ARTICLE IN PRESS

Information and Software Technology xxx (xxxx) xxx–xxx

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

Developing an agent-based simulation model of software evolution

Shallaw Mohammed Ali[a,*], Martina Doolan[b], Paul Wernick[c], Ed Wakelam[b]

[a] Al_Qalam University College, Ali_mardan Street, Kirkuk, Iraq, Naz Naz area, Gulan Street - Building# 447 Erbil, Iraq
[b] University of Hertfordshire, School of Computer Science Hatfield, Hertfordshire, United Kingdom
[c] University of Hertfordshire, Centre for Computer Science and Informatics Research School of Computer Science Hatfield, Hertfordshire, United Kingdom

ARTICLE INFO

Keywords:
Software evolution
Simulation modelling
Agent-based simulation modelling
System evolution process

ABSTRACT

Context: In attempt to simulate the factors that affect the software evolution behaviour and possibly predict it, several simulation models have been developed recently. The current system dynamic (SD) simulation model of software evolution process was built based on actor-network theory (ANT) of software evolution by using system dynamic environment, which is not a suitable environment to reflect the complexity of ANT theory. In addition the SD model has not been investigated for its ability to represent the real-world process of software evolution.
Objectives: This paper aims to re-implements the current SD model to an agent-based simulation environment 'Repast' and checks the behaviour of the new model compared to the existing SD model. It also aims to investigate the ability of the new Repast model to represent the real-world process of software evolution.
Methods: a new agent-based simulation model is developed based on the current SD model's specifications and then tests similar to the previous model tests are conducted in order to perform a comparative evaluation between of these two results. In addition an investigation is carried out through an interview with an expert in software development area to investigate the model's ability to represent real-world process of software evolution.
Results: The Repast model shows more stable behaviour compared with the SD model. Results also found that the evolution health of the software can be calibrated quantitatively and that the new Repast model does have the ability to represent real-world processes of software evolution.
Conclusion: It is concluded that by applying a more suitable simulation environment (agent-based) to represent ANT theory of software evolution, that this new simulation model will show more stable bahaviour compared with the previous SD model; And it will also shows the ability to represent (at least quantatively) the real-world aspect of software evolution.

1. Introduction

In software engineering, the process of the sequence of changes that occurs during the software systems lifetime comprising both system development and maintenance, was first termed a *software evolution process* by Lehman and Belady in the 1970s [1] "The successful evolution of software is becoming increasingly critical since the increasing dependence on computers and software at all levels of the society" [2]. Therefore, in order to find ways to manage and control the evolution of software systems, researchers and practitioners have strived to reveal the process by which software systems have evolved [3].

One of these endeavours was Lehman's description of the "*global software process*" [4] as a feedback system of a collection of people and events that control the evolution of software-based systems. Lehman presents this process as being driven by feedback which is made explicit in the 7th law of software evolution that the "E-type evolution processes constitute multi-level, multi-loop, multi-agent feedback systems" [5].

Based on this view of the software evolution process, Wernick and Lehman [6] and Kahen et al. [7] developed several simulation models of software evolution in an attempt to understand and reveal the reasons and the factors behind software evolution. Recently, Wernick et al. [8] have developed a simulation model of the software evolution process by applying social view actor-network theory (ANT) presented by Latour [9]. ANT can be described as a perspective for viewing and describing social and technological situations by considering both human and non-human elements equally as active entities within an interconnection network [10]. Wernick et al. built this model by using a system dynamic (SD) general purpose simulation environment. They justified using this tool for modelling as it provides a usable and simple toolset. However, according to [8], a pure SD simulation environment is

* Corresponding author.
E-mail addresses: shallaw.abdulkader@gmail.com (S. Mohammed Ali), m.a.doolan@herts.ac.uk (M. Doolan), p.d.wernick@herts.ac.uk (P. Wernick), e.wakelam@herts.ac.uk (E. Wakelam).

https://doi.org/10.1016/j.infsof.2017.11.013
Received 23 March 2017; Received in revised form 19 November 2017; Accepted 20 November 2017
0950-5849/ © 2017 Elsevier B.V. All rights reserved.

Please cite this article as: Mohammed Ali, S., Information and Software Technology (2017), https://doi.org/10.1016/j.infsof.2017.11.013

not the most appropriate environment for representing some ANT aspects. The SD environment provides no support for the flexibility to represent the complexity of the participants making up the model.

Therefore, Wernick et al. suggested reworking the SD model to a more appropriate, agent-based simulation environment and modifying the new model to a more accurate simulation model, then checking the behaviour of the new model against the existing SD model. Based on this suggestion, the study presented in this paper was undertaken to rework the current SD model to an agent-based simulation environment and to check the behaviour of this new model compared with the current SD model. Therefore, in this work, the Repast simulation toolkit was chosen as an agent based simulation platform that was used to develop the new model based on the specifications of the existing SD model.

Furthermore, this study addresses the issue of calibrating the SD model which was conducted without referring to any real world data due to the lack of available data in representing real world processes of evolution [8]. In other words, the ability of the model to reflect the real world of software evolution processes has not yet been investigated. This study addresses this issue through an investigation conducted to check the ability of the new agent-based model to reflect real-world aspects of software evolution. At this stage of the research, the new model does not take into account individual human aspects; the model is a starting point for future work where this aspect will be further developed.

Consequently, this work aims to address the following research questions:

RQ1 How does the ANT-based model of software evolution built in an SD simulation environment behave if reworked to an agent-based simulation environment, in comparison to the existing SD model?

RQ2 Does the new agent-based simulation model of software evolution processes have the ability to reflect the real-world process of software evolution?

Section two presents the background and literature review of actor-network theory (ANT), an explanation of the ANT-based model of software evolution 'SD model' and a description of the agent-based simulation modelling including its advantages and issues. Section three describes the methodology of conducting this research which includes specifications, design and implementation phases that were undertaken to build the Repast simulation model of software evolution. In section four, the investigation conducted to evaluate the Repast simulation model and the findings of these investigations is presented. This section consists of two parts. The first part presents a comparative evaluation between the new Repast model and the previous SD model, including their results. This part addresses the first research question. In the second part, an investigation of the Repast model to check its ability to reflect the real-world process of software evolution is described, including the calibration of the investigation results. The second research question is addressed in this part. In section five, the discussion of the above findings is presented. Finally, this paper ends with section six which presents evaluation of the research work, the potential threats to validity, and the concluding view of the contributions made in this work.

2. Background and related work

This section presents a background of ANT theory, an explanation of the SD simulation model of software evolution and a literature review of agent-based simulation modelling.

2.1. Actor network theory (ANT)

With the aim of explaining complicated interactions in a research setting, Bruno Latour and Michel Callon described, in the early 1980s, the principle of actor-network theory (ANT) as a perspective for viewing complex social situations [11]. Latour [10] claimed that ANT theory differs from the traditional view of social and technological theory. In the traditional view, elements forming the social situations are described as categories such as large, small, human and non-human [12], while ANT theory describes both human and non-human elements equally as an active entity within an interconnection network [10]. According to Wernick et al. [8], the ANT view of the social world can be described as seeing the complexity of social world behaviour and the technical situations within it as a situation caused by the correlations between the elements that form the social world. In the concept of ANT, Latour [10] presented three types of elements: actors, mediators and intermediaries.

Within ANT, the actor was described as an active entity that is "not the source of an action but the moving target of a vast array of entities swarming toward it" [9]. The actor can be a collective of human 'developers, manager', non-human 'system' and even intangible elements such as 'idea, situation' [13]. Latour [9] also stated that using the term actor was not arbitrary. According to the author, it refers to the actors in theatre shows whose acting is constrained by different factors that shape their roles.

Latour described a mediator as an element within ANT that can transform, translate, distort and modify the meaning or elements that they are supposed to carry [9]. This means that even if the mediator looks very simple it may turn out to be complex since it can create unpredictable behaviour which will affect the entire element connected with it in the actor-network. Therefore, mediators can be recognised as the elements that are both written and interpreted by humans and by whom the interpretation of the written elements may differ from one person to another [3]. Latour described possible examples of the mediator as *law, science, religion, economies, psyches, moralities, politics* and *organizations* [9].

An intermediary was described by Latour as an element whose output depends only on its inputs. It is anything that passes information from and stands between one actor and another [13], transparently moving data without affecting its meaning. Therefore, in order to define an intermediary's outputs, it is enough to define its inputs. No matter how complex the intermediaries, they can be ignored in building cause-and-effect models. However, Latour illustrates the changeability and complexity of a description of both intermediary and mediator by example of a simple intermediary such as a computer function that can turn out to be a *horrendously complex* mediator if it breaks down; intermediaries can only be ignored if they continue to make no semantic changes to their inputs over a simulation run.

The essential concept in ANT is the communications channel: networks that link together the actors, mediators and intermediaries. Within this network of communications channels, actors may join or leave a network and are constantly changing the web of relationships. By joining the network, these actors may bring their own network with them; an actor can sometimes be decomposed into one or more sub-networks. Moreover, within ANT, the level of commitment of each actor to the goal of the system is illustrated as a changeable commitment that depends on the actor's own situation and the influences on it from other network elements. [9]

2.2. System dynamic (SD) simulation model of software evolution

The following description of the SD software evolution simulation model is based on Wernick et al.'s paper [8]. The reason of focusing on Wernick et al.'s paper in illustrating SD model is that the concept and the implementation of this model was essentially presented and described through this paper. Furthermore, the new model (Repast model) is the continuing work of Wernick et al.'s paper recommendations.

By using ANT, Wernick et al. developed a software evolution simulation model of a *global software process* based on a system dynamics environment. A global software process includes in one concept the collections of people, things and events that control software-based system evolution [4]. The purpose of developing an ANT-based model

Download English Version:

https://daneshyari.com/en/article/6948099

Download Persian Version:

https://daneshyari.com/article/6948099

Daneshyari.com