali, M., Rivkin, A., Winkler, S.: CoCoMoT: conformance checking of multi-perspective processes via SMT. In: Polyvyanyy, A., Wynn, M.T., Van Looy, A., Reichert, M. (eds.) *BPM 2021*. LNCS, vol. 12875, pp. 217–234. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-85469-0_15
12. López, H.A., Debois, S., Hildebrandt, T.T., Marquard, M.: The process highlighter: from texts to declarative processes and back. In: *BPM (Dissertation/Demos/Industry)* vol. 2196, pp. 66–70 (2018)
13. Lull, J.J., Bayo, J.L., Shirali, M., Ghassemian, M., Fernandez-Llatas, C.: Interactive process mining in IoT and human behaviour modelling. In: Fernandez-Llatas, C. (ed.) *Interactive Process Mining in Healthcare*. HI, pp. 217–231. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-53993-1_13
14. Marquard, M., Shahzad, M., Slaats, T.: Web-based modelling and collaborative simulation of declarative processes. In: Motahari-Nezhad, H.R., Recker, J., Weidlich, M. (eds.) *BPM 2015*. LNCS, vol. 9253, pp. 209–225. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23063-4_15

15. Nekrasaite, V., Parli, A.T., Back, C.O., Slaats, T.: Discovering responsibilities with dynamic condition response graphs. In: Giorgini, P., Weber, B. (eds.) CAiSE 2019. LNCS, vol. 11483, pp. 595–610. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-21290-2_37
16. Pichler, P., Weber, B., Zugal, S., Pinggera, J., Mendling, J., Reijers, H.A.: Imperative versus declarative process modeling languages: an empirical investigation. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) BPM 2011. LNBIP, vol. 99, pp. 383–394. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-28108-2_37
17. Schunselaar, D.M.M., Slaats, T., Maggi, F.M., Reijers, H.A., van der Aalst, W.M.P.: Mining hybrid business process models: a quest for better precision. In: Abramowicz, W., Paschke, A. (eds.) BIS 2018. LNBIP, vol. 320, pp. 190–205. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93931-5_14
18. Stefanini, A., Aloini, D., Benevento, E., Dulmin, R., Mininno, V.: A process mining methodology for modeling unstructured processes. *Knowl. Process. Manag.* **27**(4), 294–310 (2020)
19. Weijters, A., van Der Aalst, W.M., De Medeiros, A.A.: Process mining with the heuristics miner-algorithm. TU/e, Technical report WP 166 (2017), 1–34 (2006)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

