Exploring the Conformance Space (Extended Abstract)

Burattin, Andrea; Gianola, Alessandro ; López, Hugo A. ; Montali, Marco

Published in:
Proceedings of the 1st Italian Forum on Business Process Management

Publication date:
2021

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

Citation (APA):
Burattin, A., Gianola, A., López, H. A., & Montali, M. (2021). Exploring the Conformance Space (Extended Abstract). In *Proceedings of the 1st Italian Forum on Business Process Management* CEUR-WS.

Exploring the Conformance Space (Extended Abstract)*

Andrea Burattin[1], Alessandro Gianola[2], Hugo A. López[3], and
Marco Montali[2]

[1] Technical University of Denmark, Kgs. Lyngby (Denmark)
[2] Free University of Bozen-Bolzano, Bolzano (Italy)
[3] The University of Copenhagen, Copenhagen (Denmark)

Abstract. In this paper we highlight a yet unexplored dimension in BPM in general, and conformance checking, in particular. While a breadth of research has focused in differentiating conforming versus non-conforming process executions, little it is known about the difference between different conforming traces. This is of particular importance in the light of the fact that not all conforming executions are equal: some achieve the intended outcome better than others. In this position paper, we demonstrate why existing model-driven and data-driven techniques are not enough to attack the problem in its full generality, and that novel methods to combine them in ways that have not yet been explored are essential within BPM.

1 Un-flattening Conforming Traces

A large share of research within the BPM spectrum focuses on distinguishing conforming vs non-conforming executions. While most works brought focus on non-conforming traces, little research has focused on understanding the different nature of conforming executions. Conforming executions do not only differ in terms of the sequence of performed activities but also in terms of the achieved outcome. "Outcome" is to be interpreted here in a broad sense and acts as the umbrella term for a measurable goal/reward/desired state of affairs.

There is a need to characterise conforming executions already observed as well as potential new ones (i.e., executions still to happen, but possible) with the ultimate goal of handling operational aspects of the process, thus going beyond model-driven approaches and data-driven ones.

2 Impact and Case Studies

This problem is essential in BPM: processes are introduced to regulate how internal work is carried out within an organisation towards achieving the desired

---

outcome. However, processes need to adapt to many adverse situations, leading to executions with final states that are nominally conforming but do not achieve the outcome in the best possible way. This differentiation is not present in contemporary process modelling notations, nor reflected in the recorded event data, and thus it prevents the possibility of devising operational techniques for supporting decision-makers in how to best continue with a given execution.

In the rest of this section, we present several case studies together with how these problems might be partially solved with state of the art techniques.

Use case 1. Let us consider this scenario: a delivery company, which has a fleet to distribute packages and parcels ordered online to end-users. In order to optimise the routing of the deliveries, a system depicts the exact sequence of packages to be distributed, for instance, by solving an instance of the travelling salesman problem. The route then constitutes each driver's workflow. Once the routes are dispatched to each driver unit, they can start their processes execution (i.e., their deliveries). Each driver is rewarded with a bonus based on the number of deliveries per time unit and punished if they do not deliver. It is possible to distinguish two aspects of the delivery processes: on the one hand, each driver is supposed to deliver all packages, on the other hand, drivers are meant to follow the given order to optimise the journey. The system includes an *orchestrator*: an application that suggests the driver with actions during a delivery.

How can we dynamically optimize deliveries so we take into account the driver's rewards as well customer's objectives? For example, imagine that on the first delivery location, it becomes possible to deliver another package. Should the driver use this opportunity? It seems reasonable to say yes but this would cause a non-conforming situation in the predefined process (i.e., the optimal plan) which could result in missing the delivery target for the following time unit.

A second situation is possible: let us imagine that one recipient of a package is not available and, after calling her, she will be able to collect the parcel in 30 minutes. How should the driver react? One option is to reschedule the delivery for another time, and other is to wait for the recipient, thus probably delaying the rest of the deliveries scheduled. While both solutions might be conforming, how is it possible to capture and reason about these situations/strategies "statically"?

One may be tempted to use game-theoretic frameworks, determining what should be the best decisions for each of the players based on a utility function. However, finding an equilibrium represents a challenge. As Woolridge states [1], a normative application of game theory requires clarity of a set of assumptions: everybody knows how the process runs, who the agents are, their choices and their preferences at hand, and everybody knows that everybody knows the game. This is not clear in our case. From the drivers' point of view, both the orchestrator and the customer may be cooperative or adversarial agents [2], thus his strategy will need to be modified in either case. The recommendations given by the orchestrator will then be seen as a *black box*. Having a model that can describe both the process, the existing alternative paths *and* their impact on the utility function will be key for the driver to make informed decisions.

As an alternative, we might apply reasoning based on goal modelling [3] considering multiple agents with potentially conflicting policies and interests, as investigated in multi-objective optimization and game-theory. However, these are typically agnostic to the control flow which makes them not ideal. Integrated approaches typically lack full coverage of process-related constructs.

A third attempt involves providing history-based recommendations in a reinforcement-learning style [4]. The absence of hard constraints from a model means that the driver might never wait for the customer as this will reduce his utility function. Similar limitations hold for recommendation-based systems [5].

Another way to tackle the problem is by using conformance checking techniques [6], including time [7] and data-aware algorithms [8, 9, 10], however, all these techniques focus on the separation of traces that are conforming from those that are not, without considering how to optimise (conforming) executions.

Use case 2. A common scenario in casework is the layout of all possible procedures encapsulated in legislation. Imagine the procedure of releasing construction permits in a municipality. Laws define conditions that a caseworker will use to determine whether a permit should be accepted, rejected, or investigated further. From the viewpoint of the caseworker, it is faster to reject cases instead of sending them back, but from the viewpoint of the citizens and the municipality, other parameters might play a factor in this decision. The municipality might have the political compromise of being attractive by renovating the city, which amounts to showing that there is a continuous acceptance rate of renovation projects per month. This will not be possible if all the cases are rejected.

A couple of approaches could be explored for this use case, all with limitations. First, one could consider a strategic alignment between organizational goals and business processes, which has been at the core of the BPM literature [11, 12]. However, these alignments map goals over single traces, while the goals of the municipality are described in terms of aggregation of cases. It is not clear how one can provide an operational description of outcomes that reflects at a suitable granularity the different quantitative and qualitative factors, and that can be directly measured on observed executions (in turn to distinguish them).

It might also be relevant to observe that, within the discussed scenario, all executions are typically compliant. The difference does not lie in conforming versus non-conforming traces. An initial attempt to quantify the amount of compliant behaviours has been proposed in the literature [13]. This, however, does not tackle the relationships between compliant executions and intended outcomes.

Use case 3. Let us imagine a hypothetical incident involving a container ship that, due to a series of unfortunate events, found itself obstructing a very important waterway (e.g., the Suez Canal). All shipping companies whose routes involve traversing the channel need to address the situation: either waiting for the container ship to be tug away, or take a different route. Both situations involve a violation of conformance and therefore they will result in a non-conforming execution. This non-conformance, however, stems from conditions that are not in control of the process owner and are not avoidable.

This scenario cannot be handled with data-driven techniques, since the situation is unprecedented and no historical executions are available. As such, operational decision support or reinforcement learning become useless. In turn, game-theoretic or goal modelling might be hard to setup and enforce as they require knowledge and control over an external and uncontrollable actor.

Conformance checking techniques will neither be useful since reference models will be violated. The actual reason for the violation, though, stems from the *temporary observation* of the model (i.e., it is physically not possible to cross the canal).

3 Challenges Connected to the Research

Representing and evaluating outcomes While there is extensive literature on representing goals in requirements engineering and conceptual modelling, it is not clear how one can provide an operational description of outcomes that reflects at a suitable granularity the different quantitative and qualitative factors, and that can be directly measured on observed executions.

Returning to our use case #2, the goals of a caseworker can be instance-specific, such as reaching an end state, or the throughput (# of cases over time). The citizen has the goal of getting its request accepted (reaching a certain state) as fast as possible. For the municipality, the goal is to minimise running costs (throughput), while at the same time, show the political compromise on the urban development of the city, observed in a certain number of accepted requests.

Outcome-aware process models Goal and process models are typically studied in isolation, without properly interconnecting them towards tightly integrated outcome-aware process models, with some interesting exceptions that require indeed further investigation [14]. This integration is necessary if one wants to understand how different executions conforming to the model relate to the outcome. It is important to point out that outcomes may exist in a different nature compared to processes: while processes typically live in a discrete space, where actions are finite and happen at precise time windows, outcomes might be observed in continuous domains. Another source of the challenge lies in the distinction between instance-specific outcomes and global process outcome, as well as in the presence of multiple actors who may have different ways of measuring such outcome: the dependency between these aspects is extremely challenging.

In the first use case, while the system gave an optimal goal-driven solution to the delivery problem, unexpected events at runtime bring deviations to what is considered an “optimal plan”. Rescheduling might keep the process conforming (it continues with the old, optimal route), while not achieving the goal of delivering all packages. The other setting, where it might be possible to deliver a parcel at a destination that was not the planned one, might fulfil the goal of delivering all packages while violating the conformity of the optimal route.

Combination of data- and model-driven techniques Operational techniques that suggest what to do next towards achieving the desired outcome in the best way possible need to account for the observed executions and a given outcome-aware process model. On the one hand, data-driven techniques such as recommender systems are unable to distinguish conforming from non-conforming executions, which is crucial as suggestions on how to proceed should only be based on conforming executions. In addition, they do not include in their analysis those executions that are conforming to the model but have never been observed. On the other hand, model-driven techniques need to incorporate into the model itself a plethora of sophisticated, overly detailed cases and exceptions, which would be automatically detected and classified by using data-driven, learning-based approaches. A faithful representation of the process is then in the hands of the modeller.

4 Conclusions and Outlook

We believe the problem cannot be tackled by directly defining a single, one-fits-all technical solution. In this spirit, we reflect on the two key directions we are investigating to attack it.

First and foremost, we think it is important, before even attempting at a solution, to develop a foundational framework to describe the problem in its full generality, spelling out its different dimensions. This will provide the basis to understand whether the resulting configuration of the problem can be attacked with existing techniques, or requires genuine new research. As for the latter, we see as central the need of combining model-driven and data-driven techniques (such as formal methods for strategic reasoning and adversarial reinforcement learning), taking advantage of the fact that the problem is framed in a concrete setting, with specific BPM-related features.

A second direction refers to the need of enriching existing modelling languages. Here, we are aiming at a conservative and minimalistic approach capturing key aspects such as hard/soft constraints, stochasticity, quantitative and qualitative outcomes. Here is particularly important the understandability of the models, as well as the possibility of discovering the different aspects from data.

Acknowledgements

This research has been partially supported by the UNIBZ projects VERBA and DACOMAN and the EcoKnow project.

References

1. Wooldridge, M.: Does game theory work? IEEE Intelligent Systems **27**(6) (2012)
2. Nash, J.: Non-cooperative games. Ann. Math. **54**(2) (1951) 286–295

3. Liaskos, S., McIlraith, S.A., Sohrabi, S., Mylopoulos, J.: Representing and reasoning about preferences in requirements engineering. Requir. Eng. **16**(3) (2011)
4. Sutton, R.S., Barto, A.G.: Reinforcement learning - an introduction. Second edn. Adaptive computation and machine learning. MIT Press (2018)
5. Teinemaa, I., Dumas, M., Rosa, M.L., Maggi, F.M.: Outcome-oriented predictive process monitoring: Review and benchmark. ACM Transactions on Knowledge Discovery from Data (TKDD) **13**(2) (2019) 1–57
6. Carmona, J., van Dongen, B., Solti, A., Weidlich, M.: Conformance checking. Springer (2018)
7. Zavatteri, M., Combi, C., Posenato, R., Viganò, L.: Weak, strong and dynamic controllability of access-controlled workflows under conditional uncertainty. In: BPM 2017, Springer (2017) 235–251
8. Burattin, A., Maggi, F.M., Sperduti, A.: Conformance checking based on multi-perspective declarative process models. Expert Syst. Appl. **65** (2016) 194–211
9. de Leoni, M., Muñoz-Gama, J., Carmona, J., van der Aalst, W.M.: Decomposing alignment-based conformance checking of data-aware process models. In: OTM Conferences 2014, Springer (2014) 3–20
10. Felli, P., Gianola, A., Montali, M., Rivkin, A., Winkler, S.: CoCoMoT: Conformance checking of multi-perspective processes via SMT. In: BPM 2021, Springer (2021) 217–234
11. Guizzardi, R., Reis, A.N. In: A Method to Align Goals and Business Processes. Volume 9381. Springer International Publishing (2015) 79–93
12. Koliadis, G., Vranesevic, A., Bhuiyan, M., Krishna, A., Ghose, A. In: Combining i* and BPMN for Business Process Model Lifecycle Management. Volume 4103 of LNCS. Springer Berlin Heidelberg (2006) 416–427
13. Burattin, A., Guizzardi, G., Maggi, F.M., Montali, M.: Fifty shades of green: How informative is a compliant process trace? In: CAiSE 2019, Springer (2019) 611–626
14. Dees, M., de Leoni, M., Mannhardt, F.: Enhancing process models to improve business performance: A methodology and case studies. In: OTM Conferences 2017, Springer (2017) 232–251