

A Process Deviation Analysis Framework

Benoît Depaïre^{1,2**}, Jo Swinnen¹, Mieke Jans¹, and Koen Vanhoof¹

¹ Hasselt University, Agoralaan Building D, 3590 Diepenbeek, Belgium,
{jo.swinnen;benoit.depaire;mieke.jans;koen.vanhoof}@uhasselt.be

² Research Foundation Flanders (FWO), Egmontstraat 5, 1000 Brussels, Belgium

Abstract. Process deviation analysis is becoming increasingly important for companies. This paper presents a framework which structures the field of process deviation analysis and identifies new research opportunities. Application of the framework starts from managerial questions which relate to specific deviation categories and methodological steps. Finally a general outline to detect high-level process deviations is formulated.

Key words: Conformance checking, Deviation Detection, High-level Deviations, Deviation Diagnosis, Process Mining

1 Introduction

In order to understand and control business processes, reliable normative process models are crucial. However, studies [7, 8] show that real process executions often deviate from their designed model. Some deviations are desirable and provide process flexibility, while others are errors or indications of fraud. Monitoring process deviations is needed for internal control purposes and has become increasingly important (cfr. Sarbanes-Oxley Act (2002), Basel II (2004) and HIPAA (1996)).

The current state of conformance checking research reveals a strong focus on questions such as “Does a case deviate?” and “Where does the process deviate?”. More important questions from a managerial perspective, such as “How does the process deviate?” and “What is causing these deviations?” received much less attention. Neither does the current literature make a clear distinction between different natures of process deviations and the possible consequences.

This paper provides a managerial framework which guides and structures process deviation analysis. The remainder of this paper is organized as follows: section 2 describes the process deviation analysis framework; section 3 suggests a formal approach to detect high-level process deviations; section 4 covers related work; and section 5 concludes the work.

** The authors wish it to be known that, in their opinion, both first two authors should be regarded as joint First Authors.

2 Process Deviation Analysis Framework (PDA-framework)

Figure 1 presents the Process Deviation Analysis framework (PDA-framework). This framework structures process deviation research and identifies three important dimensions, i.e. the deviation category, the methodological research steps and the managerial questions to be addressed. Note that this framework starts from a business perspective rather than focussing on the technical aspects.

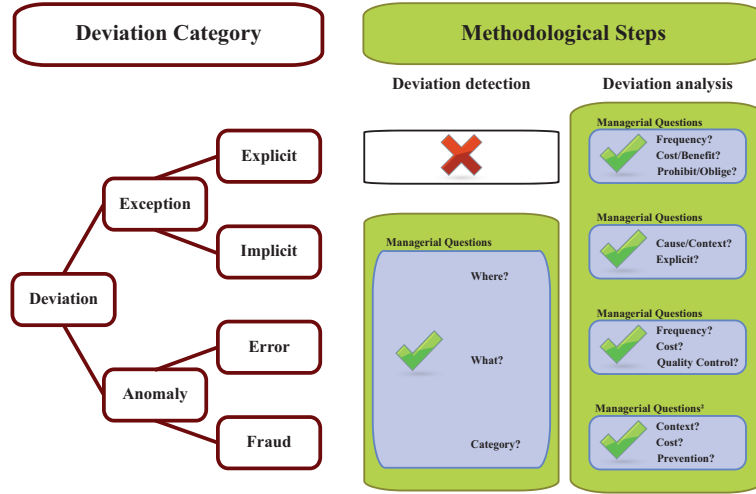


Fig. 1. The PDA-framework

2.1 Process Deviation Categories

Deviation denotes the fact that a process execution is not conform to the normative process model. A first distinction must be made between *exceptions* and *anomalies*. Exceptions are acceptable and guarantee necessary flexibility to operate effectively. *Anomalies* are deviations that provoke undesirable business results.

Explicit and Implicit Exceptions. Exceptions can be divided into *Explicit Exceptions* and *Implicit Exceptions*. Explicit exceptions are widely accepted and are either explicitly depicted in the process model or described by a set of business rules. Implicit exceptions are not formalized and are more ad-hoc, e.g. when an employee asks his supervisor to skip a less relevant activity to operate more quickly.

Errors and Fraud. Among anomalies, a distinction can be made between (operational) errors and fraud. (*Operational*) *Errors* refer to mistakes in the process execution caused by information systems, human mistakes or a combination of both. *Fraud* is the worst kind of process deviation and refers to deliberate actions by employees to work around the system for personal gains at the expense of the company.

2.2 Methodological Steps and Managerial Questions

The second and third part of the PDA-framework identify respectively two methodological steps and several managerial questions in process deviation analysis. The PDA-framework is a managerial framework and its application should always start from a specific managerial question regarding process deviations. Based on this question, the PDA-framework identifies the deviation type of interest and the appropriate methodological step.

Process Deviation Detection For all deviations, except explicit exceptions, analysis always starts with a set of three managerial questions: “Where do deviations occur?”, “What kind of deviations do occur?” and “To which deviation category do they belong?”. All three questions deal with the first methodological step of detecting process deviations. For explicit exceptions, the deviations are known and this step can be skipped.

The ‘where’ question is typically asked first. It deals with identifying deviating cases as well as locations within the process where something deviates. A large part of conformance research deals with this particular question. LTL can be used for identifying non-compliant cases and the ‘token game’ can be played to find deviating locations in a process [10]. Once process deviations are discovered, managers typically want to know how the process deviates. The current literature only partially answers this important question by indicating where activities are skipped or inserted [2]. However, managers are more interested in high-level process deviations, such as delayed or swapped activities. Section 3 presents a general approach to detect high-level deviations. Finally, a manager is often interested in a particular deviation category. Either he identifies a specific category in advance or the detected deviations are classified afterwards. To our knowledge, the work of Swinnen et al. [9] is the only research that deals with this managerial question. They summarize the detected deviations in terms of business rules to allow fast determination of the deviation category.

Process Deviation Diagnosis The second methodological step of the framework diagnoses the detected process deviations. Each deviation category relates to a different set of managerial questions. For explicit exceptions, it is important to determine their frequency and perform a cost-benefit analysis. This can lead to the integration of the exception in the main process execution, changing the pre-conditions of the exception or prohibiting the exception. For implicit exceptions, the cause or context should be determined and the desirability of the deviation has to be assessed. This can result in making the implicit exception explicit.

For operational errors, their frequency and cost should be determined and their analysis should be part of quality control programs. Corrective actions should be taken to prevent these errors. As for fraud, the frequency and cost should be determined to assess the consequences, but the process should also be redesigned or controls should be implemented to prevent fraud in the future. To our knowledge, the issue of process deviation diagnosis has not yet been addressed in the academic literature.

3 Detecting High-Level Deviations

While research predominantly focusses on where the process deviates, managers are more interested in how the process deviates. This question has only been addressed in [2], where low-level deviations of skipped and inserted activities are discovered. However, to be truly useful to management, higher level deviations need to be identified. Examples of such higher level deviations are delayed activities, activities replaced by other activities, two activities which swapped places, a set of activities which is repeated and many others. Given the space limitations, only a general approach for detecting higher level deviations is provided.

3.1 Basic Concepts

To detect process deviations, a normative model M is compared with each trace τ from an event log L . A trace can be represented as an ordered set of executed activities³, $\tau = \langle a_1, \dots, a_n \rangle$. A process model M corresponds to a directed graph, which comprises activities and control connectors (e.g. XOR-split, AND-join). Concurrency and choice constructs in the model allow multiple ways to execute the process from start to end. An execution path $p_i = \langle a_1, \dots, a_n \rangle$ is any sequence of activities that represent a valid execution of M .

3.2 Algorithm Structure

The identification of higher-level process deviations comes down to the comparison of a trace $\tau = \langle a_1, \dots, a_n \rangle_t$ with the correct execution path $p_i = \langle a_1, \dots, a_m \rangle_p$. Therefore, the first step in the deviation detection algorithm is **finding the appropriate execution path** p_i for a specific trace $\tau \in L$. A first approach matches case information with decision rules in the process model to determine the correct execution path. A second approach, applied in [2], minimizes the cost of a case execution in terms of inserted and deleted activities. Note that the alignment stage of the algorithm deals with loops, i.e. it needs to determine how many times the trace should have repeated a specific loop.

Instead of searching for high-level deviations straight away, the second step in the deviation detection algorithm **searches** for low-level deviations, i.e. **inserted**

³ More precisely, a trace is an ordered set of events where each event relates to a specific activity type

and deleted activities. Since any high-level deviation can be constructed from these two low-level deviations, it suffices to detect the low-level deviations in order to combine them into high-level deviations. This decomposition of the problem also prevents the algorithm from having to pass the event log multiple times to find all possible combinations of high-level deviations.

The third step of the algorithm is an optimization problem and combines the set of low-level deviations into high-level deviations. Note that there are multiple ways to **combine low-level deviations into high-level deviations.** For example, assume $\tau = \langle a_1, a_4, a_3, a_2, a_5 \rangle_t$ and $p_i = \langle a_1, a_2, a_3, a_4, a_5 \rangle_p$. The deviation between τ and its matching execution path p_i can be explained by a switch between activities a_2 and a_4 , but also by a delay of activity a_2 and activity a_3 .

4 Related Work

Various terms have been given to express non-conform patterns, i.e. *anomalies*, *outliers*, *discordant observations*, *exceptions*, *aberrations*, *surprises*, *peculiarities* or *contaminants*. In [5], a survey provides a structured and comprehensive overview of the research on anomaly detection in general.

With regard to anomaly detection in process aware information systems, various research has already been done [3, 4, 6, 11]. All this research is limited to the determination of whether a trace is anomalous or not and only addresses the ‘Where’ question in the deviation detection stage. It does not determine in which way and how severe these deviations are. Adriansyah et al. address these limitations [1, 2]. They defined the low-level deviations of ‘skipped’ and ‘inserted’ activities and identify their exact occurrence in a trace. The severity of these deviations is expressed by assigning a cost to skipping or inserting a certain activity. Their research focusses on the questions “Where in the process does it go wrong?” and to some extent “How does the process deviate?”, albeit at a low level of deviations.

The PDA-framework suggests that the existing research only covers a small part of process deviation analysis from a managerial point of view and one of the issues that should be addressed is the identification of high-level deviations. To define interesting high-level deviations, the work of Weber et al. [12] on change patterns in process aware information systems provides an interesting starting point.

5 Conclusions

The PDA framework decomposes process deviation analysis into three dimensions, i.e. deviation categories, methodological steps and managerial questions. The framework acts as a guidance to managers and allows them to quickly identify the type of deviation analysis they require. However, the PDA framework

also reveals that there are still many managerial questions that lack a sound scientific methodology and a set of appropriate algorithms.

The authors hope that the PDA framework will inspire other researchers to further develop the field of process deviation analysis. To this end, a general three step approach to identify high-level deviations have been suggested in this paper.

References

- [1] A. Adriansyah, N. Sidorova, and B. van Dongen. Cost-based fitness in conformance checking. In *ACSD*, pages 57–66, 2011.
- [2] A. Adriansyah, B. van Dongen, and W. M. P. Van der Aalst. Conformance checking using cost-based fitness analysis. In *EDOC*, pages 55–64. IEEE Computer Society, 2011.
- [3] F. Bezerra and J. Wainer. Anomaly detection algorithms in logs of process aware systems. In *Applied Computing 2008, VOLS 1-3*, pages 951–952, 2008.
- [4] F. Bezerra, J. Wainer, and W. van der Aalst. Anomaly detection using process mining. In *Enterprise, business process and information system modeling*, volume 29 of *Lecture Notes in Business Information Processing*, pages 149–161, 2009.
- [5] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection - a survey. Technical report, University of Minnesota, 2007.
- [6] H. Jalali and A. Baraani. Genetic-based anomaly detection in logs of process aware systems. *World Academy of Science Engineering and Technology*, 64:304–309, 2010.
- [7] A. Rozinat, I.S.M. de Jong, and C.W. Günther. Process mining applied to the test process of wafer steppers in asml. *IEEE Transactions on Systems, Man and cybernetics - Part C Applications and Reviews*, pages 1–6, 2010.
- [8] A. Rozinat and W. M. P. van der Aalst. Conformance checking of processes based on monitoring real behavior. *Inf. Syst.*, 33:64–95, March 2008.
- [9] J. Swinnen, B. Depaire, M. Jans, and Vanhoof K. A process deviation analysis - a case study. In *Business Process Management Workshops*, volume 99 of *Lecture Notes in Business Information Processing*, pages 87–98. Springer, 2012.
- [10] W.M.P. van der Aalst. Business alignment: using process mining as a tool for delta analysis and conformance testing. *Requirements Engineering*, 10:198–211, 2005.
- [11] W.M.P. Van der Aalst and A.K.A. de Medeiros. Process mining and security: Detecting anomalous process executions and checking process conformance. *Electronic Notes in Theoretical Computer Science*, 121(4):3–21, 2005.
- [12] Barbara Weber, Manfred Reichert, and Stefanie Rinderle-Ma. Change patterns and change support features - enhancing flexibility in process-aware information systems. *Data & Knowledge Engineering*, 66:438 – 466, 2008.