Structural and Behavioural Commonalities of Process Variants

Matthias Weidlich and Mathias Weske

Hasso-Plattner-Institute, University of Potsdam, Germany
{matthias.weidlich,weske}@hpi.uni-potsdam.de

Abstract. A common business process might exist in multiple variations in an enterprise, due to different legal requirements in different countries, deviations in the supporting IT infrastructure, or differences in the organisational structure. In order to explore and control such variability, we argue that the notion of a core process, the invariant nucleus of all process variants, might be applied. In this paper, we discuss the spectrum of structural and behavioural aspects that might be leveraged to define such a process.

1 Introduction

In large enterprises, it can be observed that a common business processes exists in many variations across different parts of the organisation. These variations may stem from different legal requirements in different countries, deviations in the supporting IT infrastructure, or differences in the organisational structure. Consequently, the existence of such *process variants* is often inevitable and needed to meet the concerns of a certain organisational unit. Still, excessive generation of variations of a common business process is a serious obstacle for enterprise-wide process initiatives, such as the roll-out of new IT infrastructure or an organisational restructuring. Thus, effective management of process variations is crucial for keeping the deviations within reasonable limits.

The creation of process variants might be controlled in the way that a base process is adapted following on well-defined change patterns (cf., [1, 2]) or configurations (cf., [3]). However, enforcement of such a controlled variability of processes is hard to achieve once process variants are created independently in different countries or organisational units. Addressing such scenarios of decoupled process variants, we focus on the use case of *exploration of process variability*. Given a set of independently created process variants, such an analysis assumes the knowledge about corresponding activities of these variants and tries to answer the following questions: Are there outliers, i.e., process variants that deviate significantly from the other variants? Are there redundant process variants, i.e., very similar variants that might be merged? Obviously, both questions are relative to a notion of commonality of process variants. We refer to this commonality as the *core process*, the invariant nucleus of all variants. Given the core process, the deviation between it and a dedicated variant is quantified. Based thereon, conclusions can be drawn regarding outliers and redundant process variants.

The notion of a core process is not new, but has been advocated under the term *greatest common divisor* (GCD) of a set of process models for solving other *'notorious problems'* [4]. For instance, the adaptation of a standard software solutions to the needs of an enterprise might be controlled by the GCD of the organisational processes and the processes implemented by an IT system. This use case imposes rather strict requirements, such that the proposed notion of a GCD is based on behaviour inheritance [4]. While this seems to be a reasonable choice for the aforementioned use case, process variants that are rather loosely coupled often do not have to comply to such a strict relation. Therefore, we aim at a framework that allows for flexibility in the definition of a core process. Starting with a very weak notion of a core process, the notion can be strictened in a step-wise approach in the course of variability exploration.

In this paper, we take the work on similarity of process models as a starting point. Recently, various structural and behavioural abstractions of process models have been proposed in order to search for process models in repositories [5, 6] or quantify process model consistency [7]. We show that the spectrum of these abstractions allows for a multitude of definitions of a core process. That allows for a fine-granular analysis of the degree of variability in a set of process variants.

The remainder of this paper is structured as follows. Section 2 introduces the spectrum of structural and behavioural abstractions that might be leveraged for the definition of a core process. Subsequently, Section 3 illustrates their application for quantifying variability. Section 4 reviews related work, while Section 5 concludes the paper.

2 Characterisation of a Core Process

In this section, we approach the question of how a core process can be characterised in terms of structure and behaviour given a set of process variants. Apparently, process variants might also differ with respect to the labelling of the process elements and the assumed level of abstraction. That is, corresponding activities have to be identified and refinements of activities have to be treated accordingly. However, we abstract from these aspects in the remainder of this paper and focus on the structural and behavioural dimension. Note that we also restrict our discussion to rather generic process models consisting of activities and means for routing the control flow. Therefore, we focus on the commonalities of most process modelling languages. For illustration purposes, we use the Business Process Modeling Notation (BPMN).

First, Section 2.1 discusses the structural aspects of any characterisation of a core process. Second, we elaborate on behavioural aspects in Section 2.2.

2.1 The Structural Dimension

As activities are the basic functional blocks of a process model, we argue that any characterisation of a core process should be guided by the activities. Thus, activities are selected for containment in the core process based on a certain

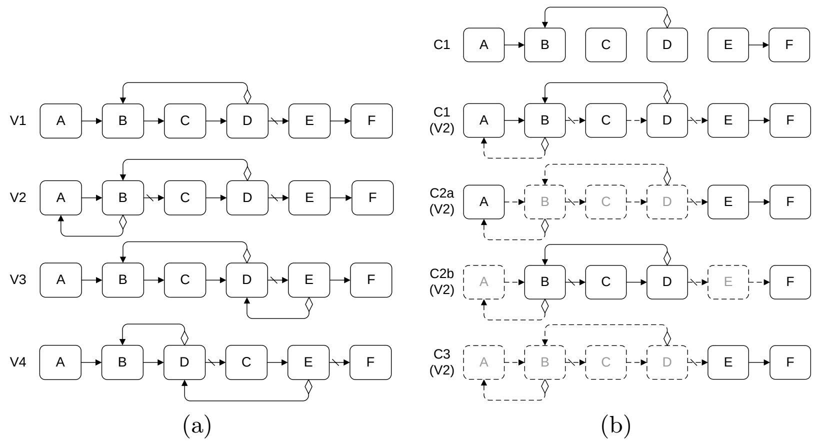


Fig. 1. (a) set of process variants, (b) different structural characterisations of a core process for the variants of (a) along with their projection on variant V2

criterion. Still, paths can be considered in the core process definition, such that all direct paths between selected activities (i.e., there is no other selected activity in between) shared by all process variants should also appear in the core process.

In order to select activities for the core process definition, we might leverage the following structural aspects.

Activity Occurrence. In the most simple case, activities of the core process are those that are contained in every process variant. However, one might also think of a relaxed interpretation that requires activities of the core process to appear in a certain share of the process variants. For instance, activities contained in at least 70% of the process variants qualify for being part of the core process.

Path Existence. If a first set of activities has been selected, the existence of paths between these activities can be used an additional filter for the selection. The rationale behind is that activities should occur in the core process, only if their dependencies in terms of the presence or absence of a path between them is consistent throughout the set of process variants. Again, the criterion might be relaxed in the sense that the path dependency is required to hold solely in the majority or a certain share of the process variants.

Subgraph Isomorphism. Selecting activities for a core process when considering the existence of paths between them might still neglect structural differences that appear throughout the process variants. Therefore, not only the presence or absence of paths, but a strict structural criterion in the sense of a graph isomorphism might be applied. Thus, we select the activities that are part of a subgraph for which there is an isomorphism across all process variants.

We illustrate these structural aspects along with their implications for the definition of a core process using the four process variants depicted in Fig. 1(a). Note that all variants share the same set of activities, but show different execution dependencies. Besides normal sequence flow, the models contain conditional flow (arc with a small diamond) that is activated based on a condition, and default flow (arc with a slash marker) that is activated if none of the other outgoing flows can be taken. A first structural characterisation of a core process might be based on the activities and paths that all variants have in common. The model C1 in Fig. 1(b) illustrates this case. Note that the resulting model is not a complete process model, but rather a set of structural aspects that is invariant across all process variants. In order to illustrate the deviation of a certain process variant, we also depicted the projection of the core process C1 on the process variant V2 in Fig. 1(b) (dashed lines highlight parts that are not contained in the core process). Applying the aforementioned criterion of path existence results in a different characterisation of a core process. In particular, there are two core process definitions, C2a and C2b, both illustrated as projections on process variant V2 in Fig. 1(b). Both clusters of activities, {A, E, F} and {B, C, D, F}, satisfy the requirement of a consistent path existence or absence between the respective activities, while there is no subsumption relation between both sets. Note that multiple definitions of a core process (or GCD, respectively) have also been observed in [4]. Finally, core process C3 illustrates the case that a graph isomorphism is required to hold between the respective activities in all process variants. Obviously, trivial core process definitions containing just one activity would also satisfy this requirement.

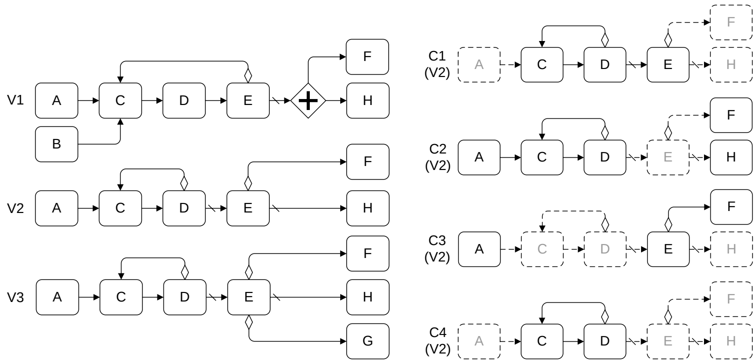
Note that the structural aspects that might characterise a core process differ significantly regarding their computational complexity. While the pure occurrence of an activity can be checked in a straight-forward manner, the path existence criterion implies additional overhead. Here, computation of the transitive closure of the flow relation requires already low polynomial time. Moreover, the subgraph isomorphism problem is known to be NP-complete.

2.2 The Behavioural Dimension

Activities that are part of the core process may be required to show the same behavioural dependencies in all process variants. As in case of the structural dimension, there are various interpretations of the notion of *equal behavioural dependencies*. In the following paragraphs, we discuss four interpretations, i.e., behavioural aspects that can be utilised to characterise a core process.

Optionality of Execution. Evidently, the fact whether a certain activity is optional or mandatory for the completion of the process can be considered. That is, solely activities that are mandatory for completion in all process variants qualify to be part of the core process. As discussed for structural, this criterion might be relaxed, such that activities of a core process have to be mandatory in the majority (or a certain share) of the process variants.

Causal Footprints. In order to take constraints regarding the order of execution into account, causal footprints can be used as a behavioural abstraction. They have been introduced for verification of processes [8] and later been applied as a similarity measure [6]. The basic idea of causal footprints is to capture the behaviour of a process by means of two relations between activities. So called *look-back links* associate a set of activities to an activity, such that any execution of the latter activity is preceded by at least one of the sources of the link. Similarly,



(a) (b)

Fig. 2. (a) set of process variants, (b) different behavioural characterisations of a core process for the variants of (a) along with their projection on variant V2

look-ahead links relate an activity to a set of activities, such that an execution of the former activity is followed by at least one of the targets of the link. Starting with neighbouring activities, the causal closure of these links can be computed. In our setting, we can leverage causal footprints as follows. In order to qualify for containment in the core process, any set of activities has to show a consistent causal footprint throughout all process variants.

(Causal) Behavioural Profiles. A similar behavioural abstraction are behavioural profiles that have been introduced to quantify the consistency of model correspondences [7]. Such a profile comprises relations between pairs of activities based on the observable behaviour. That is, two activities are considered to be exclusive, in strict order, or in observation concurrency to each other. Together with reversed strict order, these relations partition the Cartesian product of activities. While the behavioural profile focussed on the *order of potential execution*, the *causal* behavioural profile takes optionality of activity execution and, therefore, also causality between activities into account. That is, an asymmetric co-occurrence relation is defined over pairs of activities An activity is co-occurring with another activity, if any complete trace of the process containing the first activity contains the second one as well. Both behavioural abstractions (causal and non-causal behavioural profile) can be applied as discussed for causal footprints, i.e., activities in the core process must be consistent w.r.t. to the behavioural abstraction throughout all process variants.

Behaviour Equivalence. The multitude of behavioural equivalences in the linear-time – branching-time spectrum [9] can also be leveraged. That is, activities of the core process are required to be, for instance, trace equivalent throughout the set of process variants. To this end, activities that are contained in some process variants only, might either be blocked or hidden yielding the well-known notions of behavioural inheritance for trace semantics, i.e., protocol inheritance (blocked) or projection inheritance (hidden) [10, 11].

Again, we illustrate the different behavioural characterisations of a core process using exemplary process variants as depicted in Fig. 2(a). First and

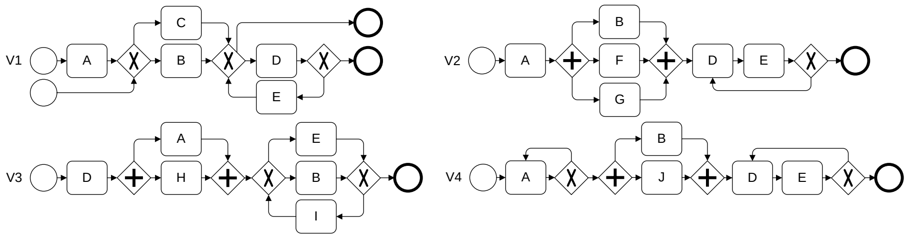


Fig. 3. Example process variants used for analysis

foremost, only common activities that are mandatory for the completion in all process variants are selected for the core process C1, which is depicted as the projection on process variant V2 in Fig. 2(b). Second, we selected the activities of the core process C2, such that their causal footprints are invariant across all process variants. Here, C2 reflects only one definition of the core process, as the set {A, E, F, H} of activities would also satisfy the respective property. For the case of behavioural profiles as the applied behavioural abstraction, there are multiple core processes as well, one depicted as model C3. Finally, C4 shows the only core process that is derived when trace equivalence is applied in the sense of projection inheritance. Protocol inheritance, in turn, would result in a core process that contains the activities {A, C, D}.

Regarding the computational complexity implied by the discussed behavioural aspects, general conclusions can hardly be drawn as it highly depends on the model characteristics. For instance, the optionality of activity execution might be determined for a block-structured model with a dedicated process start and process end quite efficiently by traversing the containment hierarchy of blocks and checking their respective type. However, we are not aware of any efficient approach to check optionality of execution for an arbitrarily structured model. Causal footprints, in turn, can be enriched step-wise. One causal footprint (such a footprint is not unique) can be derived efficiently using the algorithm proposed in [8]. Behavioural profiles might be derived in low polynomial time for sound free-choice models. The same holds true for the causal behavioural profiles as long as certain requirements are met for unstructured regions of the process model, cf., [7]. Behaviour equivalences are known to be exponentially hard in the general case.

3 Application of a Core Process for Analysis

We illustrate how the different characterisations of a core process can be applied for analysis by means of the example process variants depicted in Fig. 3. First, we apply a structural core process definition that is based on the occurrence of activities and the path existence criterion that has to be consistent for 70% of the process variants. That yields a core process definition that comprises four activities (A, B, D, E) and path relations from A to {B, C, E}, B to {C, E}, C to E, and E to C. Based thereon, we quantify the deviation from the core process for each process variant.

- The share of activities that are part of core process: $\frac{4}{5} = 0.8$ for V1, $\frac{2}{3} \approx 0.67$ for V2 and V3, and 0.8 for V4.
- The share of path relations for these activities (neglecting paths between an activity and itself) that is consistent with the core process: $\frac{12}{12} = 1$ for V1, V2, and V4, and $\frac{6}{12} = 0.5$ for V3.

Although not observable when looking solely at the overlap of activities, process variant V3 clearly shows a strong structural deviation when the path relation is also considered. Obviously, the question whether such an 'outlier' should be allowed for can only be judged by taking the business semantics into account. Assuming that the variants show a process implementation in four different countries, such a strong deviation might be explained, for instance, by the used technical infrastructures, such that V1, V2, and V4 are implemented on similar systems, whereas variant V3 is not. Still, identification of such structural deviation can trigger according actions for the respective variants.

Focussing on the three structurally similar variants V1, V2, and V4, we refine the analysis with a behavioural characterisation of the core process. That is, the core process is characterised by the four activities shared by these variants along with the relations of the (causal) behavioural profile for the Cartesian product of activities. To this end, the core process contains the relation that most process variants have in common. Behavioural deviation is quantified as follows.

- The relations of the behavioural profile that are consistent with the core process: $\frac{16}{16} = 1$ for V1 and V2, and $\frac{15}{16} \approx 0.94$ for V4.
- The relations of the causal behavioural profile that are consistent with the core process: $\frac{21}{32} \approx 0.66$ for V1, $\frac{32}{32} = 1$ for V2, and $\frac{31}{32} \approx 0.97$ for V4.

We see that the constraints w.r.t. the order of potential execution as defined by the behavioural profile are rather consistent throughout the three variants. When the constraints on causality of activity execution as defined by the causal behavioural profile are taken into account, we derive a different result. That is, V1 differs from V2 and V4, which, in turn, show nearly no behavioural deviation. Again, conclusions can only be drawn if the business semantics of these processes are considered. However, the high degree of behavioural similarity between V2 and V4 hints at potentially redundant variants. Thus, we might decide to merge both variants, such that the overall number of process variants is decreased.

4 Related Work

The question of a core process, i.e., the GCD of a set of process models, is closely related to the notion of the *least common multiple* (LCD) of a set of process models. That is, a process model might be derived that subsumes the behaviour of all process variants, cf., [12]. Such a model might be derived by mining process variants [13]. As mentioned above, LCDs can also be applied as reference models that allow for the derivation of process variants via configuration [3].

Further on, the work in [14] illustrates how structural and behavioural similarity of process variants can be leveraged to identify preferred variants, i.e., process variants that are observed frequently in an execution environment.

5 Conclusion

In this paper, we argued that the degree of variability of process variants might be explored using the notion of a core process. Such a process captures structural and behavioural aspects that are invariant across all process variants. We discussed the spectrum of structural and behavioural aspects that can be leveraged for a core process definition and sketched its application.

In future work, we aim to extend the presented ideas towards a framework that allows for a flexible analysis of variability aspects. In addition, the question of identifying process variants in a model repository has to be addressed.

References

1. Reichert, M., Rinderle, S., Kreher, U., Dadam, P.: Adaptive process management with adept2. In: ICDE, IEEE Computer Society (2005) 1113–1114
2. Weber, B., Reichert, M., Rinderle-Ma, S.: Change patterns and change support features - enhancing flexibility in process-aware information systems. Data Knowl. Eng. **66**(3) (2008) 438–466
3. van der Aalst, W.M.P., Dumas, M., Gottschalk, F., ter Hofstede, A.H.M., Rosa, M.L., Mendling, J.: Correctness-preserving configuration of business process models. In: FASE. Volume 4961 of LNCS., Springer (2008) 46–61
4. van der Aalst, W.M.P.: Inheritance of business processes: A journey visiting four notorious problems. In: Petri Net Technology for Communication-Based Systems. Volume 2472 of LNCS., Springer (2003) 383–408
5. Dijkman, R.M., Dumas, M., García-Bañuelos, L.: Graph matching algorithms for business process model similarity search. [15] 48–63
6. van Dongen, B.F., Dijkman, R.M., Mendling, J.: Measuring similarity between business process models. In: CAiSE. Volume 5074 of LNCS., Springer (2008) 450–464
7. Weidlich, M., Polyvyanyy, A., Mendling, J., Weske, M.: Efficient computation of causal behavioural profiles using structural decomposition. In: ICATPN. (2010) accepted for publication.
8. van Dongen, B.F., van der Aalst, W.M.P., Verbeek, H.M.W.: Verification of epcs: Using reduction rules and petri nets. In: CAiSE. Volume 3520 of LNCS., Springer (2005) 372–386
9. van Glabbeek, R.: The linear time - brancing time spectrum I. The semantics of concrete, sequential processes. In: Handbook of Process Algebra. Elsevier (2001) 3–99
10. Basten, T., Aalst, W.: Inheritance of Behavior. JLAP **47**(2) (2001) 47–145
11. Schrefl, M., Stumptner, M.: Behavior-consistent specialization of object life cycles. ACM Trans. Softw. Eng. Methodol. **11**(1) (2002) 92–148
12. van der Aalst, W.M.P., Basten, T.: Identifying commonalities and differences in object life cycles using behavioral inheritance. In: ICATPN. Volume 2075 of LNCS., Springer (2001) 32–52
13. Li, C., Reichert, M., Wombacher, A.: Discovering reference models by mining process variants using a heuristic approach. [15] 344–362
14. Lu, R., Sadiq, S.W.: On the discovery of preferred work practice through business process variants. In: ER. Volume 4801 of LNCS., Springer (2007) 165–180
15. Dayal, U., Eder, J., Koehler, J., Reijers, H.A., eds.: Business Process Management, 7th International Conference, BPM 2009. Volume 5701 of LNCS., Springer (2009)