Charles University in Prague

Faculty of Social Sciences
Institute of Economic Studies

BACHELOR THESIS

Agent-based modeling of dual currency economy

Author: **Bořivoj Vlk**

Supervisor: **PhDr. Ing. Jiří Skuhrovec**

Academic Year: **2014/2015**

Declaration of Authorship

The author hereby declares that he compiled this thesis independently, using only the listed resources and literature.

The author grants to Charles University permission to reproduce and to distribute copies of this thesis document in whole or in part.

Prague, May 12, 2015

Signature

Bibliography Reference

Vlk, Borivoj. *Agent-based modeling of dual currency economy* Prague, 2015. 38 p. Bachelor thesis (Bc.) Charles University, Faculty of Social Sciences, Institute of Economic Studies. Supervisor: PhDr. Ing. Jiri Skuhrovec

Extent of the Thesis

66 278 (with spaces)

Abstract

Thesis describes close relationship between Dynamic programming and reinforcement learning algorithms on the example of a model of a dual currency economy. Dynamic programming is methodology used for deriving equilibria of Search-Theoretic equilibrium monetary models, which provide evidence for emergence of fiat currency or emergence of internationally circulating currencies without any human institutions. Particular previously published Search-Theoretic framework of dual currency economy is used as a background for development of Agent-based Computational model. Both models are compared based on their ability to reach specified equilibria and their assumptions, with the conclusion, that models are closely related and with the same assumptions would have the same results. Agent-based model also provides possibility of relaxing assumptions on perfect information distribution and static environment. In this setting, the model will reach different equilibria, that correspond better to the real human behavior, observed in previously published laboratory experiments.

JEL Classification	D83, E40
Keywords	Agent-based Modeling, Currency, Currency switch, Dollarization

Author's e-mail	borivoj.vlk@gmail.com
Supervisor's e-mail	jskuhrovec@gmail.com

Abstrakt

Práce popisuje blízký vztah mezi Dynamickým programováním zpětnovazebním učením, na příkladu modelu oběhu dvou měn. Dynamické programování je metodologickým základem pro popis rovnovážných stavů monetárních "Search-Theoretic" modelů, které umožňují vysvětlit vznik fiat měny, nebo mezinárodně používaných měn, bez jakýchkoli společeských institucí. V konkrétním, dříve publikovaném, "Search-Theoretic" rámci dvou měn je vytvořen původní multiagentní výpočetní model. Oba modely jsou porovnány na základě jejich předpokladů a jejich schopnosti dosáhnou určených rovnovážných stavů. Oba modely jsou si blízké a za stejných předpokladů dosahují stejných výsledků. Multiagentní výpočetní model navíc umožňuje vynechat předpoklady dokonalé informovanosti a statického prostředí. Za těchto okolností dosahuje jiných rovnovážných stavů, které ale lépe odpovídají skutečnému lidskému chování, pozorovanému v dříve publikovaných laboratorních experimentech.

Klasifikace JEL	D83, E40
Klíčová slova	Multiagentní modelování, Měna, Změna měny, Dolarizace

E-mail autora	`borivoj.vlk@gmail.com`
E-mail vedoucího práce	`jskuhrovec@gmail.com`

Contents

List of Tables

List of Figures

Acronyms

ABM Agent Based Modeling

VSCS Very Simplified Classifier System

XCS Accuracy Based Classifier System

Bachelor Thesis Proposal

Author	Bořivoj Vlk
Supervisor	PhDr. Ing. Jiří Skuhrovec
Proposed topic	Agent-based modeling of dual currency economy

Topic characteristics Nobuhiro Kiyotaki a Randall Wright in their paper "A search-theoretic approach to monetary economics" introduced a Serach-Theroretic equilibrium model of exchange process that explained the emergence of fiat currency, without human design or any institutions. Their ideas were later used in development of multi-currency random-matching models, that were able to show, how a international currency can emerge without any legal restrictions and without factors such as inflation.
Thesis shall develop two-country, two-currency random-matching model using Agent-based computational economics(ACE). As Marimon-McGrattan-Sargent[1989] worked on the original Kiytaki-Wright concept, I will try to apply ACE on multi currency model and use their experi- ence. The model will consist of rationally acting economic agents and physical environment. Agent meet each other pair-wise according to random matching process and will be divided into two groups. Agents are more likely to meet each other inside their group. Certain proportion of agents in each group will be equipped with special fiat money. Each agent will be able to produce certain good and store that good without any costs. Storage of other agents production goods will not be possible. Trading will be possible only if two agents meet and both agree on trade. Consuming other agents production goods yields utility. Agents act rationally in order to maximize their utility. Sizes of both groups and degree of their economic integration(Represented by probability, that two agent from different groups will meet.) will be key parameters. Exogenous currency exchange machanism will be implemented.

Hypotheses

- Under defined microeconomic properties of the model, can steady state equilibria with two local, two international currencies or one international currency arise?
- Are previously mentioned equilibria pareto efficient?
- Can results of ACE approach and original approach with logical constructions differ? Can ACE give us more possibilities in design of monetary models?

Bibliography A SEARCH THEORETIC APPROACH TO MONETARY ECONOMICS
Nobuhiro Kyiotaki, Randall Wright 1993

CURRENCY EXCHANGE IN A RANDOM SEARCH MODEL
Roulin Zhou 1996

TOWARD A THEORY OF INTERNATIONAL CURRENCY
Kiminori Matsuyamat; Nobuhiro Kiyotaki; Akihilco Matsui. 1991

MONEY AS A MEDIUM OF EXCHANGE IN AN ECONOMY WITH ARTIFICIALLY INTELLIGENT AGENTS
Ramon MARIMON; Ellen McGRATTAN; Thomas J. SARGENT 1989

HANDBOOK OF COMPUTATIONAL ECONOMICS VOLUME 2
Kenneth J. Arrow; Michael D. Intriligator

INTERNATIONAL TRADE AND CURRENCY EXCHANGE
Helene Rey 2001

Chapter 1

Introduction

To explain emergence of fiat money and money exchange without any human institutions, Search-Theoretic equilibrium models were used extensively. First introduced in Kiyotaki & Wright (1989b), they utilize Dynamic programming (Not to be confused with computer programming) in monetary economics. Dynamic programming is mathematical framework developed by Bellman (1957) that is used to find possible equilibria, given model setting, and to analyze it's aspect including welfare implications. Models have usually very solid micro foundations. Only "traditional" assumptions on information and rationality are a necessity. On the other hand, out of equilibrium dynamics of the models is harder to address, due to limitations of Dynamic programming framework. This thesis developes an Agent-based computational model, that provides possible sulution to this issue.
From the original setting of Search-Theoretic monetary models, in which emergence of commodity and fiat currency is studied, researchers went further and addressed the phenomena of international currency in similar manner. One of such models, that tried to address the phenomena of international currencies was developed by Matsuyama *et al.* (1991). It managed to do so, without imposing additional assumptions, although the results were weaker as in other models, like Zhou (1997). This next model imposed additional assumptions on the agents living in the economy - agents sometimes seek foreign goods. This way, international currencies eventually emerged and the results are stronger than in Matsuyama *et al.* (1991). On the other hand, any additional assumption need proper justification. For the sake of simplicity, this thesis will consider primarily model developed by Matsuyama *et al.* (1991)
Even before massive spread of Agent-based Economics (ABE), possible applica-

tion of computational methods based on simulating large numbers of artificially intelligent agents in Kiyotaki-Wright environment was developed by Marimon *et al.* (1990). In this model, the artificially intelligent agents were sometimes unable to reach pareto efficient equilibria, even though the model environment was similar to original (Kiyotaki & Wright 1989b). The agents failed to adopt the speculative strategy (To accept trading good with high storage costs - eventually allowing two goods to circulate as media of exchange.) that was in some setting of Kiyotaki-Wright environment most efficient for some agents. This property became especially interesting after findings of Duffy & Ochs (1999) and Brown (1996), that human subject in laboratory experiment would achieve similar results as artificially intelligent agents of Marimon *et al.* (1990). ABM can bring other benefits in comparison with original approach. It is used to model economic processes as dynamic systems of interacting agents. (Tesfatsion & Judd 2006) It is able to grow general properties of the whole system from the bottom up.

The process that leads to emergence of equilibria in the Kiyotaki-Wright economy is hard to address. There are papers that focus on this topic using advanced theoretical concepts, such as Sethi (1999). But using an ABM model, we can study the process that leads to equilibria more easily and we can also compare in which settings particular types of equilibria may emerge.

In this thesis, I will develop Agent-based computational model, based in two country -two currency environment, in similar setting to Matsuyama *et al.* (1991), on which I will verify some general notions associated with a theory of international currency. I will also address the relation between artificial intelligence algorithm, similar to the one used by Marimon *et al.* (1990), and Dynamic programming. This was done already in computer science, but to my best knowledge, not in economic literature, where it can find interesting application. Economics studies behavior of dynamic systems and artificially intelligent agents can simulate its behavior.

The original goal of the thesis was to develop model with rationally acting agents. Thesis was also originally intended to include exogenous currency exchange mechanism. The assumption of rationality was in the final version omitted, because I was able to provide much the more realistic mechanism of agent's decision-making. Reinforcement learning is used instead, because it is related to both behavioral psychology(Booker & Holland 1989) and Dynamic programming. This mechanism also resulted in necessity to omit inclusion of exogenous currency exchange mechanism, because programming such model would go well beyond my capabilities as a computer programmer and would not bring comprehensive results, that would pay for additional sophistication. Now the thesis offers a comparison between agents, whose decision are based on two frameworks. One of which requires rationality assumption and perfect information and the other can relax those assumptions. Thesis also provides theoretical relation between both of those frameworks, which is surprisingly close. I approach Matsuyama *et al.* (1991) environment in very similar way Marimon *et al.* (1990) approached original Kiyotaki-Wright environment. But since then, the theory of reinforcement learning has made couple of steps forward. On top of just developing the model like Marimon *et al.* (1990) did, I will provide the reader with some very interesting relations between the frameworks used by those who make variants of Kiyotaki-Wright environments and to the one used by Marimon *et al.* (1990). I will also mention some human subject experiments, done mainly by John Duffy - Duffy & Ochs (1999) and Duffy (2004), that may explain the differences in results of both approaches.

Second chapter of this thesis describes Search-Theoretic equilibrium model developed by Matsuyama *et al.* (1991) using original author's notations and mentiones several issues of this model. In order to address those issues, Agents based computational model is developed in next chapters. Both models use similar mathematical apparatus, which is described in chapter 3. In chapter 4 the decision making mechanism of the newly developed Agent based computational model is described. Chapter 5 provides new model environment and practical implementation of decision making mechanism. Results of the execution of computational model are provided in chapter 6 and in appendix to this thesis.

Computer scientists sometimes use methods of reinforcements learning to find numerical solutions to Bellman's equations. In economics, advantage of this relation can be exploited even further. The fact, that the relationship between reinforcement learning in non-cooperative games and Dynamic programming is not fully described is viewed by computer scientists as a big drawback.(Choi & Ahn 2010) On the other hand, what computer scientist may consider to be a problematic, may actually be an advantage for economists. This topic will be addressed in the conclusion along with the relation of reinforcement learning to human subject experiments.

Chapter 2

Matsuyama et al. environment

This thesis develops Agent-based computational model based on the framework by Matsuyama *et al.* (1991). Those authors have adopted Search-Theoretic approach used in Kiyotaki & Wright (1989b) on the model of choices among two fiat monies. This chapter briefly describes their model and methodology they have used. In Chapters 3, 4 and 5, I will describe the setting of my model and the relation between mine and their methodological approach. Here, I have also adopted similar notation, but due to its limited ability to explain mathematical background properly, it should serve only for some general understanding of possible problems, their approach may face. In the next chapters, the notation will have to change.

Matsuyama *et al.* (1991) model world consists of two economies, Home and Foreign, each populated by continuum of rational agents with perfect information about their environment and other agents. Each economy has its own fiat currency. Agents meet pairwise at random and exchange goods for currency. Authors examined by the means of Dynamic programming, in what circumstances both currencies can circulate locally or internationally in both economies.

There were very similar models developed by other authors, such as Wright & Trejos (2001) or Zhou (1997). Those have several additional assumptions, like random change of tastes in Zhou (1997). Those assumptions were a necessity for achieving results, that would imply the existence of one international currency, given the used methodology. For my application, environment with less additional assumptions is better choice, because I want to focus more on the methodology itself.

In the following section, Dynamic programming and stochastic Markov decision process is used as it was by Matsuyama *et al.* (1991) and most other authors of similar models. Those non-trivial mathematical concepts are explained in Chapter 3.

Model setting

Time is discrete and extends from zero to infinity. World is populated by a continuum of rational agents, with perfect information about their environment and other agents, that live infinitely and are divided into two groups representing two economies - Home and Foreign. There are $k \geq 3$ types of agents and k types of indivisible goods. Types of agents are indexed $i = 1, ..., k$. Type i agent gains utility from consuming only type $i - 1$ $(mod\ k)$ good. Type i agent is able to produce only type i good. After agent of type i consumes good $i - 1$ it is able to produce one and only one unit of good i. At this moment, it also gains utility $u > 0$. Each agent knows how to store his production good costlessly up to one unit. He can neither produce nor store any other types of goods. In order to account for time preference, there is also a discount factor $\delta > 0$. Design of the model implies, that there is no double coincidence of wants. The expected utility of an agent at time t is defined as:

$$V_t = E[\sum_{s=0}^{\infty}(1+\delta)^{-s} I_{t+s} u | \Omega_t]$$

Where I_{t+s} is a random indicator function which equals one if agent consumes his consumption good at time $t + s$. Ω_t is the information available in the system and therefore to every agent at period t. Similar definition of expected utility function for my model is in section 4.1 and is preceded by clear derivation.

Let $n \in (0, 1)$ be the size of population living in the home economy and let the $n^* = 1 - n$ be the of size of population living in the Foreign economy.
There are two fiat monies, Home currency and Foreign currency. Each currency is indivisible and can be stored up to one unit by every agent if he does not carry his production good or other currency.
The economy begins at $t = 0$ with agents being endowed with an arbitrary and randomly generated initial distribution of holdings of goods and fiat money. Portion of agents from Home economy is endowed with holdings of goods and and portion of Home agents is endowed with Home currency. Portion of For-

eign agents is endowed with holdings of goods and portion of Foreign agents is endowed with Foreign currency. Which particular agent receives goods and which receives money is determined at random, but the relative portions of money holders and goods holders are fixed. Let M be the fraction of Home agents endowed with Home currency at $t = 0$ and let M^* be the fraction of Foreign agents endowed with Foreign currency at $t = 0$. The fraction of Home(Foreign) agents which are endowed with holdings of goods at $t = 0$ is therefore $n - M; (n^* - M^*)$. Both M and M^* are exogenous parameters. For holdings of Home and Foreign agents, following notations will be used. Let m_h (m_f) be the fraction of Home agents holding Home(Foreign) currency. Let $m^*_f; (m^*_h)$ be the fraction of Foreign agents holding Foreign(Home) currency at time t. Inventory distribution among home and foreign agents can be summarized by two row vectors:

$$X = (1 - m_f - m_h, m_f, m_h)$$

$$X^* = (1 - m^*_f - m^*_h, m^*_f, m^*_h)$$

We can think of those vectors as of random variables, whose distributions are described by agent's decisions and random matching process. Random matching process means, that during each round, agents are matched randomly in pairs and must decide, whether or not to trade. Trade means one-for-one swap of inventories and takes place only if mutually agreed. Agents agree to trade only if it would result in strict increase in expected utility. Home agent meets another home agent with probability n and foreign agent with probability $\beta(1 - n)$, where $\beta \in (0, 1)$ is coefficient that represents the degree of integration of both economies. Therefore, agents living in different economies meet less often. Agents can also meet nobody, that occurs with probability $(1 - \beta)(1 - n)$

In order to maximize overall expected utility, each agent chooses a trade strategy. This strategy is described by a set of rules denoted τ for Home agents and τ^* for Foreign agents. $\tau_{ab} = 1$ if agents agrees to trade object a for object b and zero otherwise. A complete set of rules assigned to each possible combination of tradable object forms agents trade strategy. Tradable objects are production good g, home currency h and foreign currency f

Trade strategies τ^* and τ, inventory distributions X and X^* and matching technology generate the Markov process that that agent's inventory follows.

Let's denote $P_{ab}(P^*_{ab})$ the transition probability with which Home(Foreign)

agent's inventory changes from object a to b. For example the probability that agent's inventory changes from its production good to Home currency is denoted P_{gh} A Complete sets of such probabilities form corresponding transition matrices $\Pi(\Pi^*)$, that form distribution of agent's inventories that form a $MarkovChain$.
For the transition matrices to be defined, Matsuyama *et al.* (1991) had to make restrictive assumptions about the strategies of the agents. Because with no strategy specification it is not possible to define transition matrices. Those assumptions are:

Key assumptions

1. Agents have sufficient knowledge and ability to analyze the game in rational manner.

2. Agents know the entire structure of the game.

3. Agents have agreed on their trading strategies before the first round.

The third one is especially problematic, because it limits the model only to steady-states study. Matsuyama *et al.* (1991) considered a steady-state, pure strategy Nash equilibria, which are a set of strategies, steady state inventory distributions and steady state transition matrices, that satisfy (a) Maximization: Given the steady state inventory distribution and other agents strategies, agents chooses the trade strategy to maximize their overall utility. (b) Rational expectations: the steady state transition matrices and inventory distributions are consistent with the strategies chosen by the agents.

To derive equilibria in their economies, Matsuyama *et al.* (1991) used a framework of Dynamic programming and Bellman's equations. Let V_g, V_h and V_f be the value functions of Home agents, provided that he currently holds his production good, Home currency or Foreign currency, respectively. What is a value function in Dynamic programming and what is Bellman's equation will be shown in Chapter 3. In this case, it is simply expected discounted utility of an agent, given he knows the utility maximizing trade strategy and executes it. Bellman's equation is basically a recursive definition of value function.For this environment the Bellman's equations are:

$$V_g = [(1 - P_{gh} - P_{gf})V_g + P_{gh}V_h + P_{gf}V_f]/(1 + \delta)$$

$$V_h = [P_{hg}(u + V_g) + (1 - P_{hg} - P_{hf})V_h + P_{hf}V_f]/(1 + \delta)$$

$$V_f = [P_{fg}(u + V_g) + P_{fh}V_h + (1 - P_{fg} - P_{fh})V_f]/(1 + \delta)$$

Matsuyama *et al.* (1991) used them to derive several equilibria(and their equivalent variants) and their properties: (A)Two local currencies equilibrium, in which Home currency circulates only in Home economy and vice versa. (B) One local and one international currency equilibrium, in which one of the currencies circulates in both economies. (C)Currency exchange: mixed strategy equilibrium, in which both currencies circulate in both economies. As a result of assumption that agent agrees to trade if and only if it results in strict increase in his expected utility, no currency exchange would occur in previously mentioned equilibrium C .This equilibrium can arise only, if agents are allowed to trade even if they are indifferent.

As we have already described the basics of Matsuyama *et al.* (1991) model, lets recall the assumption on the trading strategies of the agents. The assumption 3 basically means, that the agents decide, which equilibrium will be achieved and follow the appropriate trading strategies - trading strategy profile is assumed to be common knowledge, the agents know, how to coordinate or to focus on a specific equilibrium. This results in the limited ability of the model to explain real world out of equilibrium dynamics. Let's give to this problem special attention, as it will be addressed in the following chapters.

The authors have also tried to relax the third assumption and briefly introduced possible solution: Agents may follow some behavioral patterns, each round small portion of agents may change its behavior on experimental basis, to be the best response to current strategy distribution among population, eventually entire population would reach some steady state equilibrium. Unfortunately, rigorous reasoning was not provided and authors admit, that this is still too vague. There are also other models, that tried to relax this assumptions, but ended up being extremely complicated. Sethi (1999) Agent-based model, in which this assumption would be relaxed is possible solution to this issue. In the following chapter, I will show why was this assumption a necessity required by the used methodology of Dynamic programming. After that, closely related Agent based computational model of similar environment without above mentioned assumptions will be presented.

Chapter 3

Link between models

In order to solve some of the issues of Search-Theoretic model described in the previous chapter, I have developed Agent based computational model, whose setting is very similar, but agents decision making mechanism is replaced by computer algorithm. To clarify reasoning behind the decision to create Agent-based model itself and behind its design, I have to introduce the reader to several non-trivial mathematical concepts, that are used to find solutions of dynamic stochastic optimization problems. I will omit most of the theory behind, in favor of explaining basic notions and principles that should give good understanding of design of both Agent based model and analytical model of Matsuyama *et al.* (1991). I can put the mathematical apparatus behind both models in one chapter, because agent's decision making mechanism in both models is surprisingly similar, as will be described in following chapters. I'm going to provide the logic behind internal functioning of the algorithms of the model, which not only well correspond to the mathematical apparatus employed by Kiyotaki-Wright and others, but also have some very significant advantages. Theory provided here will be utilized in Chapters 4 and 5, which describe the design of agent's decision making mechanism in Agent-based model and can also serve for better understanding of Matsuyama *et al.* (1991) model and Kiyotaki-Wright models in general. In this Chapter, I'm going to put the reasons, why should anyone create a computational model like the one I did and why It was designed the way it was, in context. The model itself is described in chapters 4 and 5.

Let's departure from Matsuyama *et al.* (1991) environment, and focus on a general optimization problem, that an agent in stochastic environment may

face. At first, let's introduce the expected value of random variable using Riemann-Stieltjes integral. The notion of expected value is of special importance when considering stochastic processes.
In the first section, I follow Bartoszynski & Niewidomska-Bugaj (2008).

3.0.1 Riemann-Stieltjes Integral

Let F be a cumulative distribution function of a random variable. F is nondecreasing, continuous on the right and satisfy following conditions:

$$\lim_{x \to \infty} F(x) = 1$$

$$\lim_{x \to -\infty} F(x) = 0$$

Let $a, b \in R$ and $a \leq b$. Lets choose a sequence of partitions of the interval $[a, b]$ so that nth partition divides the interval into 2^n equal parts and that the kth point of the partitions is

$$x_{n,k} = a + \frac{k}{2^n}(b - a),\ k = 0, ..., 2^n$$

Let $g^{(-)}_{n,k}$ be the minimum of function g in the kth interval, that is

$$g^{(-)}_{n,k} = \min_{x_{n,k-1} \leq x \leq x_{n,k}} g(x)$$

Let the function $g^{(+)}_{n,k}$ be the maximum of g in the kth interval, that is

$$g^{(+)}_{n,k} = \max_{x_{n,k-1} \leq x \leq x_{n,k}} g(x)$$

The common limit, if exists of the two sequences

$$\underline{S_n} = \sum_{k=1}^{2^n} g^{(-)}_{n,k}[F(x_{n,k}) - F(x_{n,k-1})]$$

$$\overline{S_n} = \sum_{k=1}^{2^n} g^{(+)}_{n,k}[F(x_{n,k}) - F(x_{n,k-1})]$$

will be denoted $\int_a^b g(x)dF(x)$. It is called Riemann -Stieltjes integral of the function g with respect to F.

3.0.2 Expected value of a discrete random variable

Let F be a cumulative distribution function of a discrete random variable with possible values $x_1, x_2, ..$ and corresponding probabilities $P\{X = x_i\} = p_i$. F is constant between points x_i and its steps equal p_i at points x_i.
The difference $F(x_{n,k}) - F(x_{n,k-1})$ is zero, if the interval $[x_{n,k-1}, x_{n,k}]$ does not include any of the points x_i of the increase of F. Letting $n \to \infty$, the only terms that will remain in sums S_n that will remain will be those corresponding to the points of increase x_i. Therefore, the differences $F(x_{n,k}) - F(x_{n,k-1})$ for intervals covering x_i will converge to p_i and the other terms will converge to zero. Therefore, if g is continuous the limits of the sums S_n will be

$$\int_{-\infty}^{\infty} g(x)dF(x) = \sum_i g(x_i)p_i$$

For the special case where $x = g(x)$ we can see that

$$\int_{-\infty}^{\infty} xdF(x) = \sum_i x_i P\{X = x_i\}$$

I conclude this section with the following theorem:

Theorem 1

If X is a discrete random variable with cdf F and if $E(X)$ exists, then

$$E(X) = \int_{-\infty}^{\infty} xdF(x)$$

3.1 Markov decision process

Consider an agent, facing an optimization problem, in an stochastic environment, that can not be influenced by the actions of other agents and therefore, any distribution function of random variables characterizing the environment, would be without actions of the agent itself, static.
Agent, or a decision maker can by its actions influence a behavior of a discrete stochastic system as it evolves through time so to maximize some reward or to achieve a positive outcome.
Here, I follow Chapter III of Powell (2010).
Consider a discrete system that at any of times $t = 0, 1, 2, ...$ is characterized by a vector p, where $p \in D$ and D is a finite set of possible states of that system.

Key feature of this system is that the state in next period depends only on current state of the system and some transition probabilities, that govern the system from one state to another. The state of the system at the time $t+1$ is therefore independent on the evolution of the state in the past, given current state at time t. Let's denote the probability that the system will be in the state p_i at time $t+1$ if it is in the state p_j at time t - a_{ij}. The states of the system form a $Markov\ Chain$ and matrix $A = (a_{ij})$ its transition matrix.
Let's assume, that the transition probabilities a_{ij} depend on some parameter q which corresponds to agents decision, that the agent can make at any time t. Let $q \in S$ and let S be some finite set of possible decisions. Then, $a_{ij}(q)$ is the probability, that the system will be in state p_i in time $t+1$, given it is in the state p_j at time t and an agent makes decision q. Transition matrix is therefore $A(q) = (a_{ij}(q))$.
As a result of choosing decision q in state p_i, besides the system being transformed to next state p_j that is determined by $A(q)$, agent receives a reward $|r(p_j, q)| \leq \infty$. Let function $r(p, q)$ be defined for $p \in D$ and $q \in S$. So in each state of the system, each decision can result in some reward, which is to be maximized.
Finally, we can define a Markov decision process as a the tuple $< D, S, P, r, \gamma >$ where

- D is the finite set of states p_i of the system.
- S is the finite set of decisions q.
- $A(q)$ is a state transition probability matrix, given the agent's decision q.
- r is some reward function
- γ is a discount factor. $\gamma \in (0, 1)$, that accounts of the time preference of the agent - future rewards are less valuable than those obtained today.

Let the ordered set of decisions $\Pi = \{q_1, q_2..., q_N\}$ be called a policy. By ordered, I mean, that particular decision is assigned to each state of the system as it evolves. Not that the decisions are fixed to particular time periods. Let M be a finite set of all possible policies.
Let's suppose, that the process starts at some time t. First, we need to define, the expected return from $N+1$ periods, given that agent uses some policy Π. Next, let function $dG^{\Pi}(p_{t+i}, p_{t+i+1})$ be the cumulative distribution function of the state of the system in the period $t+i$, $i = 0, ..., N$. That corresponds to

the state transition matrix $A(\Pi)$. Therefore also to the policy Π, and to the current state of the system p_{t+i}. Then, the expected discounted reward over $N+1$ periods given that the agent uses policy Π and the system starts in the state p_t is:

$$R_{N+1}(p_t, \Pi) = \sum_{i=0}^{N}[\gamma^i \int_{p_{t+i \in D}} r(p_{t+i+1}, q_{t+i}) dG_{q_{t+i}}(p_{t+i+1}, p_{t+i})]$$

Utilizing Theorem 1 for given policy Π gives:

$$R_{N+1}(p_t, \Pi) = E^{\Pi}\{\sum_{i=0}^{N} \gamma^i r(p_{t+i+1}, q_{t+i})\}$$

Agent faces a problem of maximizing his expected reward:

$$\max_{\Pi \in M} R_{N+1}(p_t, \Pi)$$

Let the solution of this problem be called an optimal policy and denoted Π^* In line with Bellman's Dynamic programming framework (Bellman 1957), let's define the function $V_N(p_t)$ as expected return over N-periods if the agents uses optimal policy Π^*(Return maximizing policy), given the initial state of the system p_t.

$$V_{N+1}(p_t) = \max_{\Pi \in M} R_{N+1}(p_t, \Pi) = R_{N+1}(p_t, \Pi^*)$$

Function $V_{N+1}(p_t)$ is called a value function and in Kiyotaki-Wright framework is usually denoted V. In the model from chapter 2, it is denoted V_g, V_f or V_h depending on the agent's holdings and therefore state of the system. If the agent would be able to find the value function, he would therefore know also the optimal policy. To find the value function, Bellman's principle of optimality can be used:

An optimal policy has the property, that whatever the initial state and initial decisions are, the remaining decision must constitute an optimal policy with regard to the state resulting from the first decision.

Employing the Bellman's principle of optimality, the expected reward from N-stage process if the agents uses optimal policy is:

$$V_{N+1}(p_t) = \max_{q_t \in S}\{r(p_t, q_t) + E\{\gamma V_N(p_{t+1})\}\}$$

Intuitively this means, that if we choose the decision that yields the maximum reward in first stage and then follow the optimal policy for remaining N stages, with the regard to the resulting state from our first decision, we get optimal policy for $N+1$ stages. Note that Bellman's principle of optimality applies only in Markov processes. For the notion behind this step, I refer to the chapter III of Bellman (1957). For details on formal assumptions and existence and uniqueness of the solution see Bellman (1957) chapter IV. There are several ways, how to solve this problem and how to find the function $V_N(p_t)$ and optimal policy Π^*. I will focus only on one of them, which uses Reinforcement learning methods.
Bellman's principle of optimality also applies to the infinite or unbounded processes, which are usual for Kiyotaki-Wright environments:

$$V(p_t) = \max_{q_t \in S} \{r(p_t, q_t) + E\{\gamma V(p_{t+1})\}\}$$

For this step, I refer to section 3.3 of Powell (2010), also see Bellman (1957) chapter III. We can think of this as a $V(p_t) = \lim_{N\to\infty} V_N(p_t)$ assuming that the limit exists. This way, we obtain something like a steady state $V(p_t)$ and steady state policy. Existence and uniqueness theorems can be found also in chapter IV of Bellman (1957).
This last equation is being referred to as Bellman's equation.

Now let's recall the key assumption 3 from chapter 2. What if it would be broken? What if the system actually was influenced by the decision of other agents that would not be agreed on the trading strategy and therefore unpredictable? In this case, the transition matrix $A(q)$ would not be defined, therefore we would not be able to compute the expected value and the Bellman's equation could not be used to find optimal policy.
But the agent still could use some approximation of the value function. This topic is addressed in the following chapter.

Chapter 4

Reinforcement Learning

Stuart Dreyfus in the introduction to the 2010 edition of Richard Bellman's Dynamic Programming (Bellman 1957) states, that there is a relationship between Bellman's principle of optimality and reinforcement machine learning in which computer algorithm is used to find optimal policy controlling either deterministic or stochastic sequential process. The algorithm is given only the information resulting from the observation of the results of the various decisions in various states. It is not given any information about decisiondependent evolution of the state or about the determination of the reward. The algorithm is trained on set of state-decisions pairs and corresponding results to find the optimal policy in policy space. This basically means, that the algorithm from the experience successively approximates the optimal policy function and value function of the dynamic programming. One of such algorithms is Holland's classifier system. Before I can get to the relation of Holland's classifier system to dynamic programming and its application in my model, it is necessary to present a concept from which it is derived, called Q-learning. this concept will also serve as a proxy, between methods that are actually employed in my model, and methods used by Matsuyama *et al.* (1991) and others.

4.1 Q-learning

In this part, I follow chapter XIII of Mitchell (1997).
Let's assume, that an agent wants to find optimal policy and earn highest possible return, but he doesn't have a complete knowledge about the nature of the system, transition rules or reward function.
Here again introduce the value function $V(p_t)$, which will be again defined as a

expected discounted value of future rewards over a infinite process, given initial state of the system p_t and agents optimal policy Π^*:

$$V(p_t) = E^{\Pi}\{\sum_{i=0}^{\infty} \gamma^i r(p_{t+i}, q_{t+i})\}$$

The function $V(p_t)$ is closely related to the same function presented in Chapter 2. The agent would like to know the optimal policy and the corresponding value function. If he is able to compare expected rewards each time he makes a decision and choose the one that is most likely to bring good results, he can eventually discover the optimal policy. But in real world, an agent would often only be able to observe rewards and state variables. Unfortunately, he cannot obtain a solution Bellman's equation without the knowledge of transition matrix and, therefore the cumulative distribution function $dG(p, q)$ and the reward function $r(p, q)$. So if he cannot use the true value function to evaluate his policy, he might try to use some approximation.

4.1.1 Q-Function

One method of approximating the value function for given policies is called Q-Learning. This method tries to approximate the true optimal policy value function by introducing function that is to it directly related and from which it derives its name, Q-function.

$$Q(p_t, q_t) = E\{r(p_t, q_t) + \gamma V(p_{t+1})\}$$

Notice also this relation between the Q-function and the value function:

$$V(p_t) = \max_{q_t} Q(p_t, q_t)$$

Therefore, we can write:

$$Q(p_t, q_t) = E\{r(p_t, q_t)\} + \gamma \max_{q_{t+1}} Q(p_{t+1}, q_{t+1})$$

Learning the Q-Function corresponds to learning the optimal value function and, therefore optimal policy. Another big advantage is, that the agent is able to learn the optimal policy, without any knowledge of reward function of about the cumulative distribution function.

At each stage of the process, the agent tries to approximate the actual value of the Q-function and makes a decision based on that approximation. Eventually, using the algorithm that is presented below, he will have a very good estimate. Before I explain the algorithm, that is used to learn the Q-function, I need to define function, that represents the agent current hypothesis about the true value of the Q-function. This hypothesis will be denoted $\hat{Q}(p, q)$. Agent store the estimate $\hat{Q}$ for each state-decision pair in a table. Initially, the table can be filled with random numbers or zeros. At each stage of the process, the agent observes the state of the system p_t, makes a decision q_t, receives a reward and observes next state of the system p_{t+1}. Then he updates the entry for $\hat{Q}(p_t, q_t)$ in his estimate table according to the rule in Equation 1. Also, let's define the number $n(I, p_t, q_t)$ representing the number of iteration, in which the decision p_q is applied to the state q_t the I-th time.

Equation 1

$$\hat{Q}_n(p_t, q_t) = (1 - \alpha_n)\hat{Q}_{n-1}(p_t, q_t) + \alpha_n\{r(p_t, q_t) + \gamma \max_{q_{t+1}} \hat{Q}_{n-1}(p_{t+1}, q_{t+1})\}$$

Where

$\alpha_n = \frac{1}{\sum_{s=1}^{n} I_{q_t}^{p_t}(s)+1}$ and $I_{q_t}^{p_t}(s) = 1$ if state decision q_t was applied on state p_t at time s and zero otherwise.

Q-learning algorithm

For each p, q initialize the table entry $\hat{Q}(p, q)$ to zero
Observe the current state p_t
for $i = 0$ to ∞:

1. Select a decision q_{t+i} and execute it
2. Receive the immediate reward r
3. Observe the new state p_{t+i+1}
4. Update the table entry for $\hat{Q}(p_{t+i}, q_{t+i})$ as in equation 1.

Finally, I can present a core theorem, that provides a direct relation between my computational model and work of Matsuyama *et al.* (1991).

Theorem 2 - Convergence of Q-learning for stochastic Markov decision process

Consider a Q-learning agent in a stochastic Markov decision process with bounded rewards $\forall p \in D, \forall q \in S, c \ \in (0, \infty) \ : |r(p,q)| \leq c$. The Q-learning agent uses the training rule of equation 1, initializes its table $\hat{Q}(p, q)$ to arbitrary finite values, and uses a discount factor γ such that $0 \leq \gamma < 1$. Let $n(i, p, q)$ be the iteration corresponding to the ith time that decision q is applied to the state p. If each state-action pair is visited infinitely often, $0 \leq \alpha_n < 1$, and

$$\sum_{i=1}^{\infty} \alpha_{n(i,p,q)} = \infty, \ \sum_{i=1}^{\infty} \{\alpha_{n(i,p,q)}\}^2 \leq \infty$$

then for all $p \in D$ and $q \in S$, $\hat{Q}_n(p, q) \to Q(p, q)$ as $n \to \infty$ with probability 1.

For proof see Watkins & Dayan (1992).
The intuition behind this theorem is, that the Q-Learning agent, in an environment with the same usual Kyiotaki-Wright assumptions (Including Key assumptions from chapter 2) is proven to make very good estimates of value function with large n and therefore the optimal policy Π^*.

4.2 Nash Q-learning

Even though the agent doesn't know how the next state of the system or the reward is determined, he will eventually make his hypothesis about the Q-

function and therefore about the value function of Dynamic programming and his hypothesis will converge over time the true value with probability 1.
But, there is still one assumption that limits application of the presented principles. It is especially limiting, when we would want to apply this methodology in economics. That is, both reward function $r(p, q)$ and transition matrix that determines the next state of the system given agents decision are static over time. But in real economic systems, that is not true. It is determined by the decisions of other agents in the system. Proving convergence of non-cooperative multi-agent stochastic game to Nash equilibrium is very difficult and to my best knowledge it has not been done yet. This is one of the reasons, that application of reinforcement learning in economics is sometimes considered stuck. (Choi & Ahn 2010) There were some attempts to design convergence proof for a system, where transition matrix is determined by other players in the game and those decisions are determined by some algorithm that is based on Q-learning. One of them is called Nash Q-learning (Hu & Wellman 2003). In this method, the Q-function is modified. Newly introduced Nash-Q-function is defined as follows: Sum of agents current reward plus the sum of agents future reward when all agents follow a same Nash equilibrium strategy. This is quite restrictive in application in economics. It adds again some of the assumptions we were trying to remove in the first place. On top of that, Nash-Q-function is guaranteed to converge to its true value only in some very restricted environments.
So if one agent is in a stationary environment, where the transition matrix is given a does not evolve over time, the Q-function is guaranteed to converge. But when there are multiple agents in non-cooperative stochastic game using Q-learning, they are not guaranteed to learn the optimal strategies - the Q-function is not guaranteed to reach its true value and to my best knowledge - no one introduced modification of the Q-learning algorithm that would not bring too restrictive assumptions and would be guaranteed to make agents in non-cooperative stochastic games reach Nash equilibrium.

4.3 Machine learning algorithm

For the development of the computational agent-based model, I've decided not to go for Q-Learning but to use Holland's Classifier systems instead (Holland 1975). Mainly because it fits my purpose perfectly, it is easier to implement for Kiyotaki-Wright environments and it this application will bring same results. My approach is quite similar to the one used by Marimon *et al.* (1990). But because I need some relation of Holland Classifier system to Q-Learning, in order to have a relationshit to the value function of Dynamic programming, I've used VSCS - Very Simplified Classifier System (Dorigo & Bersini 1994). There also are many other kinds of classifier systems. For example, XCS (Wilson 1995) is used in more complicated environments. Marimon *et al.* (1990) used traditional Hollands Classifier system, but this was two year earlier to the proof of convergence by Watkins & Dayan (1992). Since the proof works for Q-Learning agents, I need some version of Classifier System, that is interchangeable with Q-Learning algorithms. That is exactly VSCS. It was developed by Dorigo & Bersini (1994) for that purpose and theorem 2 will still apply - key property of classifier system corresponds to the Q-function. But still, I've made a small change in the VSCS. Because Dorigo & Bersini (1994) developed VSCS to correspond to the deterministic variant of Q-learning, I've made my version of VSCS to correspond to the stochastic variant of Q-learning. That doesn't mean, theorem 2 should apply to the agents in the simulations in chapter 6 (It applies on one decision maker in Stochastic Markov decision Process), but becouse the simulation consists of system of many decision makers its assumption are clearly broken, transition matrix will change in time, but that is the whole point.

Chapter 5

Model Environment

This chapter describes the environment of the new Agent-based computational model. It is based on modified Matsuyama *et al.* (1991) environment. From now on, I will use the very similar notation to Marimon *et al.* (1990). The reader can therefore easily compare both models and their differences.

Time is discrete and extends from 0 to $T \in N$. Economy is populated by a finite number of agents $a = 1, ..., A$ divided into two groups. There are $k \in N$ types of indivisible goods. Each agent can produce and consume goods as in Matsuyama *et al.* (1991). If agent consumes his consumption good, he receives utility $u \in (0, \infty)$. Each round each agent also suffers life cost $s \in [0, \infty)$. Agents meet randomly each round according to random matching technology. There is a fixed probability $\beta \in [0, 1)$, that an agent will meet some agent from the other economy. Otherwise, he will meet randomly selected agent from his economy. At $t = 0$, fraction of agents in Home economy is endowed with Home currency $M \in (0, 1)$ and fraction of agents in the Foreign economy are endowed with Foreign currency $M^* \in (0, 1)$. The rest of the agents are endowed with their production goods. Agent's a holdings at time t are denoted x^+_{at}.

5.1 Application of VSCS

At each round, after random matching is performed. Agent gets a message that specifies what are his holdings and what are holdings of his counterparty. Then, the classifier system is activated. its purpose is to determine agent's decision. Booker & Holland (1989) defines classifier system as a "parallel, message-passing, rule-based system wherein all rules have the same simple form. In the simplest version all messages are required to be of fixed length over a specified alphabet, typically k-bit binary strings. The rules are in condition/action form. The condition part specifies what kind of messages satisfy (activate) the rule and the action part specifies what message is to be sent when the rule is satisfied.". In my application of VSCS alphabet consist of values 1 or 0 and a classifier is a condition/action rule. Each condition part of each classifier is exactly $2(k+2)$ bits long, where k is number of types of goods in the economy, in is increased by the factor 2 in order to account for two fiat monies. Set of classifiers for agents a denoted D_a consist of classifier whose condition part can match every state of the world that the agent can observe (its holdings and its trading counterparties holdings) and two possible decisions that the agent can make - accept trade or refuse trade. For example, a condition part of a classifier that would match the message that agent is holding good 1 and his counterparty is holding good 2 would look like this: $1, ..., 0|0, 1, ..., 0$. Therefore, if agent holds good 1, that in the first bit would be value 1. His counterparty holds good 2, so at the $k+3$-bit would be also value 1. All the other bits will contain value 0, because both of them don't hold any other good except for good 1 and 2, respectively. That uniquely represents holdings of both agents. To the condition part would be also assigned one of the two possible decisions(actions). Trade 1 or not to trade 0. The complete set of classifiers for my model looks like this:

$$\begin{gathered}
\{1\ 0\ 0\ ...0|1\ 0\ 0\ ...\ 0|1\}, \ldots, \{1\ 0\ 0\ ...0|0\ 0\ 0\ ...\ 1|1\} \\
\{1\ 0\ 0\ ...0|1\ 0\ 0\ ...\ 0|0\}, \ldots, \{1\ 0\ 0\ ...0|0\ 0\ 0\ ...\ 1|0\} \\
\{0\ 1\ 0\ ...0|1\ 0\ 0\ ...\ 0|1\}, \ldots, \{0\ 1\ 0\ ...0|0\ 0\ 0\ ...\ 1|1\} \\
\{0\ 1\ 0\ ...0|1\ 0\ 0\ ...\ 0|0\} \ldots, \{0\ 1\ 0\ ...0|0\ 0\ 0\ ...\ 1|0\} \\
\vdots \qquad \ddots \quad \vdots \\
\{0\ 0\ 0\ ...1|1\ 0\ 0\ ...\ 0|1\}, \ldots \{0\ 0\ 0\ ...\ 1|0\ 0\ 0\ ...\ 1|1\} \\
\{0\ 0\ 0\ ...1|1\ 0\ 0\ ...\ 0|0\}, \ldots \{0\ 0\ 0\ ...\ 1|0\ 0\ 0\ ...\ 1|0\}
\end{gathered}$$

It includes every combination of state of the world that the agent can observe and every decision it can make. Classifier system in this form for agent a will be denoted E^a For each state of the world there are two classifiers. One with decision part containing 1 (Trade) and one with 0 (Not Trade) Let

$e = 1, 2, ..., 2(k + 2)^2$ index this collection of classifiers. (Total number of classifiers is $2(k + 2)^2$)
Assigned to each classifier $e \in 1, 2, ..., 2(k + 2)^2$ is a strength, denoted $S_e^a(t)$ - The strength of classifier e in classifier system of agent a at the time t. The strength $S_e^a(t)$ evolves over time in a way determined by accounting system or as it is called by Booker & Holland (1989) - "a Bucket Brigade Algorithm". The strengths attached to classifiers are used to determine the decision made by the classifier system at time t. In opposite to Kiyotaki & Wright (1993) environment used by Marimon *et al.* (1990), in the model environment developed by Matsuyama *et al.* (1991) only one set of classifiers will be used. Marimon *et al.* (1990) defined two sets of classifiers, consumption set and exchange set. Since agents living in Matsuyama *et al.* (1991) environment can carry only their production good or fiat money, they don't need to decide whether to consume or not, they will consume automatically once they attain their consumption good, thus, only one set of classifiers is needed. This is also necessary for the classifier system to be considered VSCS.
How is the decision of an agent at each state determined? For each given state there are two classifiers whose condition parts are satisfied. Those classifiers will compete in an auction. Both classifiers will place a bid, which is determined by the bid function. The winner, that places the highest bid, makes agent's decision. For this, we need to define the bid function:

$$Bid(e, t) = bS_e^a,\ b \in (0, 1)$$

This function computes the bids that the classifiers will place in an auction. The size of the bid is portion $b \in (0, 1)$ of classifier strength.

5.1.1 Bucket Brigade Algorithm

Classifier system can produce a series of decisions, that will eventually result in some positive payoff to an agent. Particular decisions in that series don't have to result in payment themselves, but still agent can learn the whole series. This is thanks to the so-called "bucket brigade algorithm". The winning bid will be deducted from the strenght of winning classifier and added to the strength of the classifier that made the last decision. The classifier that won the auction at $t - 1$ is credited for setting the observed state of the world by agent a at time t. Therefore, it will receive the part of the payoff, in which this state may eventually result. To sum up, the evolution of the classifier strength is described by the following difference equation, which corresponds to the updating equation of the Q-learning algorithm. Imagine this as a series of Q-functions for each classifier. Strength of each classifier is constantly changing over time and represents agents hypotheses about the expected reward from executing decision part of that classifier. Strengths of the classifiers are therefore somehow related to the value functions of Dynamic programming - are agent's guess about those functions.

Equation 2

$$S_e^a(t+1) = S_e^a(t) - \gamma\alpha_e^a(t)[bS_e^a(t) - \sum_e I_e^a(t-1)bS_e^a(t-1) - U_a]$$

Where $I_e^a(t) = 1$ if classifier e makes the decision at time t and zero otherwise. The term α_e^a corresponds to the α_n term of Q-learning, equation 1. Therefore $\alpha_e^a(t) = \frac{1}{\sum_{s=0}^t I_e^a(s)+1}$.
Sometimes, the strength of the classifier is updated by external payoff, that represents utility gain: $U_a = u - s$ if x_{at}^+ is agents consumption good and $U_a = -s$ otherwise. This accounts for utility gains and life costs. If the model is configured without life cost - that may make the agents agree on trades in which they are indifferent - then $s = 0$. Term $\gamma \in (0,1)$ is a discount factor, that accounts for time preference.
This difference equation should according to Dorigo & Bersini (1994) correspond to the Equation 1 of Q-Learning. When one agent would live in a stationary environment, where decisions of other agents are fixed, Theorem 2 should apply to the classifier strengths, which would be therefore related to the average expected payoff of the classifier decision part. Simply if the assumption 3. from chapter 2 would apply, classifier strength approximates the value function of dynamic programming.

5.1.2 Equilibrium definition

In order to formally analyze the model results, I define the Nash-Markov Stationary equilibrium. Model is expected to reach steady state in most of its configurations, but whether those steady state will be Nash-Markov equilibrium is not clear. The definitions are the same as the ones introduced by Marimon *et al.* (1990). Let G be a set cumulative probability distribution functions $dG_a,\ a = 1, ..., A$, that corresponds to the transition matrices, that define the probability of changing agent's a holdings from it's production or consumption good to Home of Foreign currency and vice versa.

Optimal set of classifiers for agent a

Given the set cumulative probability distribution functions G and fixed sets of classifiers of the other agents $D_{a'}$ for all $a' \neq a$, a fixed set of classifiers is said to be optimal for agent a if there exists no other set $\hat{D}_a$ which yields higher long-run average utility.

Stationary Nash equilibrium

A stationary Nash equilibrium is a set G of cumulative probability distributions functions dG_a and fixed sets of classifiers $D_a,\ a = 1, ..., A$ such that:

1. Given G and $D_{a'}$, for $a' \neq a$, D_a is optimal for agent a.
2. $\{D_a, a = 1, ..., A\}$ and the random matching technology implies that $G = G_t$ for all t

If only the second condition is satisfied, the model reached steady state, but not Nash-Markov equilibrium.

5.2 Genetic algorithm

In order to enable some information sharing among agents, I've designed simple genetic algorithm. By default, this genetic algorithm was turned off, because it causes the model to reach different equilibria, when the model setting is more comlicated. More on that, in the next chaper.
This algorithm is different from the genetic algorithm that was developed by Booker & Holland (1989). Each agent is assigned a value that measures his success in pursuing utility gains till time t. This value is called fitness and is defined as follows: $Fitness_a(t) = \sum_{s=0}^{t} \gamma^{t-s} U_a(s)$, which is discounted flow of all utility gains till time t. At each round, there is certain probability, that the agents with lowest fitness will be replaced with another agent of the same type, whose classifier system is a random combination of the classifier systems of the two most successful agents of his type. We can see this a form of information sharing among agents of certain types.

Chapter 6

Simulation Results

This chapter provides commented results from running Agent based computational model. Throughout the entire simulation, the most important data was collected and it is presented in figures in this chapter and in appendix to this thesis.
The model was developed in Matlab 2010b. This brings some limitation, model requires more computational power than it would when it would be written in some language like C. On the other hand, the code is easily readable and analysis of the results is also convenient. Source code can be found in appendix B - contents of an enclosed DVD.
It was necessary to somehow limit number of agents, rounds, and types of goods. Usual configuration was 120 agents, 6 types of goods and 5000 rounds. Simulation in this configuration took about 70 minutes on IBM T60 laptop made in 2007, running Debian GNU/Linux Jessie. Due to the time demading nature of the simulations, only one simulation for each configuration will be presented.

In the first four simulations, key assumptions from chapter 2 were relaxed to be applied again in the fifth and last one. The results of both models are compared. Theorem 2 schould therefore apply only to the last fifth simulation.

6.1 Testing simulation

In the testing configuration, only 2 agents live in each economy and there is no integration between economies - agents meet only inside their own economy. One agent in each economy is endowed with currency, the other one with his production good. Complete configuration can be found in Table 6.1. All agents will eventually accept production good for currency, and in the next round accept consumption good in exchange for currency. Strength of particular classifiers responsible for those decisions are depicted in Figures 6.1 and 6.2. Figure 6.1 shows the strength of classifier that is responsible for accepting consumption good in exchange for Home currency for one of the Home agents. Clear convergence can be seen. Figure 6.2 shows the same properties for the classifier that is responsible for accepting Home currency in exchange for production good. Again, clear convergence. Figure 6.3 shows average Fitness of agents in Home economy. The value is discounter and converges. Therefore, the other agent has similar strength of his classifier. The Home economy achieved stationary Nash Equilibrium.

Table 6.1: Model Configuration

k	Types of Goods	2
n	Size of Home population	2
n^*	Size of Foreign population	2
β	Degree of integration	0
M	Fraction of Home agents holding Home currency	0.5
M^*	Fraction of Foreign agents holding Foreign currency	0.5
γ	Discount rate	0.9
b	Bid portion	0.5
s	Life Cost	YES
T	Rounds	500
	Genetic Algorithm	NO
	Preset optimal strategy	NO

6.2 Two Local Currencies equilibrium

Next, model configuration was modified. Most importantly, more agents were added. Now there are 60 in each economy and there are 6 types of goods. Table 6.2 shows the complete configuration. There is still no integration between economies, so both currencies will circulate only locally. One Home agent was chosen as an example of the evolution of classifier strengths. Also, average fitness of agents was captured in order to account for the behavior of the whole system. In the graphs, full line depicts the strength of classifier that is responsible for the decision to trade and dashed line depicts the strength of the classifier that is responsible for not accepting trade. If the full line is above the dashed one, agent accepts trade. From Figure 6.5 and Figure 6.6 we can

see that our agent follows optimal strategy. It accepts Home currency in exchange for it's production good and accepts its consumption good in exchange for Home currency. From the graphs that depict average utility I conclude, that this behavior is shared among other agents as well and therefore system achieved stationary Nash equilibria. Time series of average fitness becomes stationary after several rounds.
In this configuration, the model is able to achieve stationary Nash equilibria.

Table 6.2: Model Configuration

k	Types of Goods	6
n	Size of Home population	60
n^*	Size of Foreign population	60
β	Degree of integration	0
M	Fraction of Home agents holding Home currency	0.5
M^*	Fraction of Foreign agents holding Foreign currency	0.5
γ	Discount rate	0.9
b	Bid portion	0.5
s	Life Cost	YES
T	Rounds	5000
	Genetic Algorithm	NO

6.3 Currency Exchange

Next, same configuration was used, but the degree of integration was set in such way, that both currencies should circulate internationally. Every agent has the same probability that he meets agents from his economy, or from the other one. Matsuyama *et al.* (1991) showed, that the optimal strategy for all agents in this configuration is to accept both currencies. My model will show, whether agents approximating Dynamic programming value function by the strength of classifiers, will reach the same equilibria. It also relaxes the "key assumptions" from chapter 2. In Matsuyama *et al.* (1991) environment, the exchange of currencies occurs, only when agents accept threde when they are indifferent. This time, I also provide graphs of currency holdings. Black line depicts the share of Home agents that hold their native currency and grey line depicts the share of agents holding currency that has been originally used in the other economy. In all preceding configurations, currencies circulate independently on agents nationality. Strengths of classifiers, average fitness and agents holdings are depicted on figures provided in appendix A. Full line in figures depicting classifier strength represents the strength of the classifier that is responsible for accepting trade and dashed line represents refusing to trade. Figure A1-A7 depicts the strategy of one of the Foreign agents. In Figure A.2 we can see, that he accepts Home currency in exchange for his production good. But Figure A.4 shows, that he is not willing to accept Foreign currency in exchange for his production good. Figure A.8-A.14 shows strategy of one of the Home agents. He accepts both currencies in exchange for his production good and therefore, in opposition to Foreign agent, follows optimal strategy. Home agent also accepts currency exchange even when it doesn't increase his expected utility gains and he is indifferent between accepting and not accepting. The Foreign agent does the opposite. From this I conclude, that all agents learned strategy that yields some utility, but only some of them learned the optimal strategy. The system did not achieve stationary Nash equilibrium. At this point, I would like to note, that I've runned the simulation many times with different configurations and even more modified versions of the model. Most of the time the convergence to Nash equilibria was not achieved. This configuration is a result of my best effort to make it converge to Nash equilibria, while not breaking any *more* assumptions of Theorem 2. To my best knowledge, there is no proof of convergence of classifier strengths to the optimal values that would apply on my model. Classifier strengths are also as I've shown related to the value functions of Dynamic programming. From this I conclude, that with relaxing some of the usual assumptions of Search-Theoretic models I've mentioned to be problematic, convergence to Nash equilibria probably won't be achieved. especially when the "world" has many states and optimal strategy gets more complicated.

6.3.1 Currency Exchange and Genetic algorithm

I have used exactly the same configuration as in previous simulation, but this time, the genetic algorithm was active. This will enable some information sharing among agents and then hopefully, those who learned the optimal strategy and therefore have higher expected fitness, will pass some of their knowledge to the less successful ones.

In Figures A.20 -A.26 and A.27 - A.33 strategies of one Home agent and one Foreign agent are depicted. We can see, that both of them reached the optimal strategy. I have run the model several times with same results. Average fitness

Table 6.3: Model Configuration

k	Types of Goods	6
n	Size of Home population	60
n^*	Size of Foreign population	60
β	Degree of integration	0.5
M	Fraction of Home agents holding Home currency	0.5
M^*	Fraction of Foreign agents holding Foreign currency	0.5
γ	Discount rate	0.9
b	Bid portion	0.5
s	Life Cost	YES
T	Rounds	5000
	Genetic Algorithm	NO
	Preset optimal strategy	NO

became again stationary after several rounds. Some agents accept currency exchange when they are indifferent, some don't. Therefore I conclude, that this time the system probably reached stationary Nash equilibria and agents follow same optimal strategy as in Matsuyama *et al.* (1991). It makes sense, that when some information sharing is possible in the economy, agents may be able not only to learn a strategy that yields some payoff, but also a strategy, that yields the highest possible payoff. Like in real world, information means profit and good information distribution in the economy mean more efficiency. On the other hand, my genetic algorithm has no support in current theory and I'm not able to provide any relationship to the Search-Theoretic models. I also recall once more, that my genetic algorithm in no way related to the one developed by Booker & Holland (1989).

6.3.2 Currency Exchange and Preset Strategy

In the last experiment with this type of configuration, I've tried to simulate the assumptions about the environment made by Matsuyama *et al.* (1991) and others. Especially the assumption 3 in chapter 2. This time, all classifiers of all agents have preset strengths. Intuitively, this means, that agents agree on trading strategies exactly as in key assumption 3. The classifier strengths are set to the values that correspond to the optimal strategy for this type of configuration. There is one agent in Home economy, that has not his classifier strength preset and starts with zeros as previously. All agents, except for one, agree on which equilibrium is being played. Figure A.39-A.45 shows the devel-

Table 6.4: Model Configuration

k	Types of Goods	6
n	Size of Home population	60
n^*	Size of Foreign population	60
β	Degree of integration	0.5
M	Fraction of Home agents holding Home currency	0.5
M^*	Fraction of Foreign agents holding Foreign currency	0.5
γ	Discount rate	0.9
b	Bid portion	0.5
s	Life Cost	YES
T	Rounds	5000
	Genetic Algorithm	YES
	Preset optimal strategy	NO

opment of strengths of classifiers of the one agent without preset strengths. We can see, that he also achieved optimal strategy, that the rest of both economies was already using. This time, Theorem 2 should apply on that one agent. Indeed, he found the optimal strategy. In Figure A.46 and A.47 we can see that average level of agents fitness reached stationary point immediately after the start of the simulation. That is because agents did not have to learn optimal (or any other that yields some utility) strategy. The system reached stationary Nash equilibria once the agent without preset strategy learned optimal strategy. This is all in line with usual findings of models based on Kiyotaki-Wright methodology.

Table 6.5: Model Configuration

k	Types of Goods	6
n	Size of Home population	60
n^*	Size of Foreign population	60
β	Degree of integration	0.5
M	Fraction of Home agents holding Home currency	0.5
M^*	Fraction of Foreign agents holding Foreign currency	0.5
γ	Discount rate	0.9
b	Bid portion	0.5
s	Life Cost	YES
T	Rounds	5000
	Genetic Algorithm	NO
	Preset optimal strategy	YES

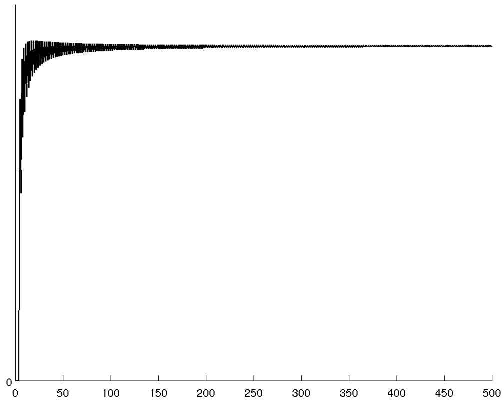

Figure 6.1: Home currency for consumption good

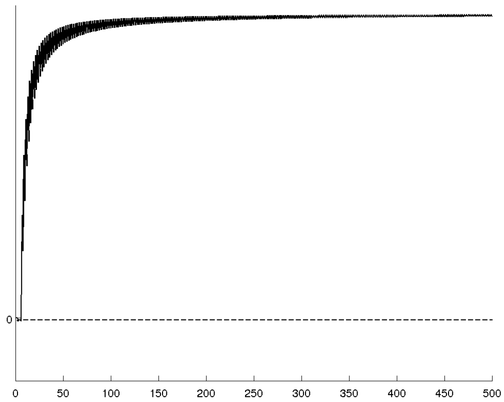

Figure 6.2: Production good for home currency

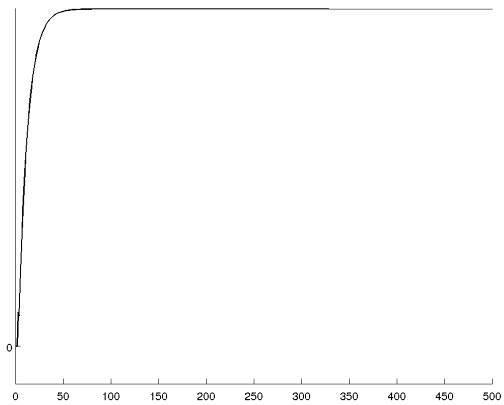

Figure 6.3: Average fitness - Home economy

Figure 6.4: Test Results

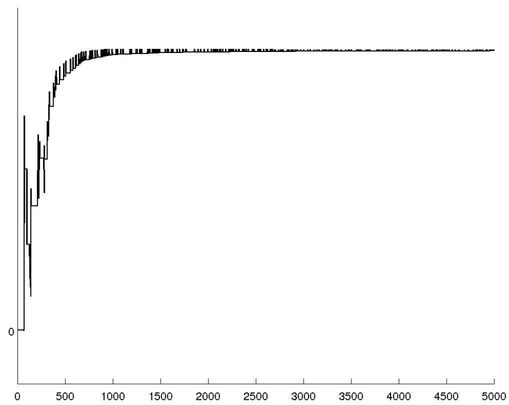

Figure 6.5: Home currency for consumption good

Figure 6.6: Production good for home currency

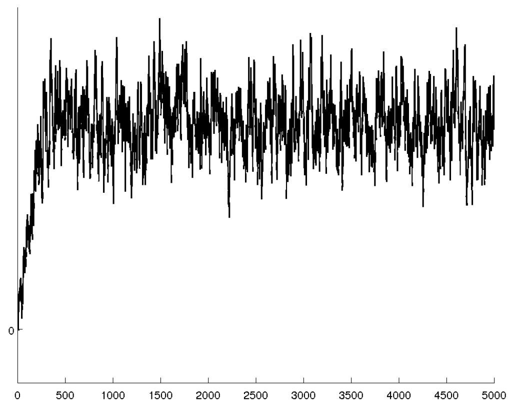

Figure 6.7: Average fitness - Home economy

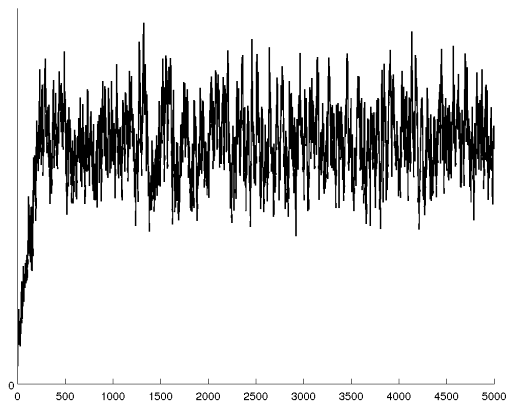

Figure 6.8: Average fitness - Foreign economy

Figure 6.9: Two local currencies - No preset strategy

Chapter 7

Conclusion

In this thesis, I've developed an Agent-based computational model with artificially intelligent agents living in Matsuyama *et al.* (1991) environment. I have adopted the similar methodology as Marimon *et al.* (1990), but on top of that, I've provided theoretical relation between reinforcement learning and Dynamic programming. There is direct relation between strength of particular classifier in VSCS and value functions of Dynamic programming. Therefore, artificially inteligent agents in my model are approximating behavior of perfectly informed agents in conventional Kyiotaki-Wright environment. Their behavior is determined by reinforcement learning decision mechanism, that may also approximate human behavior. I have shown, that artificially intelligent agents may fail to reach stationary Nash equilibria. In simple environments, such as the one with two local currencies, my results confirm that equilibrium as described by Matsuyama *et al.* (1991) will arise. In contrast -In two international currencies setting, some agents failed to learn optimal strategy. When one agent will be put in play where the rest of the economy already reached equilibrium described in Matsuyama *et al.* (1991), he will succeed in learning optimal strategy. Similar results will be reached when we allow for addition information sharing among agents in the form of genetic algorithm.

If we impose the same key assumptions on Agent-based computational model as it is done on models in Kyiytaki-Wright environments, the model will reach same equilibria. In constrast, relaxing those assumptions in more comlicated environments will lead to different results, althought the agents in both models use similar decision machanism. In Kyiotaki-Wright framework, agents use value functions of Dynamic programming and in Agent-based model, agents use some approximation of those functions in form of classifier strenghts.

If the relaxing of key assumption 3 from chapter 2 leads to different results in Agent-based model, the ability to explain real world equilibria of Matsuyama *et al.* (1991) model and other Kyiotaki-Wright models is questionable.

7.1 Relation to human subjects

There is extensive literature on the relations between economics and psychology (Rabin 1997). Brown (1996) conducted a laboratory experiment, whose setting was as close as possible to the setting of Marimon *et al.* (1990) model. Concluding, that his subjects failed to adopt speculative strategies in the cases, when they were necessary to reach optimal strategy in Kiyotaki & Wright (1989b) environment. John Duffy took this idea further, he focuses in some of his work on relation of human subjects and reinforcement learning in economic application. In Duffy & Ochs (1999) he covers the same Kiyotaki & Wright (1989b) environment, but also adds to the setting some information sharing among agents, in this case human subjects. He conducted 25 experiments on 636 subjects with very similar results. He provided the subjects with information on historical average proportions of goods held by each player type in the population, but with no effect on players ability to attain optimal strategy.

The relationship between reinforcement learning and human subjects experiments has in my opinion great potential, especially in economics. "Surprisingly, the predictions of Q-learning models have yet to be compared with data from controlled laboratory experiments with human subjects — a good topic for future research." (Duffy 2004) Unfortunately, to my best knowledge, even today there is still no paper, that would provide such research and that would be anyhow related to economics.

7.2 Possible topics for further research

In macroeconomic theory, modern models are formulated as optimization problems, that households and firms face. Pontryagin's minimum principle is used to find solutions of those optimization problems. (Kamien & Schwarz 1991) Since there is a relation between Bellman's equations and Pontryagin's minimum principle, there should be also a relation between Q-Learning (or VSCS) and Pontryagin's minimum principle. Mehta & Meyn (2009) developed algorithm, that is directly related to the Q-Learning algorithm. They were able to construct approximate solutions to the deterministic optimization problem in continuous time, creating a bridge between Hamiltonian appearing in nonlinear control theory and reinforcement learning. Therefore I wonder, whether it might be possible to obtain some sub-optimal or close to the optimal solution of simple macroeconomic models such as Ramsey-Cass-Koompas that rely on Hamiltonian, by means of reinforcement learning. One might ask, why should such thing be done? We can obtain optimal solution easily with Hamiltonian. Well, in the computer science community, researchers are often trying to find algorithm and convergence proofs, that would provide them with methods of solving some optimization problems by reinforcement learning. They often encounter great difficulties developing convergence theorems and this is one of the reasons reinforcement learning in non-cooperative games is sometimes considered not to be moving forward (Choi & Ahn 2010), (Mehta & Meyn 2009). In my opinion, in economics, one does not need optimal, or value maximizing solutions. Nor does he need convergence proofs. Sub-optimal results, that may fit the real world data better that the optimal ones are I think more useful. If one would have a clear theoretical relation between his algorithm and optimal control theory, it might be justifiable to use that algorithm even without the corresponding convergence proof. Because instead of that proof, one might try to find a relation between his algorithm and human subjects. Humans are not proven to reach optimal solutions of their "everyday optimization problems" either. An algorithm that could be considered a good approximations of Kahneman and Tversky's "heuristic rules"(Tversky & Kahneman 1974) may actually serve a some form of approximation of real life human decision maker. Then, it would be interesting to see, whether his results would fit the real world data better, than the optimal results that were found using Hamiltonian or Dynamic programming. Of course, the algorithm would produce not only sub-optimal solution of optimization problem, but also numerical one, instead of analytical. This may be hard to address, but I believe, that this topic would still deserve some attention.

Bibliography

BARTOSZYNSKI, R. & M. NIEWIDOMSKA-BUGAJ (2008): *Probability and Statistical Inference*. John Wiley and Sons Inc., 2 edition.

BELLMAN, R. (1957): *Dynamic Programming*. Princeton University Press.

BOOKER, L. B. & J. H. HOLLAND (1989): "Classifier systems and genetic algorithms." *Artificial Intelligence* **40(89)**: pp. 235–282.

BROWN, P. M. (1996): "Experimental evidence on money as a medium of exchange." *Journal of Economic Dynamics & Control* **20(4)**: pp. 583–600.

CHEN, S.-H. & Y.-L. HSICH (2011): "Reinformodels learning in ecperimental asset markets." *Easter Economic Journal* **37**: pp. 109–133.

CHOI, Y.-C. & H.-S. AHN (2010): "A survey on multi-agent reinforcement learning: Coordination problems." In "Embedded Systems and Applications," .

DORIGO, M. & H. BERSINI (1994): "A comparison of q-learning and classifier systems." .

DUFFY, J. (1994): "On learning and the nonuniqueness of equilibrium in an overlapping generations model with fiat money." *Journal of Economic Theory* **64**: pp. 541–553.

DUFFY, J. (2004): "Agent-based models and human subject experiments." .

DUFFY, J. & J. OCHS (1999): "Emergence of money as a medium of exchange: An experimental study." *American Economic Review* **89**: pp. 847–877.

HOLLAND, J. H. (1975): *Adaptation in Natural and Artificial Systems*. The University of Michigan Press.

HU, J. & M. P. WELLMAN (2003): "Nash q-learning for general-sum stochastic games." *Journal of Machine Learning Research* **4**: pp. 1039–1069.

KAMIEN, M. I. & N. L. SCHWARZ (1991): *Dynamic Optimization: The Calculus of Variations and Optimal Control in Economics and Management*. Elsevier Science, 2 edition.

KIYOTAKI, N. & R. WRIGHT (1989a): "A contribution to the pure theory of money." *Research Department Staff Report 13*, Federal Reserve Bank of Minneapolis.

KIYOTAKI, N. & R. WRIGHT (1989b): "On money as a medium of exchange." *Journal of Political Economy* **97(4)**.

KIYOTAKI, N. & R. WRIGHT (1993): "A search-theoretic approach to monetary economics." *The American Economic Review* **83**: pp. 63–78.

LUENBERGER, D. G. (1979): *Intruduction to Dynamic Systems*. John Wiley and Sons Inc.

MARIMON, R., E. MCGRATTAN, & T. J. SARGENT (1990): "Money as a medium of exchange in an economy with artificially inteligent agents." *Journal of Economic Dynamics and Control* **14**: pp. 329–373.

MATSUYAMA, K., N. KIYOTAKI, & A. MATSUI (1991): "Theory of international currency." *Discussion Paper no. 931* .

MEHTA, P. & S. MEYN (2009): "Q-learning and pontryagin's minimum principle." In "Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on," pp. 3598–3605.

MITCHELL, T. M. (1997): *Machine Learning*. McGraw-Hill, 1 edition.

POWELL, W. B. (2010): *Approximate Dynamic Programming*. John Wiley and Sons Inc.

RABIN, M. (1997): "Psychology and economics." *Department of Economics, UCB* .

SETHI, R. (1999): "Evolutionary stability and media of exchange." *Journal of Economic Behavior & Organization* **40**: pp. 233–254.

TESFATSION, L. & K. L. JUDD (editors) (2006): *Handbook of Computational Economics, Volume 2: Agent-Based Computational Economics*. North-Holland.

TVERSKY, A. & D. KAHNEMAN (1974): "Judgment under uncertainty: Heuristics and biases." *Science* **185**: pp. 1124–1131.

WATKINS, C. I. & P. DAYAN (1992): "Technical note - q-learning." *Machine Learning* **8**: pp. 279–292.

WILLIAMSON, S. & R. WRIGHT (2010): "New monetarist economics: Models." *Technical report*, Federal Reserve Bank of Minneapolis.

WILSON, S. W. (1995): "Classifier fitness based on acuracy." *Evolutionary Computation* **3(2)**: pp. 149–175.

WRIGHT, R. & A. TREJOS (2001): "International currency." *Advances in Macroeconomics* **1(1)**.

ZHOU, R. (1997): "Currency exchange in a random search model." *The Reviw of Economic Studies* **64(219)**: pp. 289–311.

Appendix A

Simulation results in Figures

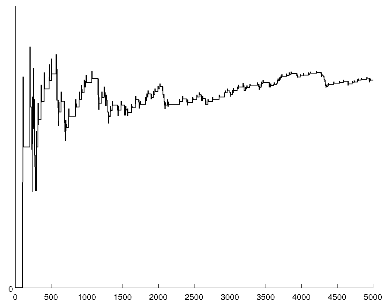

Figure A.1: Home currency for consumption good

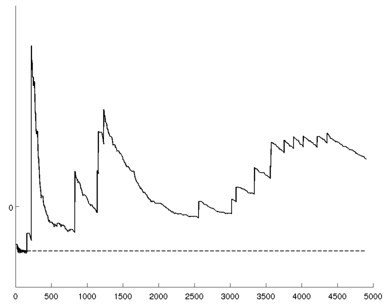

Figure A.2: Production good for home currency

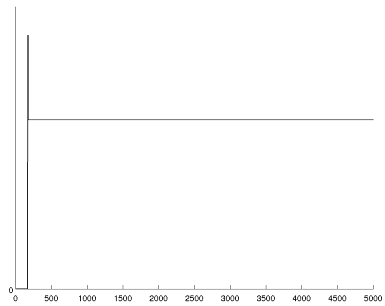

Figure A.3: Foreign currency for consumption good

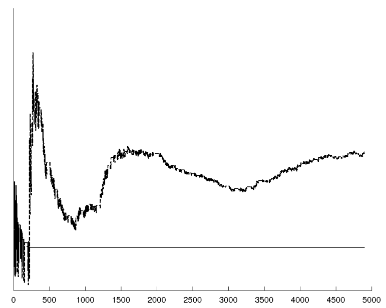

Figure A.4: Production good for Foreign currency

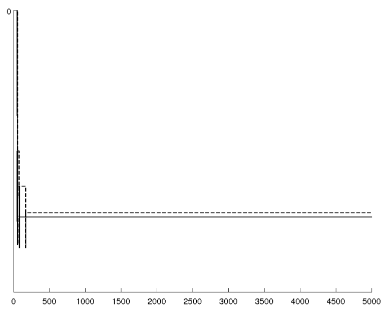

Figure A.5: Foreign Currency for Home Currency

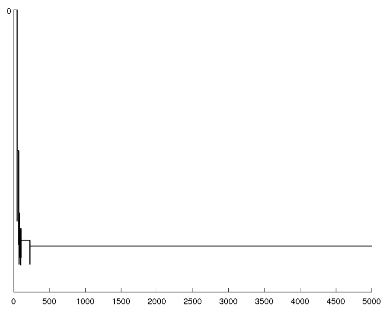

Figure A.6: Home Currency for Foreign Currency

Figure A.7: Two international currencies - Foreign Agent

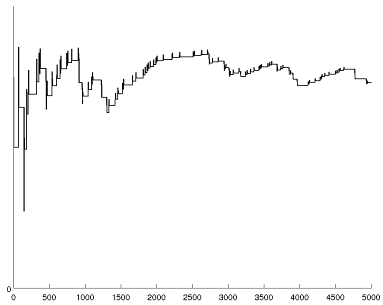

Figure A.8: Home currency for consumption good

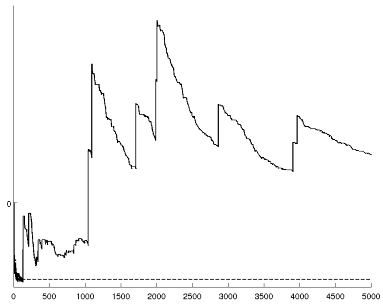

Figure A.9: Production good for home currency

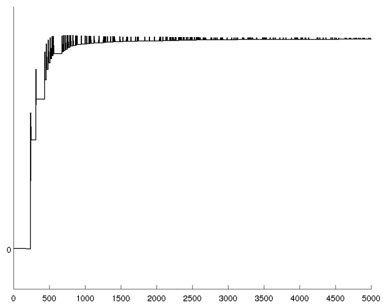

Figure A.10: Foreign currency for consumption good

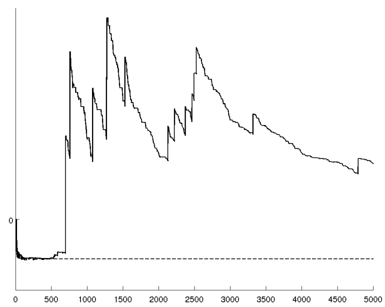

Figure A.11: Production good for Foreign currency

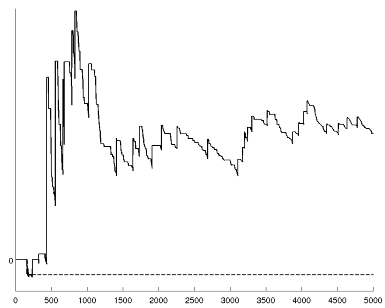

Figure A.12: Foreign Currency for Home Currency

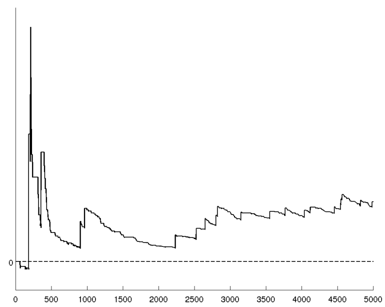

Figure A.13: Home Currency for Foreign Currency

Figure A.14: Two International currencies - Home Agent

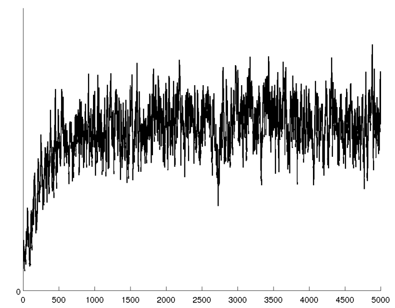

Figure A.15: Average fitness - Home economy

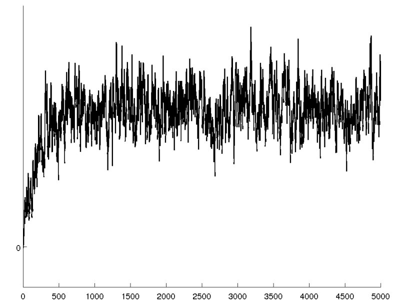

Figure A.16: Average fitness - Foreign economy

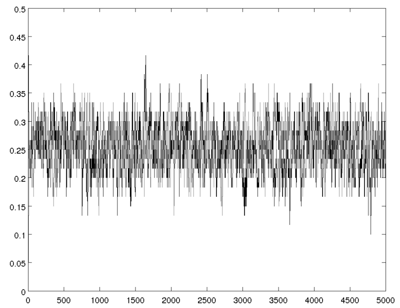

Figure A.17: Currency Holdings - Home economy

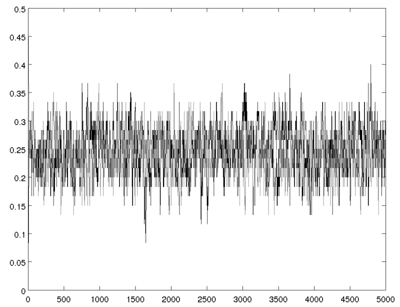

Figure A.18: Currency Holdings - Foreign economy

Figure A.19: Two international currencies - Overall

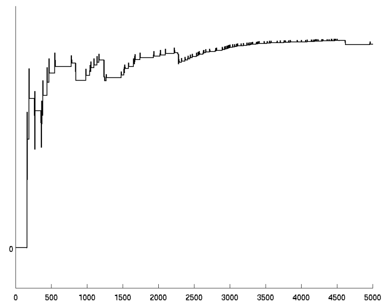

Figure A.20: Home currency for consumption good

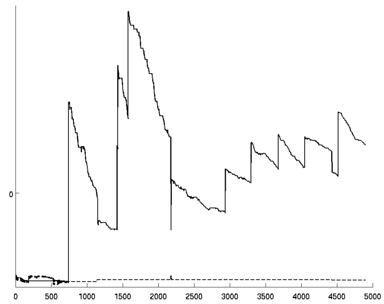

Figure A.21: Production good for home currency

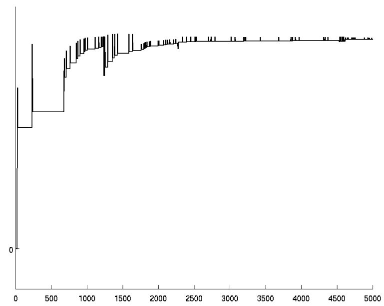

Figure A.22: Foreign currency for consumption good

Figure A.23: Production good for Foreign currency

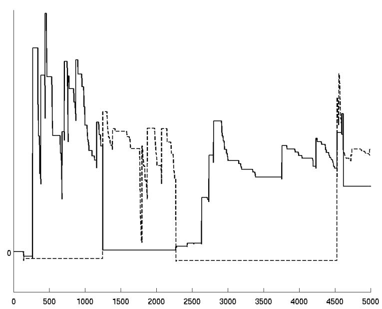

Figure A.24: Foreign Currency for Home Currency

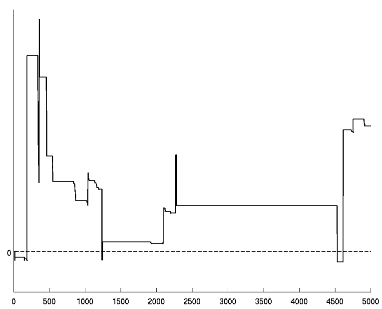

Figure A.25: Home Currency for Foreign Currency

Figure A.26: Two international currencies - Foreign Agent - Genetic

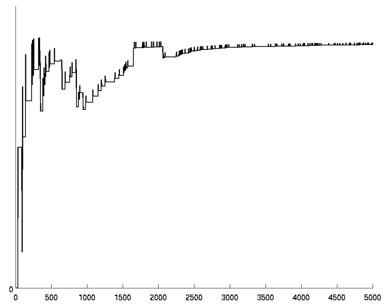

Figure A.27: Home currency for consumption good

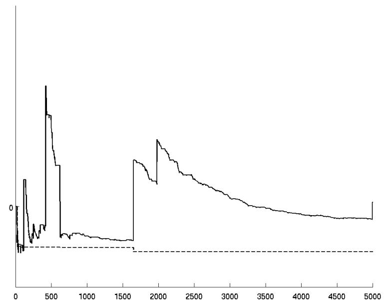

Figure A.28: Production good for home currency

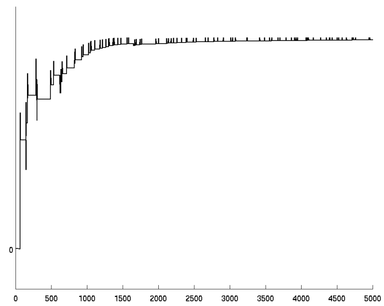

Figure A.29: Foreign currency for consumption good

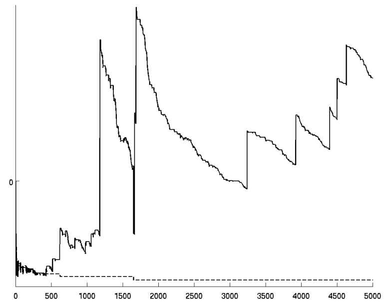

Figure A.30: Production good for Foreign currency

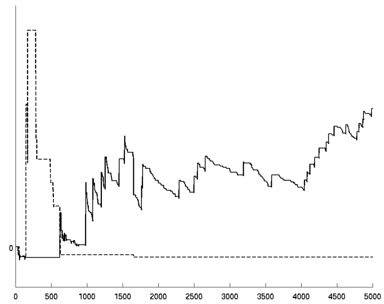

Figure A.31: Foreign Currency for Home Currency

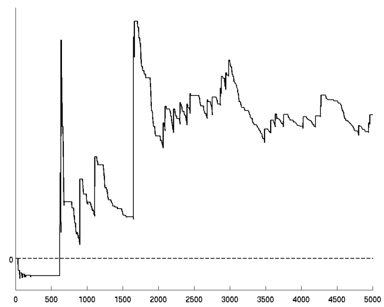

Figure A.32: Home Currency for Foreign Currency

Figure A.33: Two International currencies - Home Agent - Genetic

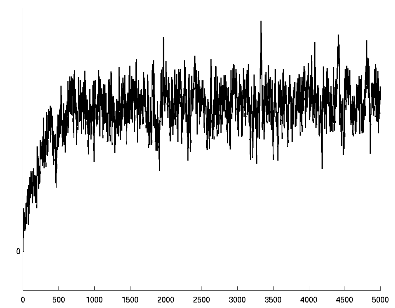

Figure A.34: Average fitness - Home economy

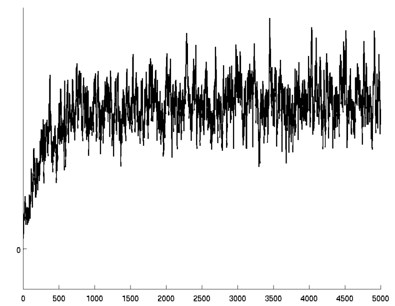

Figure A.35: Average fitness - Foreign economy

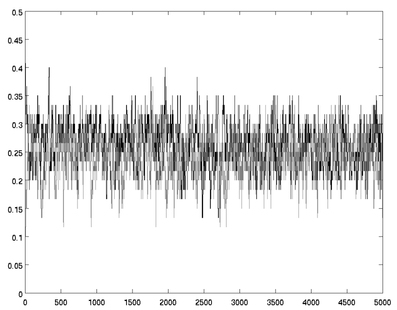

Figure A.36: Currency Holdings - Home economy

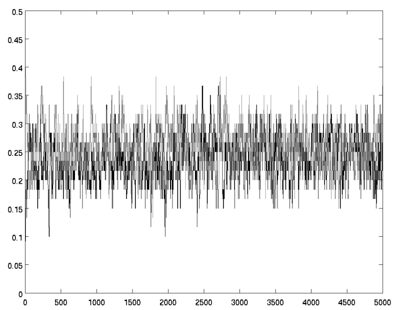

Figure A.37: Currency Holdings - Foreign economy

Figure A.38: Two international currencies - Overall - Genetic

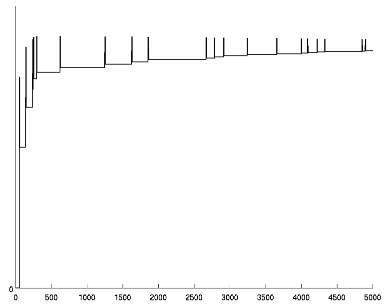

Figure A.39: Home currency for consumption good

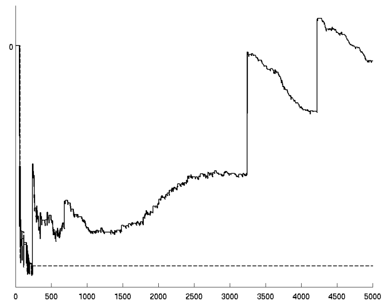

Figure A.40: Production good for home currency

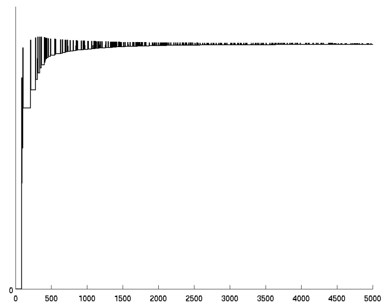

Figure A.41: Foreign currency for consumption good

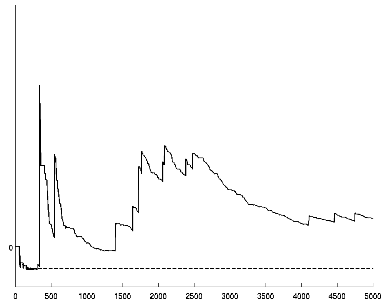

Figure A.42: Production good for Foreign currency

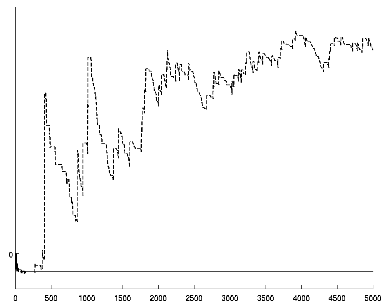

Figure A.43: Foreign Currency for Home Currency

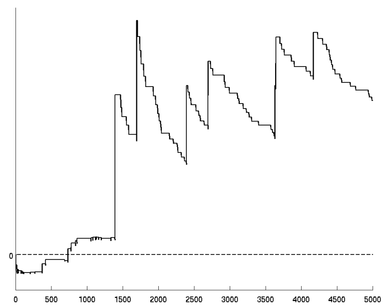

Figure A.44: Home Currency for Foreign Currency

Figure A.45: Two International currencies - Home Agent - Preset

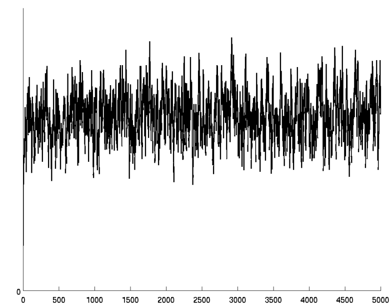

Figure A.46: Average fitness - Home economy

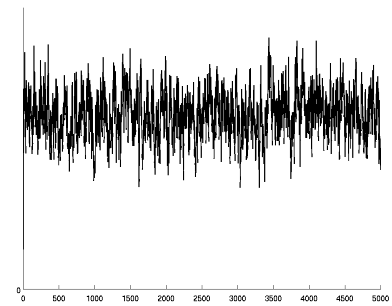

Figure A.47: Average fitness - Foreign economy

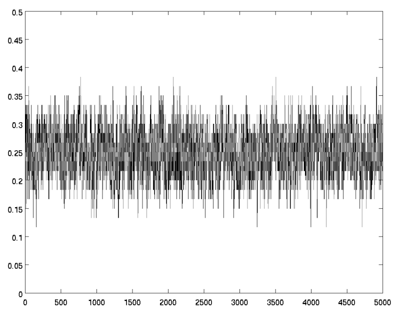

Figure A.48: Currency Holdings - Home economy

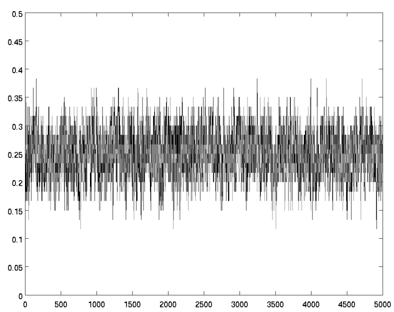

Figure A.49: Currency Holdings - Foreign economy

Figure A.50: Two international currencies - Overall - Preset

Appendix B

Content of Enclosed DVD

There is a DVD enclosed to this thesis which contains model source code in Matlab and simulation results.

- Folder 1: Model Source code for Matlab 2010b
- Folder 2: Simulation results