Towards Guidelines for Conducting Software Process Simulation in Industry

Nauman bin Ali (nauman.ali@gmail.com)
Blekinge Institute of Technology, Sweden

In the 1960s researchers and practitioners realized that software development was more than just creating effective programming languages [1]. It is a complex endeavour requiring creative effort from people guided by an appropriate organization and procedures [1]. Early descriptions of the software development lifecycle were presented detailing various stages of a software product’s lifetime e.g. Royce model [3] and the Spiral model [4]. The lifecycle models provided the conceptual scheme of how the software process should be carried out [1] [5]. These lifecycle models were the foundation for software processes, which add details to provide practical guidance and control on software development [1] [5].

A software process can be defined as: “the coherent set of policies, organizational structures, technologies, procedures, and artifacts that are needed to conceive, develop, deploy, and maintain a software product” [1]. A software process model is therefore “any representation” of the software process that captures the relevant aspects of a process for the intended modelling purpose [6]. Typical details in such models are activities to accomplish the process objectives, roles of people, artefacts developed and tools to be used [1].

Like any model, software process models provide abstraction [2] and are used when use of the complete process is undesirable or impractical [6]. Some of the typical uses of software process models include [1], [6]: Developing a shared process understanding by promoting a representation of the process through a model that can also be used for training newly hired personnel, improving the existing processes and proposing new ones through analysis and validation of process models and by prediction of process behaviour, and simulate if the models are enactable to identify potential improvements.

Most software process models treat software development as a static activity [7]. Therefore, Software Process Simulation Modelling (SPSM) becomes pertinent for the following reasons:

- Simulation can represent uncertainty in software development that cannot be captured by analytical models [2].
- Dynamic variables like productivity and defect detection rates are not captured in static models [2] [8].
- Behaviour and decisions taken at one stage affect others in complex and indirect ways that needs to be catered for when analysing the process behaviour [2].
- Manipulation of the actual software process is risky and expensive, whereas SPSM is proposed as an inexpensive way to gain insights into the actual process [8].

Simulation is the numerical evaluation of a mathematical model that imitates the real-world system behaviour [9] [8]. In case of SPSM, the system being modelled is the software process. A software process simulation model is a computerized model that focuses on one or more of the
software development, maintenance or evolution processes especially relevant to the phenomenon of interest [2]. Law [10] classifies simulation models along three dimensions: static vs. dynamic, discrete vs. continuous, and deterministic vs. stochastic simulation models.

This thesis aims to facilitate adoption of SPSM in industrial practice by exploring two directions. First it aims to establish the usefulness of SPSM for its intended purposes, e.g. for planning, training and as an alternative to study the real world software (industrial and open source) development. This is done by identifying, aggregating and assessing the evidence reported in literature to evaluate the usefulness of SPSM in real-world (industrial and open source) software development. Secondly, this thesis aims to provide a process for conducting an SPSM based study in industry.

A literature review, two systematic literature reviews (SLR), a case study and an action research study were conducted. A literature review of existing SLRs was done to identify the strategies for selecting studies [19]. The resulting process for study selection [20] was utilized in an SLR to capture and aggregate evidence regarding the usefulness of SPSM. Another SLR was used to identify existing process descriptions of how to conduct an SPSM study. The consolidated process and associated guidelines identified in this review were used in an action research study to develop a simulation model of the testing process for training purposes in a large telecommunication vendor. The action research was preceded by a case study to understand the testing process at the company.

A comprehensive process for conducting an SPSM based study in industry [17] is presented in this thesis, which reduces confusion in the choice of a process caused by the plethora of process prescriptions. It can act as a checklist for advanced simulation modellers as well as facilitate novice modellers for conducting SPSM based studies. The utility of this model was demonstrated successfully in this thesis by application in an industrial context.

This thesis by identifying the limitations in the current SPSM research [18] has taken the first step towards improvement. The criticism of SPSM is not intended to dismiss its use, but to identify the weaknesses, raise awareness and hopefully improve SPSM research and practice. Given the strong “proof-of-concept” work in SPSM literature it is likely that SPSM is a tool with potential benefits for both research and practice. However, it should be seen as such, a tool and not as a research method. Making this distinction will allow future research to focus on improving SPSM from two likely users’ perspective: the practitioners and the researchers. From practitioners’ perspective, there is a need to:

- Present evidence of the usefulness of SPSM for real-world software development.
- Investigate the requirements that practitioners have for SPSM, e.g. usability requirements on simulation tools.
- Assess the cost of adopting SPSM in terms of training, resources and required infrastructure (e.g. stability in processes, measurement programs).

In this thesis, the need to learn from other disciplines has been placed under the spotlight. Their methodology, reporting guidelines, best practices and lessons learned should be studied and evaluated for SPSM. Lastly, it is strongly recommended that the future SPSM research should follow reporting guidelines (e.g. [13]), emphasize evaluation of SPSM for intended purposes (e.g. [11]) and strictly follow the guidelines by Wernick and Hall [14] to “minimise the problems with simulations”.
It is encouraging to see that the SPSM community is also reflecting on the reasons why the SPSM research trend is slowing down [15] and why it has not been widely adopted in industry [8]. There is a lot of congruence in the future directions for research identified in this thesis and the ones proposed in position papers and panel discussions at the 2012 International Conference on Software and System Process, like [12] [16]. This thesis however supplements the motivation for these proposed directions with evidence from the current SPSM literature.

Based on the findings of this thesis the following conclusions are drawn:

- There is no conclusive evidence to claim the usefulness of SPSM for any of the intended purposes. There is a dire need for empirical evaluation of SPSM in real-world software development settings.
- Current research lacks both scientific rigor and industrial relevance.
- Future research should utilize the guidelines to conduct and report SPSM based studies with an emphasis on evaluation of SPSM.
- The analysis of existing process guidelines for conducting SPSM revealed considerable similarity among them. It was further concluded that the process for conducting an SPSM based study is independent of the factors that were highlighted as the focus of each of these processes e.g. simulation approach, modellers experience or organizational size.
- The consolidation of process proposals for conducting an SPSM based study from SPSM literature resulted in a process that is very similar to the one recommended for simulation-based studies in general. This suggests that SPSM and simulation in other disciplines are more similar than previously thought.
- The consolidated process based on the SPSM literature was successfully used to develop a system dynamics model of the testing process at the case company.

References:


