Reserach-proposal

Jonathan Thaler

January 10, 2017

Abstract

This paper describes the idea, aim and impact of the PhD study do be undertaken.

1 Project-Plan

1.1 Years

The whole PhD lasts for 3 years and thus I will structure it according to 3 years where each year will be a major milestone - which is also intended by the Computer School.

1.1.1 1st Year: Basics

In this year I will learn basics and develop and research the methodology I will use for the main work in the 2nd year. Also I want to write a paper of how to apply functional programming to ABM/S. The time-frame will be set by the date of the 1st year annual oral report which will happen beginning of July thus there are about 6 Months time (counting including January '17 and leaving out July '17). These are the things I want / need to achieve this year:

- Prototyping in Haskell, Scala and Java
- Study Actor-Model theory: Hewitt, Greif, Clinger, Agha
- Get into reasoning about programs
- Basics of Economics [2], [**?**]
- Basics of ACE: Tesfatsion
- Implement PureAgents Library
- Write the paper
- Write

- Prototype Auction/Fishmarket using Akka and Haskell and describe results in 1st year report.
- Write 1st year report

Important and mile-stone dates:

April (End)	Finished Paper: functional ABM
June (Mid)	Finished writing 1st year report
July	Oral annual report

TODO-List as of January 10, 2017

- send abstract
- investigate speed of divergence of simulation runs using the terminology of gleichzeitige ungleichzeitigkeiten
- 2 complex system can lead to a total different outcome when initial settings differ only by fractions. thus it may also be the case for specific ABM/S?
- todo: render paths of the agents in heroes and cowards
- todo: abm/s of a go game
- IDEA: what about an ABM/S of karma and rebirth? add to genesis paper
- IDEA: what about ABM/S generating sound? could be a perfect example for Yampa due to its signal functions. the sound is the result of interactions of agents which try to generate harmonies and agents trying to create dissonance
- IDEA: what about abm/s creating drawings/art? 2d continuous and each agents path is drawn
- haskell: dont have objects with methods which can call between each other but we need some way of representing agents. this is done using a struct type with a behaviour function and messaging mechanisms. important: agents are not carried arround but messages are sent to a receiver identified by an id.
- messaging mechanisms have up- and downsides, elaborate on it.
- reason for patterns: heroes try to stay 50 % in between and have selected 2 cowards which themselves are at the borders opposite. need much more cowards than heroes 75/25. TODO: investigate pair-wise movement.
- hypothesis: individual runs diverge from the first step on but the global behaviour stays the same.

- question: do emergent patterns break down / global dynamics change completely in some ABM/S when changing sim-semantic? which kind of ABM could show this behaviour? which properties are responsible for it?
- parallel more natural in haskell
- sequential more natural in java
- concurrent difficult in both, using stm in haskell it becomes very natural, STM available in java too. actors are an even better approach but is having problems with time in simulation
- implement conways game of life: parallel discrete: read global environment, write local
- implement schelling segregation
- generalize 2d grid-renderer: agentToCell maps agent to color and discrete 2d - coord
- generalize 2d continuous renderer: color, position and direction
- implement simtime Actors
- only one function: agenttransformer. replace update by call to agenttransformer with message "DT" and parameter dt. add mechanism to add sys messages
- implement sequential version: PureAgentsSeq. should be easy starting from par version. implement environment accesslike STM: each agent can read/write complete env
- implement environment for parallel
- performance unacceptable: 1000 in haskell vs 100.000 in java is a shame on haskell, more should be possible
- implement SIRS on a 8neighbourhood grid (nearly the same as wildfire). difference only in neighbourhood filter, rest of code should be same (sendrandomagent)
- send abstract to peer and thorsten in 1st week of january. try to submit paper on SIMPAT with deadline of 30th april
- embed PureAgentsPar in Yampa: PureAgentsYampa
- implement agent monad: PureAgentsMonadic. but what is an AgentMonad?
- when having agent monad, run in dunai
- look into QuickCheck and HPC

- problem: so far only agents with same static messagetypes, environment and states, can communicate: the agents are homogenous. how can we implement hetereogenous agents in this library?
- run PureAgents inside Dunai/Yampa
- where does 1st and 2nd year research meet: reasoning about emergent properties of the simulation. in case of HAC the crosspattern, in case of ACE the convergence to stable prices
- Continue investigations regarding the heroes and cowards game
- Explore the literature on complex systems to find an explanation for the emergence of the macro level patterns in the Heroes & Cowards game
- Consider the next step: Implementing an economics example at different levels of complexity
- Continue writing conference paper and looking out for an appropriate conference
- paper abstract: local-only immutable data, explicit dataflow, higher order functions and recursion
- paper: we also present a novel approach to implementing ABM/S by implementing 3 of the semantics in pure functional Haskell and comparing them to OO Java and examine for which semantics boths methods are well suited an which not.
- paper: contribution: 1. development of terminology of simulation semantics 2. pure functional approach and comparison to state-of-the-art oo 3. influence of semantics on dynamics 4. reasoning about dynamics

January TODO

February TODO

March TODO

April TODO

May TODO

June TODO

July TODO

1.1.2 2nd Year: Main Work

Applying 1st year results, methods and experiences to develop and write main paper to be published in a journal in 3rd year thus in 2nd year the main work and implementation will be done. The idea is to start from Ionescus Framework [**?**] and build on his paper.

- Implement Ionescous framework using the methodology developed in 1st year
- Generalize implementation to market models
- Learn Agda and dependent types
- Dig deeper into equilibrium theory
- Get into market-microstructure: [**?**], [1]
- Dig into *emergent properties* of systems. Can they be formalized?
- Get into basics of Category-Theory [**?**] [3]
- Get into basics of Type-Theory (found good lectures on Youtube)

1.1.3 3rd Year: Finalizing, Publishing & Writing

I plan to be finished - or nearly finished - at the end of the 3rd year. In this year I will finalize the work of the 2nd year, publish the my main journal paper (and optional fun-papers if possible) and will write down the thesis.
To have a bit of distraction and to prevent myself to become too locked in in writing on the thesis I will also work on my optional fun-papers (see below) and hope to at least finish them and maybe publish them - at least I want to present them to 2-3 audiences (e.g. FP Lunch) to test the reaction (especially the Genesis-Paper).

- Finalize research of 2nd year
- Publish journal paper
- Write thesis
- Work on fun-papers

1.2 Papers

This is the list of papers I want to work on. The goals of publishing I set for myself are given beside the paper.

1. *Dynamics of Agent-Based Simulation & Modelling under different Simulation-Semantics.* - Conference paper

2. *Pure Functional ACE (Catchy title yet to be defined)* - Journal paper
3. *Pure by Nature: A Library for pure Agent-Based Simulation & Modelling in Haskell* - Journal paper ?
4. *Time in Games: a Tron Light-Cycle Game in Dunai* - Journal/Conference paper
5. *The Genesis According to Computer-Science: Reality as Simulation of Free Will* - Optional publishing
6. *Pure Functional Islamic Design* - Optional publishing

1.2.1 Dynamics of Agent-Based Simulation & Modelling under different Simulation-Semantics.

The first paper which describes how one can implement ABM/S in Haskell and compares the implementation and results to Java and Akka. A major focus are update-strategies, parallelism, reproducibility, reasoning and comparability between the various implementations.
Actors: The Future in Agent-Based Simulation & Modelling? Although the actor-model is quite old (beginning of the 70s) it seems to have a revival both in Erlang in the 90s and now in the Framework Akka (based on Scala). It is one way of organizing highly parallel (and optionally distributed) applications. Also the actor-model is very close to the agent-metaphor where the latter one was strongly inspired by the former one. Thus It would be very interesting to look closer into how the Actor-Model can be utilized to ABM/S as it seems that this has not been properly done yet.
This paper will establish my methodology in using Haskell / pure functional programming in the 2nd year main work.

1.2.2 Pure Functional ACE (Catchy title yet to be defined)

Is the main work of the PhD and targeted at publication in a Journal. The exact topic and content will be clarified at the beginning of the 2nd year. Mainly it will describe how to implement Ionescus Framework of Gintis trading model and extend it to a more general Market-Model. It will also give an outlook on implementing it using dependent types.

1.2.3 Pure by Nature: A Library for pure Agent-Based Simulation & Modelling in Haskell

This paper describes the ideas and theory behind the implementation of my ABM/S library "PureAgents" in Haskell.

1.2.4 Time in Games: a Tron Light-Cycle Game in Dunai

This paper describes the 2D light-cycle game inspired by the movie Tron implemented in Dunai. It allows to turn back time.

1.2.5 The Genesis According to Computer-Science: Reality as Simulation of Free Will

I've always been interested in a deeper meaning behind things so I want to look into the philosophy and future of simulation: why do we simulate, what can we derive from simulations, what does it say that we humans simulate, what will the future of simulation be?
I claim that our ability to "simulate" in our mind separates our intelligence from those of the animals and that this is a unique property of humans. Also i think the future of simulation will be that humankind will do its own creation/live (artifical life, conciousness) which allows to accurately simulate a given setting - this of course could have ethical implications.
This is fun paper 1.

1.2.6 Pure Functional Islamic Design

Inspired by the paper "Functional Geometry" by Peter Henderson I had the idea to come up with a EDSL for declaratively describing pictures of islamic design which are then rendered using the gloss-library.
This is fun paper 2 - from its focus totally unrelated to the PhD topic but still a great opportunity to learn Haskell, to learn to think functional, to learn to design my own EDSL - thus it may be a great paper to pursue even if I won't finish or produce something publishable.

References

[1] Baker, H. K., Kiymaz, H., Alan, N. S., Bildik, R., and Schwartz, R. Market Microstructure in Emerging and Developed Markets. *Business Faculty Book Gallery* (Jan. 2013).

[2] Bowles, S., Edwards, R., and Roosevelt, F. *Understanding Capitalism: Competition, Command, and Change*, 3 edition ed. Oxford University Press, New York, Mar. 2005.

[3] Spivak, D. I. *Category Theory for the Sciences*, 1 ed. Mit Press Ltd, Cambridge, Massachusetts, Nov. 2014.