TEXAS STATE
THE GRADUATE COLLEGE

# Thesis Proposal Form

Submit **one copy** of this form with **original or electronic signatures and proposal attached** to The Graduate College. It is the student's responsibility to make sure The Graduate College receives this form. Failure to submit the thesis proposal prior to completing a thesis could delay graduation.

| | | | |
|---|---|---|---|
| Student Name: | Mark McDermott | Texas State ID: | A04474453 |
| Major: | Computer Science | Degree: | Master's in CS |
| Student Signature: | member: 5109659C-76AE-4AFE-957D-6E6F 7343FE36 EEBD18A2-39A8-4175-ABB5-DC1363DBF657 Digitally signed by member: 5109659C-76AE-4AFE-957D-6E6F7343FE3 6 EEBD18A2-39A8-4175-ABB5-DC1363DBF657 Date: 2019.10.16 08:30:10 -05'00' | Date: | 10/16/2019 |
| Tentative Thesis Title: | Rigorous analysis of combined software processes via runtime verification | | |

Is the committee-approved proposal attached to this form? [✓ Yes] No

Does research involve human subjects (including surveys or use of secondary data)? Yes [✓ No]

If yes, is Texas State IRB Approval letter attached? Yes No

Does research involve the use of vertebrate animals? Yes [✓ No]

If yes, provide Texas State IACUC approval code: ____________

By signing this form, the thesis committee members approve the attached thesis proposal.

| Printed Name of Committee Member | Department | Signature | Date |
|---|---|---|---|
| Rodion Podorozhny<br>Thesis Committee Chair/Co-Chair | CS | Rodion Podorozhny Digitally signed by Rodion Podorozhny Date: 2019.10.16 11:44:09 -05'00' | 10/16/2019 |
| Anne Ngu<br>Co-Chair (if applicable) | CS | [illegible] Digitally signed by Anne DN: cn=Anne, o, ou, email=angu@txstate.edu, c=US Date: 2019.10.18 09:23:07 -05'00' | 10/18/2019 |
| Guowei Yang | CS | | |
| | | | |
| | | | |

Signatures below indicate the departmental approval of the above recommendation:

| Printed Name | Signature | Date |
|---|---|---|
| Graduate Program Advisor | | |
| Hongchi Shi<br>Department Chair | [illegible] | 10/28/2019 |

**For The Graduate College Use Only**

Based on the recommendation above, the Thesis Proposal is: ☐ Approved ☐ Disapproved

Dean or Associate Dean of The Graduate College ____________ Date ____________

# Thesis proposal:

# Rigorous analysis of combined software processes via runtime verification

by

Mark McDermott

A04474453

One the main goals of software engineering is to be able to evaluate, compare and improve software processes to increase software qualities produced by such processes. To be able to do so rigorously, in automated fashion, a process programming approach was suggested [1]. A number of process languages and associated enactment systems have appeared over the years that either tried to adapt regular programming languages to process definition or suggested new process oriented languages. One important feature of a language for rigorous analysis is rigorously defined semantics. Such rigorously defined semantics enables one to use existing program analysis algorithms to verification of processes. While there is a great deal of work in the area of process and programming we will focus on the existing process language called Little-JIL and its associated enactment system [2].

This has already been used to rigorously analyze software processes: e.g. [3], [4]. In these works a process definition was mapped to an input language of a data flow analysis system called Flavors and then analyzed against given properties.

In this work we will focus on runtime verification of a software process as it is enacted by Little-JIL enactment system. Similar systems exists for programming languages, e.g. Java Pathfinder for Java programs. Yet this has not been done for software process definitions.

In addition, the proposed process analysis approach will allow to analyze several combined processes enacted together in a single resource environment. Both in the software development domain and other domains, e.g. medical processes, it is quite often the case that several processes developed by different developers have to run in the same environment and comply with the same set of properties. For instance, in software development domain, we might have a software synthesis process such as object oriented development, run concurrently with a testing plan process. In medical domain, several processes can be applied to the same patient.

In the proposed process analysis system it will be possible to verify properties of several combined processes.

This project will involve the following stages:

1. Modify the current Little-JIL enactment system with an ability to check predefined properties at given states of process enactment
2. Implement a non-deterministic execution of a software process definition in the Little-JIL enactment system
3. Design and perform an experiment of verification of a combination of two or more processes in the same resource environment against given properties

**References**:

[1] "Software Processes are Software Too, Revisited" by Leon Osterweil, International Conference on Software Engineering (ICSE), 1997

[2] "Using Little-JIL to coordinate agents in Software Engineering" Alexander Wise et al., Automated Software Engineering Conference 2000

[3] "Verifying Properties of Process Definitions" by Jamieson M. Cobleigh, Lori A. Clarke, Leon J. Osterweil, ISSTA 2000

[4] "Analyzing Medical Processes" by Bin Chen, George S. Avrunin, Elizabeth A. Henneman, Lori A. Clarke, Leon J. Osterweil, Philip L. Henneman, International Conference on Software Engineering (ICSE'08), Leipzig, Germany, May 2008