Business Process Improvement with Performance-Based Sequential Experiments (Extended Abstract)

Suhrid Satyal

University of New South Wales, Australia

Abstract. Various life-cycle approaches to Business Process Management (BPM) have a common assumption that a process is incrementally improved in the redesign phase. While this assumption is hardly questioned in BPM research, there is evidence from the field of AB testing that improvement concepts often do not lead to actual improvements. In this thesis, we propose a methodology named AB-BPM and a set of supporting techniques that facilitate rapid validation of business processes by conducting sequential experiments. We evaluate our methodology and techniques with real-world and synthetic case studies.

Keywords: Business process management, AB Testing, Shadow Testing, Trace simulation, Process performance indicators

1 Introduction

Business process improvement ideas often do not lead to actual improvements.Works on business improvement ideas found that only a third of the ideas observed had a positive impact [3,4] This is also illustrated by an anecdote of a European Bank. The bank improved their loan approval process by cutting its turnaround time down from one week to a few hours as a means to boost their business. What happened though was a steep decline in customer satisfaction: customers with a negative notice would complain that their application might have been declined unjustifiably; customers with a positive notice would inquire whether their application had been checked with due diligence. These observations emphasize the need to carefully test improvement hypotheses in practice because the customers and the process participants might not act as anticipated by the process analyst.

Contemporary BPM research does not provide techniques and guidelines on quickly testing and validating the supposed improvements in a fair manner. There are two major challenges for such an immediate validation. The first one is *methodological*. Classical BPM lifecycle approaches build on a labour-intensive analysis of the current process, which leads to the deployment of a redesigned version [2]. This new version is monitored in production, and if it does not meet performance objectives, it is made subject to analysis again. All this takes time. The second challenge is *architectural*. Contemporary Business Process Management Systems

(BPMSs) enable quick deployment of process improvements, but they do not provide support for validating improvement assumptions.

In this research, we address these challenges by integrating business process execution concepts with ideas from modern software engineering practices. We propose an iterative and incremental process improvement methodology named AB-BPM that integrates business process execution concepts with the idea of AB testing. AB-BPM supports the design of AB tests with simulation and shadow tests.

First, we develop a simulation technique that estimates the performance of a new process version using historical data of the old version. Since the results of simulation can be speculative, we propose shadow testing as the next step. Our Shadow testing technique partially executes the new version in production alongside the old version in such a way that the new version does not throttle the old version. Finally, we develop techniques for AB testing for redesigned processes with immediate feedback at runtime. AB testing compares two versions of the deployed process by observing users responses to versions A/B, and determines which one performs better. We propose two algorithms, LTAvgR and ProcessBandit, that dynamically adjust request allocation to the versions during the test based on their performance. We evaluate these techniques with real world and synthetic case studies.

2 Problem Statements

Despite the prevalence of live testing and documentation of the successes, contemporary business process management has not embraced this idea. There are three problems with rolling out new process versions in contemporary business process management practices:

P1: New versions of business processes are not tested in production.

The traditional BPM lifecycle encourages replacement of old versions with new versions. The newly redesigned version makes the implicit assumption that this version improves the business process. User interactions are not observed before the replacement because the new version was never executed in production. So, this improvement assumption is not validated before committing to the new version.

Live testing approaches are shown to have addressed such issues in the context of websites (e.g. [4]). However, such existing approaches do not address complexities of BPM scenarios such as lengthy execution times and involvement of human workers.

P2: Evaluating new versions of a process in production is risky.

There is an inherent risk of executing new versions that have not been validated in production. We henceforth refer to this as *risk of exposure.* Careful offline analysis, as suggested by the BPM lifecycle, can reduce the possibility of producing bad redesigns but the risk of exposure still remains because a redesigned process completely replaces the old version. Alternatively, live testing approaches follow the motto “test fairly, fail fast”, which means that there is minimal upfront analysis before testing

[1]. Typically, in techniques like AB Testing, the old and the new versions are deployed simultaneously and half of the user requests are directed to the new version. As such, the risk of exposing inferior versions is high.

P3: Process executions have to be evaluated with incomplete observations. To reduce risk of exposure, advanced AB testing techniques for websites can observe metrics like click-through rate and dynamically adjust user allocation during the test. This ensures that the better version receives more traffic over time. However, such approaches cannot be used as-is in BPM because of the complexities in performance indicators. Process instances can have unique identifiers, and each execution can have multiple process performance indicators such as duration and cost. Quality of a process version can be assessed by observing performance indicators of the corresponding process instances. First, the performance indicators may be tied to duration. The delays in which these performance indicators are observed influences when and how the allocation of users to the versions are adjusted. Second, not all of the required performance indicators may be observable at the same time. For instance, duration and user satisfaction scores may be obtained at different times with delays of varying length. Furthermore, indicators like user satisfaction may not be observed at all. Finally, the performance of a process can also be influenced by factors such as resource constraints, the environment, and market fluctuation. To adopt advanced AB testing techniques in BPM, we require monitoring support, mechanisms of evaluating process versions in presence of these complications with performance indicators, and request allocation algorithms for the BPM scenario.

3 Contributions

This research makes original contributions in the area of business process improvement, business process simulation, and live testing. In this research, we have proposed a business process improvement methodology named AB-BPM [6,8]. It extends the business process lifecycle to provide support for rapid and fair validation of process improvement hypotheses. The AB-BPM methodology is supported by three classes of techniques: simulation, shadow testing and AB testing. Fig. 1 illustrates how an old version (A) is compared with a new version (B) using these techniques.

Simulation
Our simulation technique takes historical data (event logs) of a process version, the process model of the new version, and estimates of the metrics of some activities in the new version as the inputs. It produces event logs for the new version, which is then used to estimate the performance of the new version. We construct a Transition Simulation Tree (TST), a rooted tree data structure that summarizes decisions and metrics available in an event log of the existing process version.The TST created from the old version is used in conjunction with a BPMS to derive the execution log of the new version [8]. We demonstrate the efficacy of this approach with case studies from the domain of helicopter pilot licensing and banking.

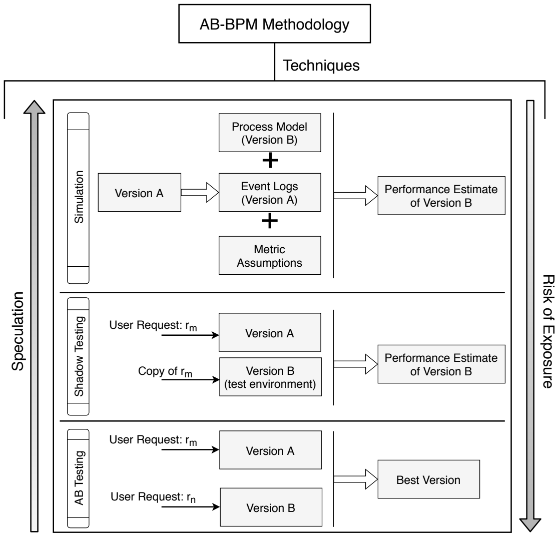

Fig. 1. Process improvement approaches of AB-BPM

The risk of exposure is low in simulation because this technique is performed offline. However, the results can be speculative because of the implicit assumptions about customer behavior and the explicit assumptions about the execution of some tasks in the new version.

Shadow Testing
Shadow testing executes the new version in production but hides the execution from customers. User requests from the production environment are duplicated and executed in both versions. Execution of the new version can affect the performance of the old version during test execution. Therefore, the risk of exposure is higher than simulation. However, the new version is executed online and the results are less speculative.
In our approach of shadow testing, we reduce the risk of exposure further by partially simulating the new version and instantiating the new version only when the performance overhead of testing is acceptable. As a result, there is a trade-off between speculativeness and risk of exposure. This is achieved through a new modular BPMS architecture, an overhead management algorithm, and a decision tree for partial simulation [5]. We evaluate this approach with synthetic and realistic processes from the literature.

AB Testing
AB testing executes the two versions in production such that each version serves a portion of customers. This approach has high risk of exposure, but the results of the tests are fair and reliable. To mitigate the risk of exposure, we use *multi-armed bandit* algorithms that dynamically allocate user requests to the process versions based on their performance. These algorithms observe performance indicators of each version, and shift the

user requests towards the better performing version in the given context. Effectively, the best version can be identified by the end of the test.

We propose a modular architecture that supports the execution of such algorithms. We design two algorithms, *LtAvgR* and *ProcessBandit*, for the BPMS case and propose reward designs for these algorithms [6,7] Our AB testing solution observes performance of process execution in the form of Process Performance Indicators (PPIs), convert the PPIs into rewards, updates the rewards in case of incomplete observations (as explained in P3), and dynamically allocates more requests to the version that has better performance.

We evaluate this solution with real-world case studies from the domain of building permit approval, banking, and helicopter license approval. We also demonstrate the efficacy of the algorithms with experiments on convergence to optimal solution and scalability using synthetic cases.

4 References

References

1. L. Bass, I. Weber, and L. Zhu. *DevOps - A Software Architect's Perspective*. SEI series in software engineering. Addison-Wesley, 2015.
2. M. Dumas, M. La Rosa, J. Mendling, and H. A. Reijers. *Fundamentals of Business Process Management, Second Edition*. Springer, 2018.
3. C. W. Holland. *Breakthrough Business Results With MVT: A Fast, Cost-Free "Secret Weapon" for Boosting Sales, Cutting Expenses, and Improving Any Business Process*. John Wiley & Sons, 2005.
4. K. Kevic, B. Murphy, L. A. Williams, and J. Beckmann. Characterizing experimentation in continuous deployment: A case study on Bing. In *International Conference on Software Engineering: Software Engineering in Practice Track*, pages 123–132. IEEE Press, 2017.
5. S. Satyal, I. Weber, H. Paik, C. D. Ciccio, and J. Mendling. Shadow testing for business process improvement. In *Confederated International Conferences: CoopIS*, pages 153–171, 2018.
6. S. Satyal, I. Weber, H. Paik, C. Di Ciccio, and J. Mendling. AB-BPM: performance-driven instance routing for business process improvement. In *International Conference on Business Process Management, BPM*, pages 113–129, 2017.
7. S. Satyal, I. Weber, H. Paik, C. Di Ciccio, and J. Mendling. AB Testing for process versions with contextual multi-armed bandit algorithms. In *International Conference on Advanced Information Systems Engineering (CAiSE)*, pages 19–34, 2018.
8. S. Satyal, I. Weber, H. Paik, C. Di Ciccio, and J. Mendling. Business process improvement with the AB-BPM methodology. *Information Systems*, 2018.