Simulation and Visualization for Population Protocols and Dynamic Networks

Author:	Haoxuan Wang
Supervisor:	Dr. Othon Michail Dr. Vitaliy Kurlin

April 25, 2018

Declaration of academic integrity

I confirm that I have read and understood the University's Academic Integrity Policy.

I confirm that I have acted honestly, ethically and professionally in conduct leading to assessment for the programme of study.

I confirm that I have not copied material from another source nor committed plagiarism nor fabricated, falsified or embellished data when completing the attached piece of work.

I confirm that I have not copied material from another source, nor colluded with any other student in the preparation and production of this work.

Date: April 25, 2018

Abstract

The abstract[1] [2] [3]

Contents

1 Introduction **1**

1.1 Aims and Objectives 1

1.2 The challenges in the project 1

1.2.1 Heterogeneous for different types of Population Models 1

1.2.2 Heterogeneous for different types of Protocols 1

1.2.3 Human factor: Lacking Experience for model visualization 1

1.2.4 The Programme 1

1.2.5 Evoluation of the project 2

2 Background **2**

2.1 Brief Introduction to Population Protocols [1] 2

2.2 From Population Protocols to Network Constructor 2

2.3 Terminating Shape Constructor 2

References **3**

1 Introduction

1.1 Aims and Objectives

The project aimed to study general population protocols [1] and its two derived model, network constructor [2] and terminating shape constructor [3] (specifically for grid network construction). It also attempted to experimentally simulate, visualise and compare these protocols via building the simulator and visualizer.

1.2 The challenges in the project

1.2.1 Heterogeneous for different types of Population Models

The theoretical models involved in three main different models initially originated in population protocols. These three models share inherently common points but there are also some conceptual differences in between them. For instance, the network constructor [2] and terminating shape constructor [3] involves state of connections in between two nodes while the original population protocol does not. The node of terminating shape constructor has its complexity structurally compared with the other two types of model.

1.2.2 Heterogeneous for different types of Protocols

The protocols discussed in the related papers [1, 2, 3] involves many different protocols. The protocols may totally different on many characteristics, such as their different computational ability, different ending in either or termination, computation target. These differences between protocol to protocol may lead the simulator and visualizer hardly to be developed and fully tested.

1.2.3 Human factor: Lacking Experience for model visualization

Prior to this project, the author has totally no experiences on model simulation and also no knowledge on what the related code library will be involved. Learning may take more time than its expected.

1.2.4 The Programme

The final programme contains an UI with an fix-sized area to illustrate the process of a particular protocol. The state of elements[*] In addition, it contains a information panel contains some related information with regard of the population itself, including:

- Number of nodes
- Number of nodes distinguished in different status
- Number of selections for pairs of nodes[†] that scheduler had took
- Number of effective interactions the population executed

[*] "Elements" refers nodes in general population protocol, but also includes edge if the protocol involves edge states.

[†] may also include pair of ports for terminating shape constructor

Additionally, it provides a set of parameters' settings regarding the initial configuration for the protocol and the population to be simulated, which includes:

- The number of nodes included in the simulation
- The initial state for the nodes[‡]
- The protocol type (and also different sets of transition rules for the protocol)
- Option on whether to jump forward simulation of initially n times selection from scheduler, and the value of n if the option is enabled

1.2.5 Evoluation of the project

In general, the simulator successfully implemented a series of different protocols for the three model mentioned above.

UI The UI functions of simulator is verified through a large number of different population simulations. This ensures the UI functions work as it expected in design stage. These experiments on theoretical model may also be asserted the correctness of model through the output configuration of these simulations.

Population Model The model partition of the simulator developed through Testing driven development method, which indicates the testing suits according to the specification had been written before any model code starts.

2 Background

2.1 Brief Introduction to Population Protocols [1]

Population protocols are theoretical models for distributed computation. The model contains a collection of indistinguishable agents. They (the agents) carry out computation tasks through directly pair-wised interactions. The interaction pattern of agents is unpredictable from perspective of agents themselves but is controlled through an adversary scheduler with

2.2 From Population Protocols to Network Constructor

2.3 Terminating Shape Constructor

[‡] The state of edge for dynamic networks (i.e. network constructor and terminating shape constructor) should be always ”0” (i.e. inactivated) at initial, so it is omitted here.

References

[1] J. Aspnes and E. Ruppert, “An introduction to population protocols,” *Bulletin of the European Association for Theoretical Computer Science*, vol. 93, pp. 98–117, Oct. 2007.

[2] O. Michail and P. G. Spirakis, “Simple and efficient local codes for distributed stable network construction,” *Distributed Computing*, vol. 29, no. 3, pp. 207–237, 2016. [Online]. Available: https://link.springer.com/article/10.1007/s00446-015-0257-4

[3] O. Michail, “Terminating distributed construction of shapes and patterns in a fair solution of automata,” *Distributed Computing*, pp. 1–23, 2017. [Online]. Available: http://link.springer.com/article/10.1007/s00446-017-0309-z