Universidade de Lisboa
Faculdade de Ciências
Departamento de Informática

EXPLORATION, DESIGN AND ANALYSIS OF SOCIAL SPACES FOR SOCIAL SIMULATION MODELS

Davide Delgado Nunes

DISSERTAÇÃO

MESTRADO EM ENGENHARIA INFORMÁTICA
Especialização em Interação e Conhecimento

2012

Universidade de Lisboa
Faculdade de Ciências
Departamento de Informática

EXPLORATION, DESIGN AND ANALYSIS OF SOCIAL SPACES FOR SOCIAL SIMULATION MODELS

Davide Delgado Nunes

DISSERTAÇÃO

Projecto orientado pelo Prof. Doutor Luís Alberto dos Santos Antunes

MESTRADO EM ENGENHARIA INFORMÁTICA
Especialização em Interação e Conhecimento

2012

Acknowledgments

It feels like it was just yesterday. I was a freshmen entering college, tasting the sweet and sour flavour of the road to an academic career. I would like to express my deepest gratitude to all the people that accompanied me throughout this journey and contributed to what I am and what I accomplished today.

First, I would like to thank my supervisor Prof. Luis Antunes for teaching me the stepping stones of science and guiding me to this very moment. He gave me the liberty to explore not only the theme in this thesis but also my own capabilities, becoming not only a great mentor but also a great friend. I would like to thank him for accompanying me this far. A special note for his gastronomic endeavours, which provided moments that can only be categorised with one word. Epic. I would like to thank Prof. Helder Coelho for the interesting discussions and his unique perspective over every subject.

I would like to thank Sara Rosa for her never-ending love and support (and delicious cakes) - you undoubtedly became the best part of my life, without which, I could not imagine to get this far.

I would also like to thank my saxophone teacher Mário Marques for showing me the beauty of the instrument I grown to love. It got me through this harsh academic world that I'm just starting to grasp. Music brings the best and the worst of my creative process to life and this is as important to science as it is to art. As *Plato* once said: "*Music gives soul to the universe, wings to the mind, flight to the imagination, and charm and gaiety to life and to everything*".

Last, but not least, I would like to thank all my friends and co-workers from the Laboratory of Agent Modelling (LabMAg). I would like to thank Phil Lopes with whom I could share, discuss and curse my hopes and dreams; Fernando Silva for our occasional coffee breaks that kept my sanity in place; Nuno Henriques, who, like me, still hopes for a change in the world and believes we can and will make a difference; André Bastos for being awesome; Christian Marques for being a great friend and sharing one of the things that is dear to me, music, and João Costa for showing me that everything is possible, especially if you have never done it before.

Resumo

Em simulação social, a estrutura das relações sociais é não só fundamental para a concepção de cenários de simulação plausíveis, mas também para a compreensão de processos de interação guiados por tais estruturas. Cada ator interage em múltiplos contextos integrados em múltiplas relações que constituem o seu espaço social. Nesta tese, tomando como base o trabalho prévio acerca de modelos de trocas de contextos sociais, é estudada a noção de segregação social e seu impacto na disseminação dos fenómenos através da sociedade. Os agentes não só trocam entre contextos sociais, levando com eles a sua identidade social única, mas também escolhem os contextos de acordo com razões pessoais. A noção de segregação entre contextos é aplicada a um jogo de consensos simples. Neste jogo, os agentes tentam coletivamente alcançar um consenso sobre uma opinião ou escolha binária que devem fazer. Este trabalho compreende o desenho e análise de um conjunto de experiências destinadas à observação da influência de mecanismos de segregação na velocidade de convergência para um consenso global numa sociedade de agentes. É criado um modelo onde existe uma abstração de escolha estratégica de vizinhanças (os contextos sociais). Neste modelo, um conjunto de valores de tolerância associados a cada contexto social define um limiar para que uma vizinhança seja considerada adequada ou não. Um agente decide trocar de contexto se o número de vizinhos a adoptar uma escolha contrária à sua esteja acima da tolerância definida para o contexto corrente. Os resultados do modelo concebido são posteriormente comparados com os resultados do modelo de troca de contextos previamente desenvolvido.

Neste trabalho, é confirmada a hipótese feita sobre um modelo de troca de contextos desenvolvido anteriormente. A primeira conjetura confirmada é que a formação de consensos locais ajuda à aceleração da convergência da sociedade para um consenso global, especialmente se esses grupos de consenso local tiverem as condições estruturais adequadas para serem criados. Uma ideia interessante é que nas condições certas, as dinâmicas de formação de grupos de consenso podem ser localizadas e relacionadas com determinadas topologias construídas à custa de várias redes sociais. Este facto é especialmente interessante para o desenvolvimento de estudos de disseminação de informação em estruturas sociais bem conhecidas como campanhas políticas ou de marketing em redes sociais on-line.

As experiências concretizadas sobre o mecanismo de segregação apresentado mostraram resultados interessantes. Foi observado que, sob as condições adequadas, a segregação acelera a convergência para um consenso global. É curioso observar que este mecanismo não só acelera a velocidade de auto-organização da sociedade de agentes, mas também é transversal às estruturas sociais utilizadas. Por outras palavras, o efeito de otimização de convergência para um consenso é observado para diferentes redes sociais. Ainda mais intrigante é esta otimização se manter mesmo quando é aumentado o número de relações sociais concomitantes.

Foi ainda verificado que o mecanismo introduzido não altera drasticamente o comportamento do modelo de troca de contextos anteriormente desenvolvido. Globalmente, as tendências de frequência de troca entre contextos são preservadas mantendo-se fortemente relacionadas com a probabilidade de trocar de contexto social (parâmetro integrante do modelo de troca de contextos que define, de forma abstrata, o tempo que cada agente passa em cada um dos contextos sociais).

Adicionalmente ao modelo de simulação social desenvolvido, foi feita uma recolha extensa do estado da arte referente a simulação, simulação baseada em agentes e dos mais recentes avanços em simulação social relacionados com o trabalho desenvolvido. Para além de métodos e modelos de simulação, foi feita uma recolha de modelos generativos de redes sociais complexas bem como uma detalhada apresentação de alguns conceitos base. É feita uma revisão sobre conceitos base de teoria de grafos e análise de redes sociais. Neste último tópico são incluídas algumas das medidas que podem ser feitas sobre estruturas como redes sociais. Estas são fundamentais para compreensão dos modelos generativos apresentados.

A concepção de modelos de simulação social requer que um conjunto de componentes sejam modeladas antes que uma experência possa ser concebida. Estes componentes não estão normalmente desacoplados dos processos experimentais, o que cria um problema relativamente à reprodução e reusabilidade de modelos. Para contribuir com uma solução para este problema, esta tese fornece alguns avanços no desenvolvimento da b-have workbench. Este é um projecto dedicado à criação de componentes reutilizáveis para modelos de simulação social. Tais componentes compreendem: redes sociais complexas, modelos de agentes, regras comportamentais e modelos abstractos de ambientes. Estes componentes introduzem uma separação entre processos de modelação e simulação. O foco do trabalho realizado neste contexto, é na criação de modelos de redes sociais complexas posteriormente integradas no modelo desenvolvido de segregação entre contextos. Para cumprir este objectivo, foi criada uma API (Application Programming Interface) para permitir a criação de instâncias de redes sociais em Java, API essa que está integrada no âmbito do desenvolvimento da plataforma b-have workbench.

Finalmente, este trabalho aborda outro problema da simulação social, a criação

de uma infra-estrutura adequada para o desenvolvimento e execução de simulações de uma forma escalável. A exploração de modelos de simulação social baseados em agentes leva muitas vezes demasiado tempo para que seja obtida uma quantidade de resultados suficiente para uma análise significativa dos dados gerados. Nesta tese é feita uma análise do problema e fornecida uma solução que o minimiza utilizando computação em grelha. Este trabalho fornece uma discussão detalhada acerca de processos de construção de modelos de simulação social bem como de desenho e distribuição de experiências utilizando uma grelha de computadores. Esta infra-estrutura é posteriormente utilizada para a execução experiências sobre o modelo de simulação social desenvolvido.

Palavras-chave: simulação social, simulação baseada em agentes, redes sociais, contextos sociais

Abstract

In social simulation, the structure of the social relations is not only fundamental for the construction of plausible scenarios, but also important to construct an understanding of interaction processes shaped by such structures. Each actor interacts in multiple social contexts located within multiple social relations that constitute their social space. In this thesis, we build on previous work about context switching to study the notion of context segregation. The agents not only switch between social contexts, carrying with them their unique social identity, but also choose the contexts according to personal reasons. We apply the notion of context segregation to a simple game of consensus in which agents try to collectively achieve an essentially arbitrary consensus. This work comprehends the design and analysis of a set of experiments towards the understanding of the influence of the segregation mechanism in the speed of convergence to global consensus, comparing the results with the previous model of context switching.

Social simulation requires a series of components to be modelled prior to the experiment set-up. These components are usually not decoupled from the experiment process. This creates a problem of experiment reproduction and model reusability. To contribute with a solution to this problem, this thesis also provides advances to the development of the b-have workbench. This is a project dedicated to the creation of reusable social simulation components such as complex social networks, agent models, behaviour rules, and environment abstract models. We focus on the creation of complex social network models and integrate them in our model of context segregation.

An extensive review over the state-of-the-art on simulation methods, agent-based simulation and social simulation models is also presented in this thesis. Moreover we have also reviewed a series of generative models for complex network structures and the basic associated with it. We describe the fundamental background on graph theory and social network analysis. Within social network analysis, we present a series of measurements essential for the understanding of the network models here presented.

Finally, this work tackles another problem in social simulation, the creation of a proper infrastructure for scalable simulation deployment. The exploration of agent-based social simulation models often takes too much time to get enough results for a significant analysis of the data generated. In this thesis, we show how to

minimise this problem by using grid computing. We provide insights on social simulation model construction, experiment design and experiment distribution using a computer grid. The developed infrastructure is then used to deploy the social simulation model created.

Keywords: social simulation, agent-based simulation, social networks, social-contexts

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Simulation introduces the possibility of a new way of thinking about social and economic processes, based on ideas about the emergence of complex behaviour from relatively simple activities (Simon, 1996). Social simulation arises as a recent area or research that explores simulation models as a way to obtain a better understanding of some features of the social world (Gilbert and Troitzsch, 2005).

There is a certain scepticism about the possibility of making social predictions, based on both the inherent difficulty of doing so and the possibility, peculiar to social and economic forecasting, that the forecast itself will affect the outcome (Gilbert and Troitzsch, 2005). Nonetheless, social simulation models are useful, first and foremost, to construct conceptual explanations for target modelled phenomena. Models that closely represent the target phenomena can then be used to create *"what if scenarios"*. These are particularly useful, not only to formulate questions about a problem, but also to answer such questions with a predicted outcome in mind. This process is particularly useful to areas like policy-making (Murata et al., 2007).

To make plausible explanations about social systems we must create social simulation models that can be validated by real-world phenomena. To accomplish that, we rely on complex models that mimic real-world structures such as complex social networks.

This dissertation aims to contribute to the state-of-the-art in social simulation by creating a model that integrates different social networks as the core component to model social relations. We consider a simple game of consensus to explore the properties of our model. In this game, the agents try to achieve an arbitrary global consensus using a very simple set of interaction rules. The proposed mechanism is simple enough not only to observe the influence of different social networks in the outcome of the auto-organisation process, but also to explore different social space designs with multiple concurrent social contexts (Antunes et al., 2009; Nunes and Antunes, 2012a).

We build on previous work on context switching mechanisms (Antunes et al.,

2009) to explore a hypothesis presented in (Antunes et al., 2009). Our conjecture states that a society of agents converges to consensus more rapidly due to the local consensus group formation. We explore this hypothesis by introducing a segregation behaviour in the context switching model. In this model, the agents avoid undesired neighbourhoods within their multiple social context space. They segregate by avoiding neighbourhoods in which a choice, contrary to their own, is above a given tolerance value (associated with each social context).

We chose this simple game of consensus because the focus of this work is on the segregation mechanism itself and not in the game. Our focus is on the dynamics that segregation introduces when using different network structures to represent the social space of the simulation model. A more complex game would make the exploration of segregation and switching dynamics much more difficult.

1.1 Motivation

It is important to understand the influence of different complex social networks in the construction of social simulation models. Different network models allow us to construct simulation scenarios with distinct social structural properties. The set of properties of interest depends on the social process one is trying to model. As the research work regarding this issue is very limited, we aim to contribute with the exploration of a social simulation model in which different social structures are used and compared.

In real social world scenarios, agents interact in multiple complex social relations with other agents and/or institutions. Each one of these relations may be of different kind and quality, possessing different topologies and social dynamics. Multi-context models (Antunes et al., 2009; Nunes and Antunes, 2012a), introduce a framework to test social context dynamics. We are particularly interested in making observations on how different social relation topologies influence emergent auto-organisation processes like the achievement of arbitrary consensus in a society of agents.

Another problem in social simulation is that it suffers often from the lack of standardisation in what concerns to the conception of simulation models. It is truly a multi-disciplinary area of research but shamefully the creation of new models tends to be time consuming. This is due to the lack of clear methodologies and tools to help in the modelling and simulation processes. We contribute to this problem by discussing a clear methodology and supplying a set of tools (Nunes and Antunes, 2012b) in the process of conception of our model of context segregation (Nunes and Antunes, 2012a).

To help researchers to focus on the modelling processes without being absorbed by technical difficulties of implementation, we propose the definition of a social

simulation methodology focused on the exploration of the design space of artificial society structures. To support this methodology we use a workbench (Nunes and Antunes, 2011) with modules and libraries that support the formal design of the models to be created. By focusing the model design on complex social networks that shape the social space, we create adequate models for the study of social processes. The workbench is intended to support the models by supplying a wide variety of out-of-the-shelf complex social network models. We aim to collect a state-of-the-art set of social network models, exposing a set of modelling options to construct social relation structures and supplying candidates to complement the current range of algorithms available in the b-have workbench (Nunes and Antunes, 2011).

1.2 Objectives

This dissertation comprises three main objectives. The first and main objective is to construct a social simulation model building on previous work regarding multi-context models (Antunes et al., 2008, 2009). The main goal of the model to be developed is to explore the dynamics introduced by segregation between multiple social contexts. With this, we also want to model an abstract strategic neighbourhood selection, similar to what happens in the social segregation model introduced by *Schelling* (Schelling, 1969), considering already existing complex social network structures as our social environment. By creating this model we want to catalyse the conditions that allow us to confirm a hypothesis drawn from our previous work on context switching (Antunes et al., 2009). The hypothesis states that local groups of consensus within multiple social contexts, contribute to a faster convergence to global consensus formation. Moreover, we also want to confirm not only that these groups exist, but also that the way they are formed is strongly influenced by different social network topologies used to model abstract social relations.

The second objective is to present an extensive review on simulation methods, agent-based approaches to simulation and social simulation models. We want to give particular attention to social simulation models that make use of network structures. We also pretend to review the state-of-the-art generative network models. In order to create a better understanding of these models, we aim to create an introduction to some fundamental notions of graph theory and social network analysis. Regarding the later, the focus should be on the measurements that can be made over network structures and that promote a clear insight over their properties.

Our final goal is to describe in a formal and informal manner, the way one can assemble the multiple-context models and distribute experiments over a computer grid. We want to create an implementation that can then be used in the experiments over the model constructed in this dissertation. This work aims to reduce the time

needed to perform exploratory simulations over huge parameter spaces.

1.3 Contributions

In this section, we outline the concrete contributions made in this thesis. These contributions were presented to the scientific community in several papers subject to a peer reviewing process and presented in top-level workshops and conferences in the area.

- This thesis provides an extensive review over simulation methods, agent-based modelling techniques and social simulation models. We also provide an extensive review over complex network generative models and the theory behind such structures. To support our discussion, we also present introductory notions on graph theory and social network analysis. In the latter, we present a set of measurements fundamental for the understanding of our social network structures. These notions serve as the building blocks used to conceive and comprehend complex social network models.

- We then contribute to the previous line of work on the *b-have workbench* (Nunes and Antunes, 2011):

  Davide Nunes, Luís Antunes – "***Introducing the b-have workbench – creating reusable components for social simulation experiments***", 7th European Social Simulation Association Conference, ESSA 2011

  The advance on this project was the development of an *API* suitable for social network model integration in Java-based social simulation environments.

- We present a social simulation environment that allows for the usage of network generation algorithms, a multi-agent system simulation model and the distribution of experiments in a grid environment. This environment is described both formally and informally in what regards to its implementation. We extended the state-of-the-art by providing an implementation that can serve as a working example to create similar models.

  The work in this context was selected for an oral presentation and published as a full paper (Nunes and Antunes, 2012b):

  Davide Nunes and Luis Antunes - "***Parallel Execution Of Social Simulation Models In A Grid Environment***", 13th International Workshop

on Multi-Agent Based Simulation, MABS 2012

This contribution also serves as a starting point for a future tutorial on various social simulation platform and grid computation infrastructures.

- We also extend the state-of-the-art by providing a model of context segregation based on previous work regarding context switching and a set of experiments that explore the segregation dynamics and its role to the achievement of a global consensus. The basis for our current work can be found on the following publication (Antunes et al., 2009):

  Luís Antunes, Davide Nunes, Helder Coelho, João Balsa, Paulo Urbano – "***Context Switching versus Context Permeability in Multiple Social Networks***", 14th Portuguese Conference on Artificial Intelligence, EPIA 2009

  The work on the model presented in this thesis was selected for an oral presentation and published as a full paper (Nunes and Antunes, 2012a):

  Davide Nunes and Luis Antunes - "***Consensus by segregation - the formation of local consensus within context switching dynamics***", 4th World Congress on Social Simulation, WCSS 2012, 2012

1.4 Document Structure

This document is organised as follows.

Chapter 1 (Introduction) presents the overview of the work here presented and summarises the main context of research done in this thesis. We describe our mains goals and motivations for the construction of a social simulation model of context segregation based on a complex network structure. We also highlight the contributions of this thesis for the state-the-art in social simulation.

Chapter 2 (Related Work) presents the related work. This section is divided in three main sections. In the first section we present a comprehensive overview over graph theory, social network analysis and complex network models. We then review simulation techniques and some core concepts present in social simulation, contextualising the work here presented. Finally we describe the previous work regarding the definition of social simulation models (Antunes et al., 2008, 2009) that make use of multiplex social networks to represent multi-dimensional social spaces. The social simulation model to be developed builds on top of this work and

aims not only to explore the dynamics of different complex network structures, but also to serve as a proof-of-concept for the formal model definitions we provide.

Chapter 3 (Model Construction and Deployment) presents the core formal frameworks proposed to define social simulation models and experiments as well as informal methodologies and model development guidelines. It presents both the tools used to carry out the prototyping and implementation of the social simulation model designed. One of the tools described is the b-have workbench project (Nunes and Antunes, 2011), depicting the state of this project and the contribution for its continuous development. Finally, we describe how one can improve the model deployment performance by showing how to use grid computing to do a parallel exploration of the model parameter space.

Chapter 4 (Exploring Context Switching With Segregation), explores a model constructed building on (Antunes et al., 2008, 2009). In this chapter, we compare different social structure model designs and explore the properties of multiplex social networks in the design of artificial social spaces. We present the devised model of context segregation and discuss the results of multiple experiments carried out with this model.

Finally, in **chapter 5 (Conclusions)** we wrap up, analysing the contribution of the developed work regarding its importance to social simulation standardisation, modelling guidelines and overall advances in the state-of-the-art.

Chapter 2

Related Work

In this chapter we discuss some relevant research areas fundamental for the understanding of social simulation and modelling and for the usage of complex social networks in the conception of models. The sections in this chapter are focused around the following issues: complex social networks, computational simulation methods, agent-based approaches to simulation and the recent advances in social simulation models that make use of social network structures. We also discuss a model that makes use of multiplex social networks to represent social spaces.

In the complex social network section, we lay out some theoretical foundations on graph theory, social network analysis (discussing some fundamental measurements that can be made over networks)and a set of network models that can be used in social simulation experiments. In the simulation section we present the core concepts regarding simulation methods and review relevant tools and approaches to modelling. At the end of this section we present and contextualise some social simulation models that focus on relevant problems and make use of social network structures. Finally, we present the model from which the work presented in this thesis builds on a social simulation multi-agent-based model that uses multiple state of the art complex social networks to construct a more realistic representation of the structures present in real social systems.

2.1 Complex Social Networks

The definition of complex network is closely related to the definition of graph in mathematical literature. A network is a set of nodes, with connections between them, called edges or links. Perhaps the main difference between a network and a graph resides in the set of non-trivial properties associated with the former. A network is essentially an entity represented using graphs but with extra elements such as time, non-trivial relationships between nodes and links and some behaviour of nodes and links versus time (Lewis, 2009). Network theory is then built on top

of graph theory (Diestel, 2006). The study of networks in the form of mathematical graph theory is, in fact, one of the pillars of discrete mathematics. Euler's celebrated 1735 solution of the Königsberg bridge problem is often cited as the first proper proof in the theory of networks (Newman, 2003b).

There are plenty of systems taking the form of networks in the world. Examples include the Internet, the World Wide Web, social networks or other connections between individuals, organisational networks and networks of business relations between companies, networks of citations between papers, and many others. These network systems are intrinsically complex. Complex network research lie at the intersection between graph theory and statistical mechanics (Costa et al., 2007), it is truly a multi-disciplinary area with contributes from various domains from biology (de Silva and Stumpf, 2005; Hintze and Adami, 2008) to physics (Deng et al., 2011). This area provides tools to construct and analyse models that closely relate to real world structures.

As our main focus is on complex social networks, one must define what makes them *complex*. Complexity comes from the non-trivial properties associated with their structure that often occur in real-world network structures. One example of these properties is the set of topological features that comprehend the existence of connection patterns between nodes that are neither purely regular nor purely random. Such features include a heavy tail in the degree distribution, a high clustering coefficient, community structures, and hierarchical structures. These features will be discussed in the following sections when we describe the state-of-the-art complex network models. One of the sub-areas of this field of research is Social Network Analysis (SNA) (Wasserman and Faust, 1994). This area supplies various methods for the extraction of the described features. The main measurements and high level property detection methods intrinsic of SNA will also be described in the following sections.

The study of complex social networks is fundamental for computational sociology and social simulation. A powerful idea drawn from social sciences is the notion that individuals are embedded in thick webs of social relations and interactions. As such, theory of networks yields explanations for social phenomena in a wide variety of disciplines from psychology to economics. Another fundamental issue is that social network theory provides an answer to a question that has preoccupied social philosophy since the time of Plato, namely, the problem of social order: how autonomous individuals can combine to create enduring, functioning societies (Borgatti et al., 2009).

2.1.1 Graph Theory and Social Network Analysis

In this section, we present the relevant core concepts regarding graph theory and social network analysis. These concepts are fundamental to the understanding of the complex network model properties to be presented in the next section. We thus present some basic formal definitions regarding graphs which are a good way of representing network structures and some social network analysis methods, in particular, relevant measurements that can be made from networks in order to characterise them. We finish by presenting high level property extraction methods, such as community detection.

Graph Theory Basic Concepts

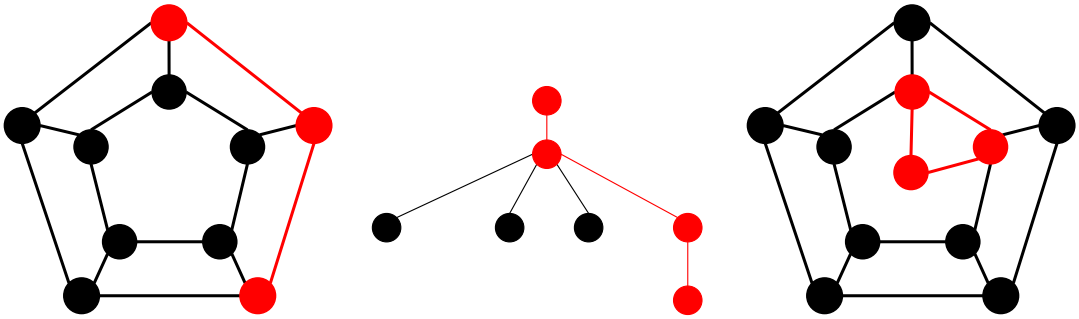
Networks can be represented using graphs. The reader can find a complete overview over graph theory in (Diestel, 2006). A graph G can be formally represented as an ordered pair $G = (V, E)$ where V is the set of all the vertices of the graph and E is the set of all the edges of the graph. An edge e can then be defined as $e \in E$, where $E \subseteq \{\{v_i, v_j\} : v_i, v_j \in V\}$. An edge is characterised by the vertices it connects and the type of connection (*undirected* or *directed*), making the graph respectively *undirected* or *directed* (or *digraph*). As such, two vertices v_i, v_j are neighbours if there is an edge $e = (v_i, v_j)$ such that $v_i, v_j \in V$ and e is an edge of G. As graphs can be *directed* or *undirected*, we should clearly outline the main difference between those two types. In a directed graph the order of a pair of nodes that defines an edge is important, thus, an edge represented by $e = (v_i, v_j)$ is not the same as an edge defined by $e = (v_j, v_i)$. In undirected graphs the order of the vertices is irrelevant making the two representations the same. We consider complex networks to be represented as *undirected* graphs by default as we are only interested in the underlying structural properties.

The number of neighbours of a vertex v is its *degree* (also called *connectivity*) and can be represented as $d_G(v) = d(v) = k_v$.

Another important concept is the concept of a *path*. A *path* in a graph $G = (V, E)$ is as a non-empty graph $P = (V_P, E_P)$ with $V_P \subseteq V$ and $E_P \subseteq E$. A path in a graph represents a way to transverse from an origin vertex to a destination vertex by traversing edges in the graph without repetition of vertices. Formally we can define a path as an ordered list of edges $P = ((v_1, v_2), (v_2, v_3), \cdots, (v_k, v_{k+1}))$ where each $v_i \in V_P$, v_1 is the origin vertex and v_{k+1} is the destination vertex. The *path length* is the number of edges of P (this is $|E_P|$). The shortest path (see figure 2.1(a)) or graph geodesic between two distinct vertices v_i, v_j can defined as being a path with less edges than all the other paths in the graph between those two vertices. The distance between two nodes can also be defined as the minimum length of the paths connecting them (the length of a shortest path). The greatest distance between

any two distinct vertices in a graph G (longest shortest path) is the *diameter* of G, in figure 2.1(b) we have an example of a graph with diameter 3. We say a graph $G = (V, E)$ is *connected* if a path exists between any two vertices v_i, v_j belonging to G.

Finally we have to concept of *clique*. A *clique* in an *undirected graph* is a subset of its vertices such that every two vertices in the subset are connected by an edge (see figure 2.1(c)).



(a) Shortest path (b) Graph with diameter = 3 (c) Clique

Figure 2.1: Examples of graph theory concepts

Complex Network Definition

We build on graph theory to define formal complex network models. A network model M defines a way to construct complex networks using functions that give rise to specific topologies. We call our vertices *nodes* and our edges *links* to make a distinction between the levels of abstraction of graphs and networks. Such models can be can be formally defined (see (Lewis, 2009)) as being an entity of the form :

$$M(t) = (N(t), L(t), F(t), J(t))$$

where:

t =time, simulated or real
$M(t)$ =network model instance on moment t
$N(t)$ =set of all *nodes* on moment t
$L(t)$ = set of all *links* on moment t
$F(t)$ = mapping function that connects node pairs, creating a network topology on moment t
$J(t)$ = function describing the behaviour of *nodes* and *links* over time.

The component $J(t)$ is left abstract on purpose as it can represent properties associated with the nodes and links that can vary over time. Such properties vary from model to model. We use the time component t to describe models of network evolution. With this, one can represent dynamic network models approximated to real network systems. The models are used either to generate network structures with or without relying on empirical data or to help on the understanding of real-world network formation. We can extract information about the structures conceived by these models by relying on *Social Network Analysis (SNA)* measurements. A quick overview over those measurements is given in the following section.

Network Measurements

We rely on social network analysis methods to analyse and characterise the different topological properties of complex networks, being those generated by models or based on real data. This section presents a brief overview over measurements that can be made over networks. These measurements serve as support to understand the state-of-the-art complex network models that will later be discussed. For more detail on this topic, the reader is referred to a survey regarding measurements by *Costa et al.* (Costa et al., 2007).

Definition 1 (Degree of a node)**.** The *degree* of a node n_i also called *connectivity* of a node and denoted as k_i can be defined as the number of the edges connected to it:

$$k_i = \sum_j (n_i, n_j) \tag{2.1}$$

where (n_i, n_j) is an edge between the node n_i and the node n_j. If the network is directed we have two kinds of degrees, the *in degree* and the *out degree*. We define the *in degree* of a node n_i as the number of edges directed to the node $(\sum_j (n_j, n_i))$ and the *out degree*, as the number of edges directed from the node $(\sum_j (n_i, n_j))$.

Definition 2 (Average degree of a network)**.** The *average degree* of a network $\langle k \rangle$ is the average of k_i for all the nodes in the network, that is:

$$\langle k \rangle = \frac{1}{N} \sum_i k_i \tag{2.2}$$

Definition 3 (Betweenness centrality)**.** In networks, the greater the number of paths in which a node or link participates, the higher its importance for the network. It is possible to quantify the importance of a node or link u in terms of its *betweenness centrality* as following:

$$B_u = \sum_{ij} \frac{\sigma(n_i, u, n_j)}{\sigma(n_i, n_j)} \tag{2.3}$$

where $\sigma(n_i, u, n_j)$ is the number of shortest paths between the nodes n_i and n_j that contain the node or link u and $\sigma(n_i, n_j)$ is the number of shortest paths between n_i and n_j. The sum is conducted between all the pairs of distinct nodes (n_i, n_j).

Definition 4 (Clustering Coefficient)**.** The clustering coefficient is a measure of the degree to which nodes in a network tend to cluster together. In most real-world networks, and in particular social networks, nodes tend to create tightly knit groups characterised by a relatively high density of links. In real-world networks, this likelihood tends to be greater than the average probability of a tie randomly established between two nodes (Watts and Strogatz, 1998; Costa et al., 2007).

This measure quantifies how close the neighbours of a node are to forming a *clique. Duncan J. Watts and Steven Strogatz* (Watts and Strogatz, 1998) introduced the measure in 1998 to determine whether a network is a small-world network. The local clustering coefficient for a node n_i, denoted by C_i, is then given by the proportion of links between the nodes within its neighbourhood divided by the number of links that could possibly exist between them. For a network represented by a directed graph $G = (V, E)$, an edge $e_{ij} = (n_i, n_i)$ is distinct from $e_{ji} = (n_j, n_i)$, and therefore for each neighbourhood N_i there are $k_i(k_i - 1)$ links that could exist among the nodes within the neighbourhood (remember that k_i is the total (in + out) degree of the node). Thus, the local clustering coefficient for directed graph is given as:

$$C_i = \frac{|\{e_{jk}\}|}{k_i(k_i - 1)} : v_j, v_k \in N_i, e_{jk} \in E \tag{2.4}$$

For a network represented by an undirected graph we just have to take into account that an edge e_{ij} is the same as e_{ji}, as such, the expression to calculate the local cluster coefficient of the nodes is:

$$C_i = \frac{2|\{e_{jk}\}|}{k_i(k_i - 1)} : v_j, v_k \in N_i, e_{jk} \in E \tag{2.5}$$

The cluster coefficient for the whole network represented by the graph $G = (V, E)$ is defined as the average local cluster coefficients for all the nodes. This is:

$$\langle C \rangle = \frac{1}{n} \sum_{i=1}^{n} C_i \tag{2.6}$$

Other Network high level traits

Real-world complex networks of actors, more commonly known as social networks, can exhibit other high level interesting properties. When analysing complex networks, researchers tend to focus only on a few properties that seem to be common to many networks: *power-law degree distributions*, *network transitivity* (short average path length), etc. There are other high level measurements that can be interesting for discovery and analysis of interesting patterns within social networks one of such examples is the existence of *community structures*.

Girvan & Newman (Girvan and Newman, 2002) highlight a property that is found in many networks, the property of community structure (see figure 2.2), in which network nodes are joined together in tightly knit groups, between which there are only looser connections. They proposed a method for detecting such communities (Girvan and Newman, 2002; Steinhaeuser and Chawla), built around the idea of using centrality to find community boundaries. The method is meant to be an alternative to *hierarchical clustering*[1]. Instead of trying to construct a measure that tells which edges are most central to communities, they focus instead on edges that are least central, the edges that are most "between" communities. If a network contains communities or groups that are only loosely connected by a few intergroup edges, then all shortest paths between different communities must go along one of these few edges. Thus, the edges connecting communities will have high *edge betweenness*. By removing these edges, the method separates groups from one another and reveals the underlying community structure of the network. The algorithm they propose for identifying communities is stated as follows:

1. Calculate the betweenness centrality (see definition 3) for all links in the network.
2. Remove the link with the highest betweenness.
3. Recalculate betweenness for all links affected by the removal.
4. Repeat from step 2 until there are no links to be removed according to a user-defined threshold.

The result of the algorithm is a graph where each community is isolated from each other. The removed edges are the edges that connect these communities to each other.

[1]Hierarchical clustering is a method of cluster analysis which seeks to build a hierarchy of clusters. The strategies can generally be categorised as *agglomerative* (a bottom-up approach) or *divisive* which is a top-down approach (Johnson, 1967).

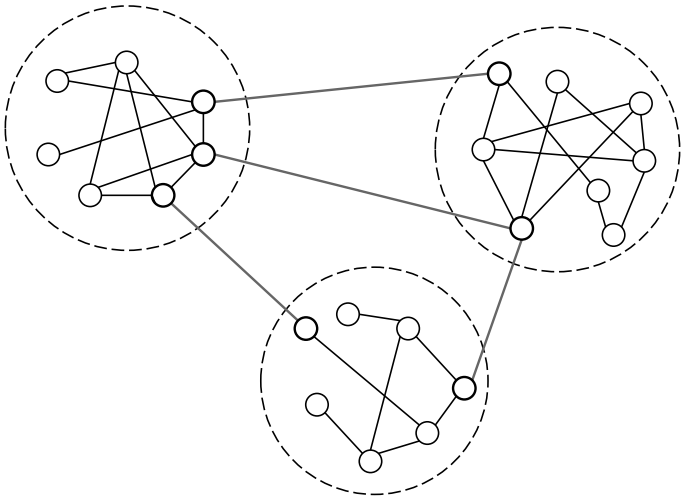


Figure 2.2: A schematic representation of a network with community structures. In this network there are three communities of densely connected nodes.

2.1.2 Complex Network Models

Each complex network or class of complex networks, presents specific topological features which characterise its connectivity and the influence on processes that depend on their topology (Costa et al., 2007). Depending on the network, on the analysis task and on the processes using the network, a specific set of features may be considered. We are interested in complex networks especially because real networks have characteristics which are not explained by uniformly random connectivity. Instead, networks derived from real data may involve community structure, power law degree distributions[2] and hubs (nodes of a network with a large number of connections), among other structural features. In this section we shall then describe models of complex network generation that can help on the construction of real world process models and measurements that can help us characterise those models. Comparative analysis of some of the presented models can also be found in (Costa et al., 2007; Toivonen et al., 2009).

[2]A power law degree distribution is a distribution of network node degrees that follows a power law. A power lay exists when the frequency of an event (in this case the quantity of nodes with a given degree) varies as a power of some attribute of that event (such as the size of a network). Scale-free networks degree distribution follow a power law (Barabási and Albert, 1999).

Erdös-Rényi Random Graphs

Erdös-Rényi random graphs can be considered the most basic models of complex networks described in (Erdős and Rényi, 1959). These models generate random graphs consisting of N vertices and M edges. Starting with N disconnected vertices, the network is constructed by the addition of M edges at random, avoiding multiple and self connections. Another similar model defines N vertices and a probability p of connecting each pair of vertices (see figure 2.3 as an example). The latter model is widely known as Erdös-Rényi (ER) model.

For the ER model, in the large network size limit $N \longrightarrow \infty$, the average number of connections of each vertex $\langle k \rangle$, given by:

$$\langle k \rangle = p(N-1) \tag{2.7}$$

diverges if p is fixed. Instead, p is chosen as a function of N to keep $\langle k \rangle$ fixed:

$$p = \frac{\langle k \rangle}{(N-1)} \tag{2.8}$$

For this model, $P(k)$ (the degree distribution) is a *binomial distribution* (see figure 2.3). For large random networks the distribution is *Poisson*. This distribution can be derived as a limiting case to the binomial distribution as the number of nodes goes to infinity and the attachment success is fixed $n \times p = constant$. Therefore it can be used as an approximation of the binomial distribution if n is sufficiently large and p is sufficiently small.

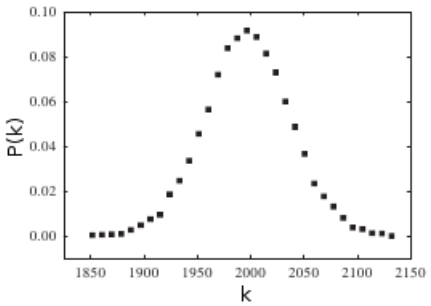


Figure 2.3: Average degree distribution over 10 random networks constructed with an ER model, formed by 10,000 vertices using a probability $p = 0.2$ (Costa et al., 2007).

This degree distribution makes the random network a poor approximation to real-world networks highly skewed degree distributions. On the other hand, the random graph has many desirable properties, particularly the fact that many features of its behaviour can be calculated exactly (Newman et al., 2002).

Watts & Strogatz Small-world Model

The small-world network model proposed by Watts & Strogatz (Watts and Strogatz, 1998) (also reffered as Watts & Strogatz (WS) model) is constructed by rewiring regular networks to introduce increasing amounts of disorder. These networks can be highly clustered, like regular lattices, yet have small characteristic path lengths, like random graphs. They are called small-world by analogy with the small-world phenomenon (Travers and Milgram, 1969), popularly known as six degrees of separation. This phenomena refers to the idea that everyone is on average approximately six steps away, by way of introduction, from any other person on Earth. A structured overview over the small-world research can be found in (Schnettler, 2009).

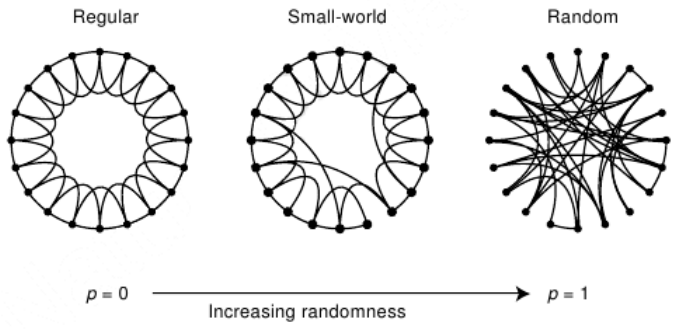


Figure 2.4: Random rewiring procedure for interpolating between a regular ring lattice and a random network (Watts and Strogatz, 1998).

The model constructs the networks by starting from a ring *lattice*[3] with n vertices and $2k$ edges per vertex, each edge is then rewired at random with a probability p (see figure 2.4. This construction allows for the adjustment of the graph between regularity ($p = 0$) and disorder ($p = 1$), and thereby to probe the intermediate region $0 < p < 1$, about which little is known.

The structural properties of this small-world network model are characterised by their path length $L(p)$ and clustering coefficient $C(p)$, as defined in figure 2.5. $L(p)$ measures the typical separation between two vertices in the graph (a global property), whereas $C(p)$ measures the cliquishness of a typical neighbourhood (a local property). We can see in figure 2.5 that the path length $L(p)$ stays almost as small as a random graph for a broad interval of p. We can also observe that the *cluster coefficient* stays as high as on a lattice except for large values of p. The rewiring mechanism creates "shortcuts"that reduce the distance not just between

[3]A regular lattice is a network where each node has the same number of connections. These networks are constructed by arranging the nodes in a ring and connecting each node to their next k neighbours.

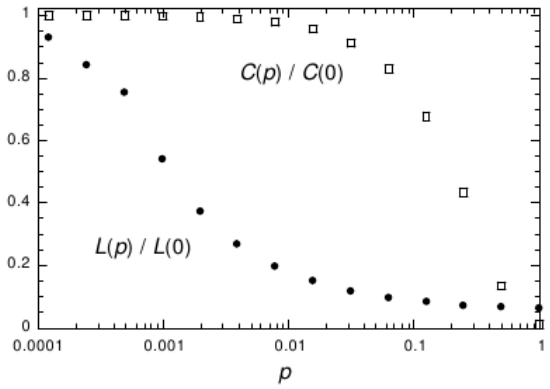


Figure 2.5: Characteristic path length $L(p)$ and clustering coefficient $C(p)$ (normalised by the values L(0), C(0) for a regular lattice) for the family of randomly rewired graphs described in figure 2.4 in relation to the rewiring probability p. The data showed in the figure averages 20 random realisations of the rewiring process (see figure 2.4) with 1000 nodes (Watts and Strogatz, 1998).

pairs of vertices that it connects, but between their immediate neighbourhoods, neighbourhoods of neighbourhoods and so on. There is a high non-linear influence on L (*average path length*) especially for low values of p.

Scale-free Network Models

Barabási Albert Model The Barabási Albert (BA) scale-free model (Barabási and Albert, 1999) , builds upon the perception of a common property of many large networks, a scale-free power-law distribution of node connectivity. This feature was found to be a consequence of two generic mechanisms: networks expand continuously by the addition of new vertices, and new vertices attach preferentially to sites that are already well connected.

In the model proposed in (Barabási and Albert, 1999), *Barabási* and *Albert* expose a property of large real networks, independent of the system and the identity of its constituents. This is, the probability $P(k)$ of two nodes being connected to each other decays as a power law, following $P(k) \sim k^{-\gamma}$. This result indicates that large networks self-organise into a scale-free state, a feature unpredicted by the previous random network models.

Large random networks share the common feature that the distribution of their local connectivity is free of scale, following a power law for large k with an exponent γ between 2.1 and 4, which is unexpected within the framework of the existing network models.

The main contrast one can make with the previously discussed models of *ER* and *WS* is that the probability of finding a highly connected vertex (that is, a large k)

decreases exponentially with k; thus, vertices with large connectivity are practically absent. In contrast, the power-law tail characterising $P(k)$ for the networks studied in this model indicates that highly connected (large k) vertices have a large chance of occurring, dominating the connectivity. This also means that *small-world* properties are also present in this model due to its highly connected nodes also denoted as *hubs*.

The model proposed by *Barabási* and *Albert* in (Barabási and Albert, 1999) is constructed starting with a small number m_0 of vertices. At every step we add a new vertex with $m(\leqslant m_0)$ edges that link the new vertex to m different vertices already present in the system. To incorporate preferential attachment, we assume that the probability $\prod$ that a new vertex will be connected to vertex i depends on the connectivity k i of that vertex, so that $\prod(k_i) = \frac{k_i}{\sum_j k_j}$. After t time steps, the model leads to a random network with $t + m_0$ vertices and $m \times t$ edges. The network evolves into a *scale-invariant*[4] state with the probability that a vertex has k edges, following a power law with an exponent $\gamma = 2.9 \pm 0.1$. Scale invariance is basically a feature of objects or laws that does not change if scales of length are multiplied by a common factor. In this case, the scale-free properties of this model are not disturbed by the number of nodes in the network.

Scale-free networks with adjustable cluster-coefficient We can create networks with the properties described in the BA, but considering tunable *cluster coefficient*. A model capable of achieving this is presented in (Herrera and Zufiria, 2011). The model generates scale-free networks by growing a scheme which employs random walks as a local approximation to the preferential attachment criterion.

Fitness-Based Model for Complex Networks Other interesting variation of the BA model is a fitness-based network model (described in (Fan, 2005)) that uses a "better-get-richer"instead of "richer-get-richer"growth algorithm (the later being the one used in the BA model).

Flexible Network Models

Structurally Induced Random Graph Model The Structurally Induced Random Graph (SIRG) model, described in (Conway, 2009) means to compensate for a shortcoming of the previously described models, which is the assumption that vertices exist in a vacuum, bringing no exogenous structure to the network system and only forming endogenous structure once inside a network. The model tries to overcome this limitation by imposing some structure for nodes entering a network.

[4]Scale invariance is a feature of objects or laws that does not change if scales of some variable are multiplied by a common factor. In this case, scale-free network properties are independent from network size.

This structure will likely be similar to structure components already observed in the network. The network generation process can be stated as follows:

1. Begin with some base structure G of arbitrary size and topology.

2. Given some integer $\tau > 1$, get a set I containing all the single-component (connected graph) subgraphs i formed by τ vertices.

3. Define S as an ordered n-tuple of the form $S = i_1, i_2, \cdots, i_n$ ordered by number of vertices and edges.

4. Define the function $c(i_k, G)$ to count the number of subgraph isomorphisms (see figure 2.6 for an example of isomorphic graphs) of $i_k \in G$ and generate a probability distribution over S as defined in equation:

$$F(i_k) = \frac{c(i_k, G)}{\sum_{m=1}^{n} c(i_m, G)} \tag{2.9}$$

   This function gives us the probability of some subgraph $i_k \in S$ being the next structural component of the graph G by calculating the number of subgraph isomorphisms found for $i_k \in G$ divided by the total number of subgraph isomorphisms counted $\forall i \in S$.

5. Draw i_k from the previously generated probability distribution and add it to the network structure using a decision rule. This decision rule is denoted $R(.)$ and is denoted as a mapping $R : i_n \to G$. This rule must ensure that the new elements are added according to the network theoretical constructs. It is left ambiguous as the SIRG is meant to be a framework from which any number of possible models could be derived.

6. Repeat steps $4 - 5$ until the network has grown sufficiently to meet some termination criteria, for example a desired property being achieved.

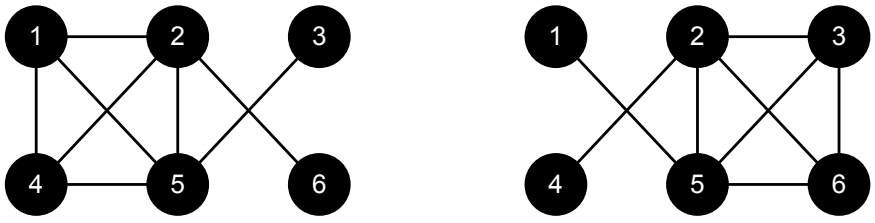


Figure 2.6: Example of graph isomorphism. Two objects are isomorphic if they are indistinguishable given only a selection of their features. In this case, two isomorphic graphs have the same edge structure regardless of their identification labels.

The rule by which the new structure is added to the current graph G will follow some kind of *fitness* according to the the fundamental constructs of G, meaning that they must be relevant to the structures represented by the base graph. To determine the next construct of G it is necessary to generate some finite set of possible realisation of G based on the subgraphs selected from S and the decision rule for adding that structure. The most fitted graph is then selected from that set.

This model has an interesting characteristic that makes is very appropriate for the work here presented. One can use it to grow previously generated networks towards desirable properties. An example of such growth is presented in the figure 2.7.

(a) SIRG base Structure

(b) Resulting structure by using the SIRG model to grow the base structure in figure 2.7(a) maximising the transitivity.

Figure 2.7: Example of the usage of the SIRG model to grow a base structure 2.7(a) into a network 2.7(b) according to a desired property, in this case, the maximisation of transitivity. The $\tau = 4$ vertices and the termination criteria is the number of vertices n being $n \geq 200$) (Conway, 2009).

Generalised random graphs The Generalized Random Graphs (GRG) described in (Molloy and Reed, 1995) allow for the generation of networks with a given degree distribution. This aspect is particularly useful for comparison with real networks with similar characteristics.

The method used to generate this kind of random graph involves selecting a degree sequence specified by a set $\{k_i\}$ of degrees of the vertices drawn from the desired distribution $P(k)$. Afterwards, to each vertex it is associated a number k_i of "stubs"(ends of edges emerging from a vertex) according to the desired degree sequence. Next, pairs of such stubs are selected uniformly and joined together to form an edge. When all stubs have been used up, a random graph that is a member of the ensemble of graphs with that degree sequence is obtained. The procedure was later addressed by Newman et al. (Newman et al., 2002) regarding its application to real-world social network modelling.

Tunable Cluster Coefficient and Degree Distribution In (Newman, 2003a), *Newman* describes a model of networks with both a tunable degree distribution and a tunable clustering coefficient. Such models are desirable when we need to use network structures with values for these properties. Particularly, this becomes useful when we want to model specific target social systems from which we have such information just like the previous *generalised random graph model*.

The model is based on the idea that clustering in networks arises because the vertices are divided into groups (Ravasz and Barabási, 2003), with a high density of edges between members of the same group, and hence a high density of triangles, even though the density of edges in the network as a whole may be low (Newman, 2003a). These structures, also known as *communities* are commonly found in many real-world social scenarios.

The model is constructed by considering a network of N individuals divided into M groups. A social network, for example, might be divided up according to the location, interests, occupation, and so forth. Individuals can belong to more than one group, the groups they belong to being chosen at random in the model. Individuals are not necessarily acquainted with all other members of their groups. If two individuals belong to the same group then there is a probability p that they are acquainted and $q = 1 - p$ that they are not; if they have no groups in common then they are not acquainted. In addition to the probability p, the model is parametrised by two probability distributions: r_m is the probability that an individual belongs to m groups and s_n is the probability that a group contains n individuals.

Other Models Of Interest

Another models of interest include:

- a model based on genetic variation in human social networks, presented in (Fowler et al., 2009).
- models based on social interaction theory that include growth by meeting strangers and friends of friends (see (Jackson and Rogers, 2007)).
- complex network structures formed by models based on social interaction theory and social distances (Boguñá et al., 2004; Pujol et al., 2005; Jager and Amblard, 2008).
- models that try to span their topology characteristics, generalising some models previously presented (Leskovec et al., 2010; Leskovec and Faloutsos, 2007).

Finally there is a very interesting model that generators networks with a very particular topology with intrinsic community structures (see section 2.1.1). The networks generated are based on empirical data of terrorist organisations (Tsvetovat and Kathleen M. Carley, 2005). These networks are of the most importance as our aim is to explore different social space structures to be included in social simulation models. Using these, one can model social systems that have behaviours similar to those of highly clustered, sparsely distributed organisations. This means that while large *hubs* still dominate the network connectivity, the presence of tight clusters (cells) continues to provide local connectivity if the *hubs* were to be removed (see figure 2.8).

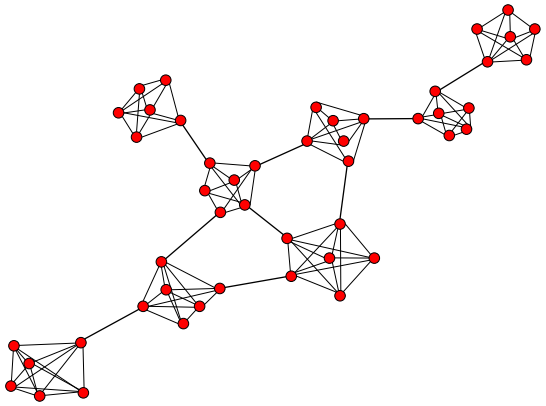


Figure 2.8: An illustrative example of a cellular network approximate topology (Tsvetovat and Kathleen M. Carley, 2005).

2.2 Computational Simulation, Agent-based approaches And Social Simulation

In this section we outline simulation techniques and tools, basic concepts of *social simulation* and appropriate methods used in this area of research. It shows the power of some approaches like *agent-based* social simulation to help on the understanding of the social simulation model to be later presented. Finally, we describe some social simulation models that are relevant to our work. The models described encompass the usage of complex social networks to model social systems and the processes executed within those topologies. This is relevant to our work especially because we rely on them to construct our *multiplex network structures* (Hamill, 2006; Antunes et al., 2008, 2009) capable of representing the complexity of real multi-dimensional social spaces.

2.2.1 On Computational Simulation

Simulation can be seen as a technique to represent or abstract a process or behaviour for analytical, decision support or learning purposes (Pitt, 2008). The technology boom of the 1990s brought the ability to use models and simulations in nearly every domain. This technique allows us to better understand human behaviour, enterprise systems, disease proliferation, etc. In general, it allows us to understand the behaviour of complex systems by exploring different model and experimental designs for simulating systems that would be very difficult to manipulate and directly experiment with in real-life (El Sheikh et al., 2007). One case of such complex domains is the construction of models for social sciences. The major reason for social scientists becoming increasingly interested in computer simulation is its potential to assist in discovery and formalisation of the dynamics within the simulated processes. Social scientists can build very simple models that focus on some small aspect of the social world and discover the consequences of their theories in the *artificial society* that they have built (Gilbert and Troitzsch, 2005).

It follows that simulation models generally cannot be solved like analytic models which can be computed by mathematical techniques, like algebra, calculus or probability theory (Law and Kelton, 2000).

2.2.2 Simulation And Modelling Process

The process of modelling and simulation passes through four phases of a cyclic movement: modelling, code, execution, and analysis. Each phase depends on a different set of supporting technologies (Sokolowski and Banks, 2010):

1. **modelling phase** - modelling technologies;

2. **implementation phase** - development technologies;

3. **execution phase** - computational technologies;

4. **analysis phase** - data/information technologies;

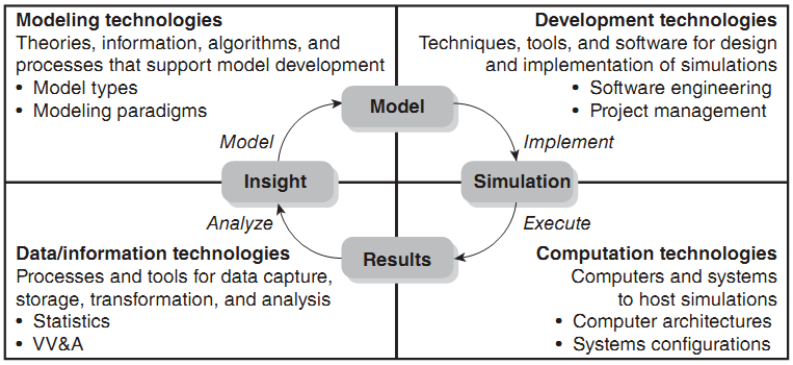


Figure 2.9: Modelling and simulation cycle and relevant technologies (Sokolowski and Banks, 2010).

Figure 2.9 illustrates the modelling and simulation phases and their related technologies. The figure also depicts two processes: the phases used in the development and testing of computer models and simulations and the phases involved in the application of simulation and modelling to the investigation of real-world systems.

2.2.3 Simulation Types And Discrete-event Simulation

We can identify three types of general simulation models (El Sheikh et al., 2007; Sokolowski and Banks, 2010):

1. scale models of the real system;

2. physical system models *vs* a mathematical representation of the system;

3. a set of mathematical equations and logical relationships;

Within the third type, and when the equations cannot be solved analytically or numerically, we can use computer simulation models to construct scenarios based on these mathematical entities.

In this kind of models we can identify two specific types of simulations: discrete and continuous. The continuous simulation models focus on a smooth change through time while the discrete ones focus on specific events occurring at specific points in time (El Sheikh et al., 2007). The variables in a simulation model can thus change in time in the following manner (Pidd, 2004):

1. Continuously at any point of time: values change smoothly and are accessible at any point of time;

2. Continuously changing but only at discrete time events: values change smoothly but only at a predetermined time;

3. Discretely changing at any point of time: state changes are easily identified but occur at any time;

4. Discretely changing at specific points in time: state changes can only occur at specific points in time;

The last two cases are often described in the community as *discrete-event simulation* which is a simulation where state changes occur in a discrete manner but possibly at random simulated points in time (Wainer, Gabriel A. and Mosterman, 2010; Schriber and Brunner, 2006).

2.2.4 Agent-based Modelling and Simulation

Agent-Based Modelling and Simulation (ABMS) is a computationally demanding technique having its origins in discrete event simulation and cellular automata. It is a powerful technique for simulating dynamic complex systems and observing emergent behaviour (Allan, 2009). The systems are modelled using autonomous, interacting agents. ABMS promises to have far-reaching effects on the way that businesses use computers to support decision-making and researchers use electronic laboratories to support their research. Some like *Axelrod* (Axelrod, 1997a), have gone so far as to contend that ABMS is a third way of doing science besides deductive and inductive reasoning.

What Is An Agent? There is no universal definition for an agent, but one can thing of it as a software entity with independent component behaviour that can range from primitive reactive decision rules to complex adaptive intelligence. A comprehensive computer science view over the theory of agency can be found in (Jennings, 2000). From a practical modelling standpoint, we can consider agents to have certain characteristics (see (Macal and North, 2005)) such as:

- An agent is identifiable, a discrete individual entity with a set of characteristics and rules governing its behaviours and decision-making capability. Agents are self-contained. The discreteness requirement implies that an agent has a boundary and one can easily determine whether something is part of an agent, not a part of an agent, or a shared characteristic;

- An agent is situated in an environment. It interacts with it as well as with other agents. Agents have protocols for interaction with other agents, such as communication protocols, and the capability to respond to the environment. Agents have the ability to recognise and distinguish the traits of other agents;

- An agent is goal-directed, having goals to achieve with respect to its behaviours;

- An agent is autonomous and self-directed. An agent can function independently in its environment and in its dealings with other agents, at least over a limited range of situations;

- An agent is flexible, and has the ability to learn and adapt its behaviours over time based on experience. This requires some form of memory. Moreover, an agent may have the capability for modifying its own behaviour;

The need for Agent-based Modelling We aim to model and analyse social systems, the phenomena to be addressed are complex in terms of their interdependencies. This means that the traditional modelling tools may not be as applicable as they once were. The approach we consider for the work presented in this thesis is based on *multi-agent models*.

Agents bring the notion of locality of information together with locality of intent or purpose. The relation between multi-agent and simulation systems is multi-faceted. Simulation systems are used to evaluate software agents in virtual dynamic environments. Agents become part of the model design to represent for instance the behaviour of human social actors in a context of social simulation (Uhrmacher and Swartout, 2003).

Discrete-event Agent-based Simulation Agent-based simulation models are not necessarily separated from the rest of the techniques presented. In (Dubiel and Tsimhoni, 2005), agent-based modelling is considered a technique suitable to simulate the real-time interaction of people with their environment. The approach integrates agent-based modelling with discrete event simulation to simulate the movement of people in a discrete event system. We are interested in this integration to construct the model presented in this thesis as it provides a good abstraction of real-world scenarios.

There are numerous tutorials on agent-based modelling and simulation, among them, (Macal and North, 2005, 2008). There is also a wide range of agent-based simulation applications that can be found in (Davidsson et al., 2007).

2.2.5 Agent-based Simulation and Modelling Tools

In this section we describe some simulation tools of interest. Our main focus is on the tools that allow us to prototype and create agent-based social simulation models. In particular, we want to integrate complex network structures in our model, as such, the tools considered should allow for that modelling aspect.

MASON: Multi-Agent Simulator Of Neighbourhoods (MASON) is an extensible, discrete-event multi-agent simulation toolkit developed in Java (see (Luke et al., 2005)). It is designed to serve as the basis for a wide range of multi-agent simulation tasks, ranging from swarm robotics to machine learning to social complexity environments. MASON carefully delineates between model implementation and visualisation, allowing models to be dynamically attached to visualisers. To create new models and visualisation schemes one must extend the Java framework provided. The design appears to have been driven largely by the objectives of maximising execution speed and ensuring complete reproducibility across hardware. The ability to detach and re-attach graphical interfaces and to stop a simulation and move it between computers are considered a priority for long term simulations (Railsback et al., 2006; Allan, 2009). MASON was not a very mature platform in the past but it now seems a promising platform with lots of features added recently. This platform is particularly suitable to the model to be developed in this thesis as it includes some facilities to deal with social network integration, representation and analysis.

NetLogo: NetLogo (Wilensky, 1999) clearly reflects its heritage as an educational tool, as its primary design objective is clearly ease of use even for non-experts in programming. Its programming language includes many high-level structures and primitives that greatly reduce programming effort, and extensive documentation is provided. The language contains many but not all the control and structuring capabilities of a standard programming language. Furthermore, NetLogo was clearly designed with a specific type of model in mind: mobile agents acting concurrently on a grid space with behaviour dominated by local interactions over short times. While models of this type are easiest to implement in NetLogo, the platform is by no means limited to them (Railsback et al., 2006; Allan, 2009). It should be stated that, in spite of its educational context, NetLogo is an excellent tool for quick prototyping and visualisation of simple simulation models. This is an adequate tool to design our *social segregation* model prototype prior to its implementation in a platform more suited for large-scale experiments.

Repast: Repast (Collier, 2003) development appears to have been driven by several objectives. One of them is the intent to support one domain in particular, social science. It includes tools specific to this domain. The additional objective of making it easier for inexperienced users to build models has been approached in several ways by the Repast project. These approaches include a built-in simple model, and interfaces through which menus and Python code can be used to begin model construction (Railsback et al., 2006; Allan, 2009). One interesting thing is that Repast is one of the few simulation/modelling software systems that supports the integration of geospatial data out-of-the-shelf (*MASON* also includes extensions that supply such functionalities) (see (Crooks, 2007)) which can be useful to create models when geographical data is available.

2.2.6 On Social Simulation

In this section we describe some important social simulation models as well as examples of models that make use of complex networks to represent social systems.

Social Simulation and Modelling (SSM) is regarded as the usage of computational simulation and modelling techniques to the study of human social phenomena, including residential segregation (Schelling, 1969), group formation (Desjardins, 2005), transmission of culture (Axelrod, 1997b), propagation of disease and population dynamics (Epstein and Axtell, 1996; Gilbert and Troitzsch, 2005).

The first attempts to apply explicitly, *agent-based computer modelling* to social science explicitly are made by *Thomas Schelling* in a series of papers, "Models of Segregation"(Schelling, 1969) , "On the Ecology of Micromotives"(Schelling, 1971b) , "Dynamic Models of Segregation"(Schelling, 1971a) and later in a book "Micromotives Macrobehavior"(Schelling, 1978). In these papers, *Schelling* anticipated many themes in the contemporary literature on agent-based modelling, social complexity and economics. His efforts were constrained by the limited computational power available at that time. Only in the last decade advances in computing have made large-scale agent-based modelling practical (Epstein and Axtell, 1996).

The reader is referred to (Epstein and Axtell, 1996; Axelrod, 1997a; Gilbert and Troitzsch, 2005; Bardón, 2009) for a deeper understanding of social simulation. Some insights on tools and techniques used in social simulation model construction can be found in (Suleiman et al., 2000). For agent-based models applied to complex system modelling and social sciences in particular, one should refer to (Epstein, 1999; Goldspink, 2000; Sansores and Pavón, 2006; Troitzsch, 2009). Finally some interesting insights regarding the construction of adequate and reliable models to correctly describe social phenomena are addressed in (Edmonds, 2010).

2.2.7 Social Simulation Models

In this section we refer to some interesting models that demonstrate the power of using complex network structures in social simulation. We show some of phenomena that can be modelled recurring to these structures. We describe models that include: diffusion processes spreading an innovation or behaviour through a social network of agents, opinion dynamics and segregation phenomena.

Models of Segregation

Social segregation phenomena refer to the separation of social actors into different social interaction groups. In (Schelling, 1969, 1971b,a), *Schelling* devised a simple spatially distributed model of the composition of neighbourhoods, in which agents prefer that at least some fraction of their neighbours be of their own "colour". He found that even quite colour-blind preferences produced quite segregated neighbourhoods. The model was constructed using a bi-dimensional grid where the agents interact with local neighbours and relocate according to a value of tolerance to differences in colour in the neighbourhood. The results show that segregation occurs regardless of the order of the agents. Extreme ratios lead to minority forming large clusters, disrupting majority. Increasing the neighbourhood size considered by the agents increases segregation.

The model has been revised many times. An example includes (Pancs and Vriend, 2007), were the robustness of *Schelling's* model is analysed, focusing in its driving force: the individual preferences. This analysis show that even if all individual agents have a strict preference for perfect integration, best-response dynamics may lead to segregation.

In this dissertation, we aim to present a model that integrates a segregation process over existing network structures. Similar work also includes the adaptation of the *Schelling* model to complex social networks (Fagiolo et al., 2007). Although the essence of segregation is similar, our work is focused on the usage of multiple coexisting social networks to represent the social space. The structures used to represent social relations are extremely important in multiple-context dynamics (Antunes et al., 2008, 2009). The importance of network topologies along with segregation phenomena has been reported in (Tassier and Menczer, 2008), where a model of equality in a labour market is explored.

Models Of Diffusion over Social Networks

Network structures are fundamental for the representation of diffusion processes. Social structure provides ways to naturally spread information through its members. The way some entity spreads through a social system can be modelled very well through the usage of complex social networks. These networks can represent either the structure of social relations and roles or concrete infrastructures like on-line social networks. Understanding the way information spreads through a network is particularly useful not only in the context of social sciences but also in practical applications like marketing over real on-line social networks.

As an example of a model of diffusion, one can consider the work in (Nekovee et al., 2007) which presents a model with a rumour spreading mechanism. Rumours are an important form of social communications, and their dynamics plays a significant role in a variety of human affairs. The spread of rumours can shape the public opinion in a country, greatly impact financial markets and cause panic in a society during wars and epidemic outbreaks. The information content of rumours can range from simple gossip to advanced propaganda and marketing material (Nekovee et al., 2007). One could for instance explore such models to study the basis of viral marketing phenomena. Companies dedicated to this activity often exploit on-line customer social networks of their customers to promote products.

The model presented in (Kempe and Kleinberg) is another example of a diffusion model that uses social networks. In this model the problem being studied is the maximisation of the expected spread of an innovation or behaviour within a social network, in the presence of "word-of-mouth"referral. In some cases individuals decisions to purchase a product or adopt an innovation are strongly influenced by recommendations from their friends and acquaintances. Understanding and getting a leverage this influence may thus lead to a much larger spread of the innovation than the traditional view of marketing to individuals in isolation. This study lacks on the analysis of the topologies of social networks used in the diffusion process.

Different domains present different network structures, the difference in the topology of the networks may greatly affect the diffusion processes (Antunes et al., 2008, 2009). As such, it is important to understand the influence of different complex social network models on this and other processes that make use of such structures. The work presented in this thesis provides some insights in this area.

Models of Consensus, Auto-organisation and Cooperation

Auto-organisation is the process where a structure or pattern appears in a system without a central authority or external element imposing it through planning. These patterns emerge from the local interaction of the elements that make up the system, thus the organisation is distributed by nature.

The majority of social simulation models encompass such behaviour. The study of the dynamics of collective phenomena is desired for models that integrate such concepts. *Vazquez et al.* (Vazquez et al., 2009) address such phenomena by dealing with social consensus. In this work the problem of acquiring a global consensus is regarded as an auto-organisation phenomenon. This problem is approached from the perspective of non-linear dynamics of interacting agents in a complex network. The model construction is based on*Axelrod's* model (Axelrod, 1997b) for the dissemination of culture. Both *Axelrod's* model (Axelrod, 1997b) and *Vazquez* study on non-linear dynamics (Vazquez et al., 2009) are very relevant for a deeper understanding of similar collective phenomena dynamics such as the ones present in the model discussed in this thesis.

Another interesting model, described in (Chen et al., 2009), presents an insight on *social tolerance* and its role on *cooperation*. This is relevant to our work as we aim to construct a model of segregation between social contexts based on thresholds of *social tolerance*. In this work, social actors have moderate tolerance toward ambient cooperative environment. In this environment they tend to avoid unfavourable interactions and search for favourable ones. The focus of the study was on how social tolerance affects the evolution of cooperation. To address this issue, they present a model of co-evolutionary prisoner's dilemma. Although this work is centred in evolution of cooperation networks which is a little bit different from our proposed model, it presents an interesting result in which they state that moderate tolerance thresholds can result in the optimal cooperation levels.

Finally, we address a class of models focused on the dynamics and evolution of opinions. In (Weisbuch, 2004) a model called *bounded confidence model* is discussed. In this model, agents can influence each other's opinion provided that opinions are already sufficiently close enough. This work discusses the influence of social networks topologies on the dynamics of the bounded confidence behaviour.

Other model with similar characteristics to those of opinion dynamics is discussed in (Castelló et al., 2008). This work presents a model of language competition over complex social networks is used to study the dynamics of social consensus. The study is conducted using an agent-based model of competition between two socially equivalent languages, addressing the role of bilingualism and social structure.

A final interesting model (Rosvall and Sneppen, 2007) analyses the opinion phenomenon from a different perspective. It focus on agents that self-organise a dynamic network to facilitate their hunter-gatherer behaviour in information space. They maintain local opinions about the importance of their neighbours. The results show that tribal organisations and modular social networks can emerge as a result of contact-seeking agents that reinforce their beliefs among like-minded. They also found that prestigious social agents can streamline the social network into hierar-

chical structures around themselves. These findings are consistent with some of the network creation mechanisms previously presented and reinforce their importance.

2.3 Multi-context Model

As discussed previously, some models of multi-agent-based social simulation represent the social connections between agents with social network structures. The models described in (Antunes et al., 2008, 2009; Nunes and Antunes, 2012a) are a step further in modelling methodologies as they consider a multitude of concomitant social relations. This setting can be seen in a simulation as a n-dimensional scenario where each dimension surface represents a different social relation (see figure 2.10). Agents belong to distinct contexts in these multiple relations. In this line of work the concepts of *context permeability* and *context switching* in social relations are proposed. These concepts are illustrated using the simulation models in which agents interact using a simple consensus game. The society of agents has to adopt a binary "option "according to a majority rule. The speed of consensus (which is a good measure for auto-organisation) is observed for different network topologies.

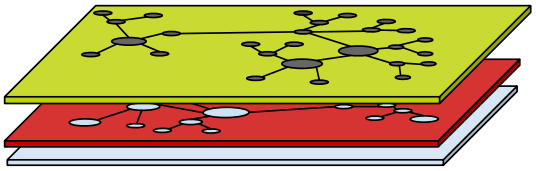


Figure 2.10: Multiplex social network structure forming the social space for our models of multiple concurrent social contexts.

The models described above are specially adequate because they capture the notion that different social relations may have different topologies forming a multiplex network structure (figure 2.10). For clarification purposes we should state that when we use the term *multiplex*, like many other concepts in SNA, this appears to be borrowed from communications theory, which defines multiplex as combining multiple signals into one to facilitate transmission, in such a way that they can later be separated as required. We can also conjecture that similar relations will have more local overlaps between them. Probably the earliest, formal recognition of multi-dimensionality among relationship was described by *Granovetter* (Granovetter et al., 1973), who suggested that "the degree of overlap of two individual's friendship networks varies directly with the strength of their tie to one another". One can also find an interesting insight on multiple parallel network structures in (Hamill, 2006).

Understanding the structure of social relations has been the focus of the social sciences. There is a particular interest in understanding how social structures are

formed and evolve. A social structure is a system of social relations tying distinct social entities to one another. To construct plausible scenarios in social simulation models, an understand the basics of social structure theory is desirable. The reader is referred to (Blau, 1977) for a deeper understanding of the basic building blocks for social system structure. The concepts illustrated in (Blau, 1977) regarding social relations, contexts and roles are part of the main idea behind the previously described models ((Antunes et al., 2008, 2009; Nunes and Antunes, 2012a)).

2.3.1 Context Permeability

Regarding *context permeability* (Antunes et al., 2008), we can link this concept to the previously described models of diffusion over social networks. By doing this, one can easily see the drawbacks of representing the social space using only one layer with a single network structure. By using the previously discussed network models with only one dimension, bottlenecks are easily identified within the network topology. Such bottlenecks present an obstacle for information diffusion. This can hold back information spreading phenomena and more importantly, it is harder or even impossible for a society of agents to achieve global consensus under certain scenarios. Context permeability states that a social actor is engaged in multiple complex networks of social links, and the existence of multiple-modality paths between agents allow for a permeability between different context topologies. This set-up makes it possible to overcome bottlenecks and allows for a more efficient fostering of global consensus.

Figure 2.11 shows an example for an abstract scenario in which two agents can belong to different social contexts at the same time. Within these kinds of scenarios we can represent the multiplicity of social roles social actors take into account when interacting with each other. In real world scenarios, social peers interact at different levels, sometimes at the same time. For example, two family members can work together or even belong to the same on-line social network or engage in other concomitant social relation. Modelling social spaces with multiple relational planes is thus a step forward in what concerns to modelling methodologies. The aggregation of social space into single network structures or abstract bi-dimensional grids can be sufficient for some models but it is too much simplistic to produce believable complex social scenarios.

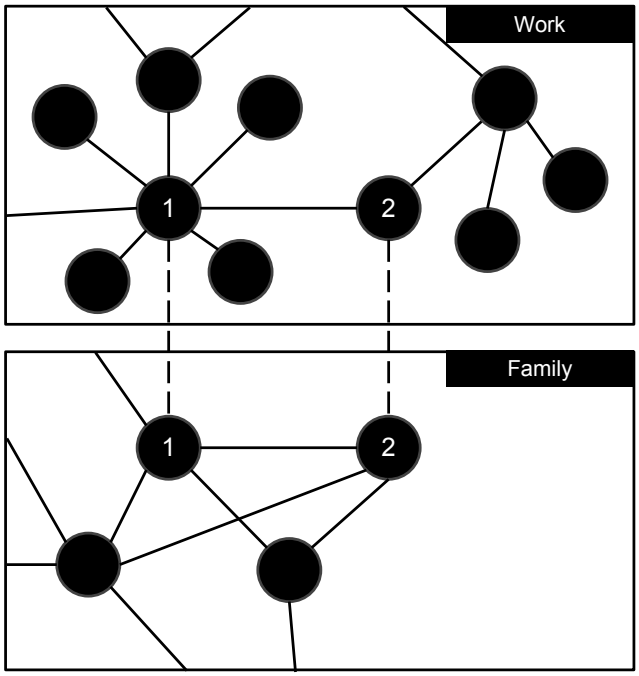


Figure 2.11: Example of context permeability (Antunes et al., 2008) considering two contexts for social agent denoted by the number 1 and 2. In this case, we have a family context and a work context. What happens in context permeability is that agents 1 and 2 can be interaction at the same time. This is possible in real scenarios. As an example, a social actor can have a social connection which is simultaneously his co-worker and a family member. In result, an interaction from these two actors can be affected by the two distinct social contexts in which they interact.

2.3.2 Context Switching

In the context switching model (Antunes et al., 2009), the society of agents engages in the same consensus game.

The agents are embedded in multiple relations represented as static social networks and they switch contexts (see figure 2.12) with some probability ζ_{C_i} associated with each context C_i. The agents are only active in one context at a time and can only perform encounters with available neighbours of the current context. We can think of context switching as a temporary deployment in another place, such as what happens with temporary immigration.

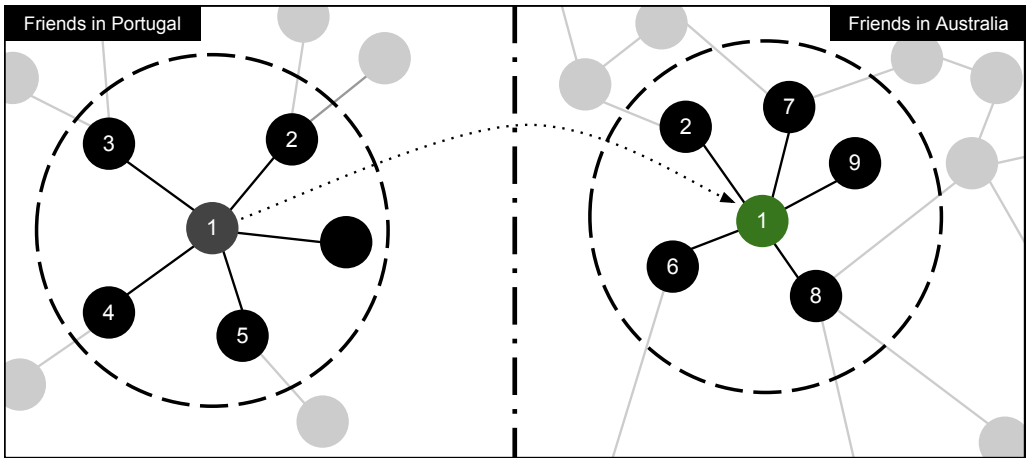


Figure 2.12: Example of context switching (Antunes et al., 2009) considering two contexts for social agent denoted by the number 1. In this case, these contexts are created by two distinct physical spaces. Common nodes in both neighbourhoods (like agent 2) represent the same social actor being able to travel between both distinct contexts, representing an acquaintance of actor 1 in both of them. The dashed circle represents the scope of each context.

The behaviour of the agents in this simple model can be described as follows:

1. choose an available neighbour from the current context (neighbourhood of the network structure where the agent is currently located);

2. check the selected interaction partner current choice and increment the memory for the number of individuals "seen" with that choice;

3. check for the choice that has the majority and switch to it if the current opinion differs;

4. switch to a random distinct context C_j (located in another network) with a probability ζ_{C_i}, which is a parameter of the model related to each social context C_i.

This model presents a way to represent time spent in different contexts in an abstract manner using the switching probability ζ_{C_i}. With this wan can focus on the temporal component of permeability between contexts. Context switching introduces one notions that has not been explored in the literature so far: the fact that, although some social contexts can be relatively stable if we consider short to moderate periods of time, our social peers are not always available at all times and spend different amounts of time in distinct relations.

2.4 Summary

In this chapter, we made an extensive review over complex social network models and the underlying notions of graph theory and social network analysis. We also discussed related computation simulation methods, agent based approaches and the most recent advances in social simulation that contemplate the usage of network structures to construct social space scenarios. Such review did not exist in the literature and part of our future work will be focused in making it available to the scientific community.

We have also presented the previous work on multi-context models, introducing the notions of context permeability and context switching. These models form the fundamental basis for the model that will be later presented in chapter 4. We chose to use the simple game of consensus in our models as we are interested in exploring the mathematical properties of our multi-relational model and the influence of different network topologies in dissemination processes. With this we want to construct the basis for a consistent multi-context modelling approach that, although being currently treated as abstract, can be applied to more complex social scenarios.

Chapter 3

Model Construction and Deployment

In this chapter we describe formal and informal methodologies used to construct our model of *context segregation*. We also give some insight about the tools used in the model prototyping and development process as well as the methods to explore a model parameter space using grid computing. Our aim is to create a comprehensive understanding on how the models and experiments are designed and deployed.

This chapter is organised as follows. In the first section, we cover the tools used to create the prototype of the model developed in this thesis. The following section then overviews the *b-have workbench* project and explain its role in our experiments. The contribution to this project is also outlined in this section. We finish by discussing how one can use grid computing to explore social simulation experiments in a parallel manner. In this final section we discuss, both formally and informally, the tools used to create our final model of experiments, how this model is integrated with existing grid computing technologies and what are the expected performance gains when using such technology.

3.1 Simulation Model Prototyping

In this section we present our prototyping platform. We used *NetLogo* (Wilensky, 1999) to create a first prototype of our model of *social segregation* (Nunes and Antunes, 2012a). The model itself will be presented in detail in the next chapter. For now, we focus on the tools and the experimental setup itself.

We prototyped our model using this tool as the previous model of *context switching* (Antunes et al., 2009) (from which our model derives) was completely designed in it. With some changes, we constructed a working model incorporating the target phenomenon of *social segregation*.

NetLogo is great for prototyping as it allows to quickly build a user interface with which we can configure our model of experiments. An example of the prototype

model interface can be seen in figure 3.1.

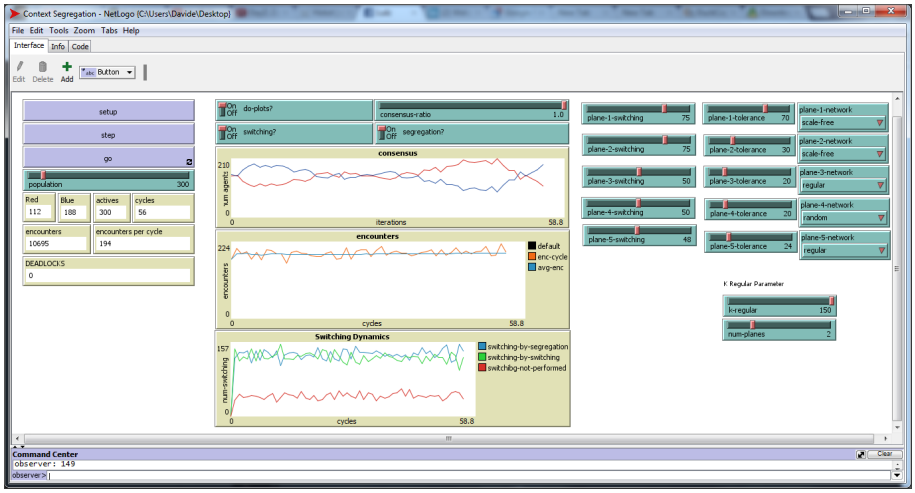


Figure 3.1: NetLogo prototype interface. The buttons, sliders, combo boxes and toggle buttons can be used to configure the simulation model. The plots and monitor boxes can be used to display the simulation measurements.

This interface also allows to quickly visualise the results of simulations. This provides a quick debugging mechanism through observation and monitoring of the models behaviour. Our model also used a three-dimensional (3D) view for representing multiple social layers (see figure 3.2). The population of agents is distributed throughout all the layers (the number of layers can be configured in the interface).

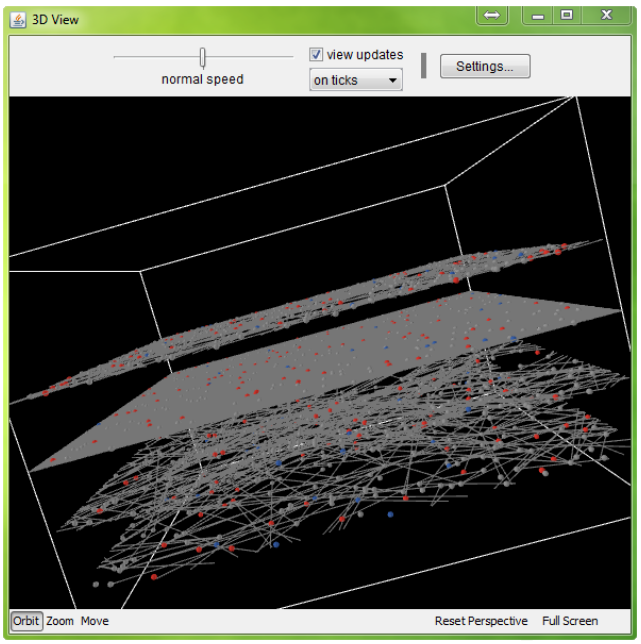


Figure 3.2: NetLogo 3D cube view. Each plane represents an abstract social relation. The current choice of an agent is represented by its colour. An agent with grey colour is inactive in the planes in which it is grey.

Each plane on the 3D cube has all the agents in the population although each agent is only active in one plane on one given moment. Network structures are built for each social relation. These networks, like in the previous model of *context switching*, were constructed with the network models discussed in the previous chapter. The algorithms for network creation were written in *NetLogo*.

To compare our models, we used the *b-have workbench* (to be discussed in the next section) to create network instances, export them to files and load them both on the previous model of *context switching* and our current model of *context segregation*. This allowed us to compare the behaviours of both models while ignoring the variability of the networks being used. This allows for a better debugging process. We also used these network to compare the prototype model with a final simulation model written in MASON (Luke et al., 2005).

We used this first mode prototype written in *NetLogo* to test some modelling approaches and measurements before the construction of our MASON implementation. Nonetheless, we made a fully functional, self-contained model instance available[1].

The final model implementation in MASON and its deployment on a grid environment will be discussed in section 3.3.

[1]The model prototype can be found at the *OpenABM* model library: http://www.openabm.org/model/3081

3.2 B-have Workbench

The *b-have workbench* (Nunes and Antunes, 2011) is a tool to create social simulation components such as social networks (see figure 3.3), agent models, behaviour rules and environment abstract models. We use it to construct complex network models that can be stored and exported to be used as components in the model developed in this thesis.

Social simulation requires a series of components to be modelled prior to the experiment setup. Such components are not usually decoupled from the experiment process. This fact creates a series of problems regarding experiment reproduction and model re-usability. This tools aims to provide a separation of concerns between modelling and experiment processes. Moreover, the techniques proposed in (Nunes and Antunes, 2011) present a set of methods to partition various models in simple components that can be easily combined to create new ones.

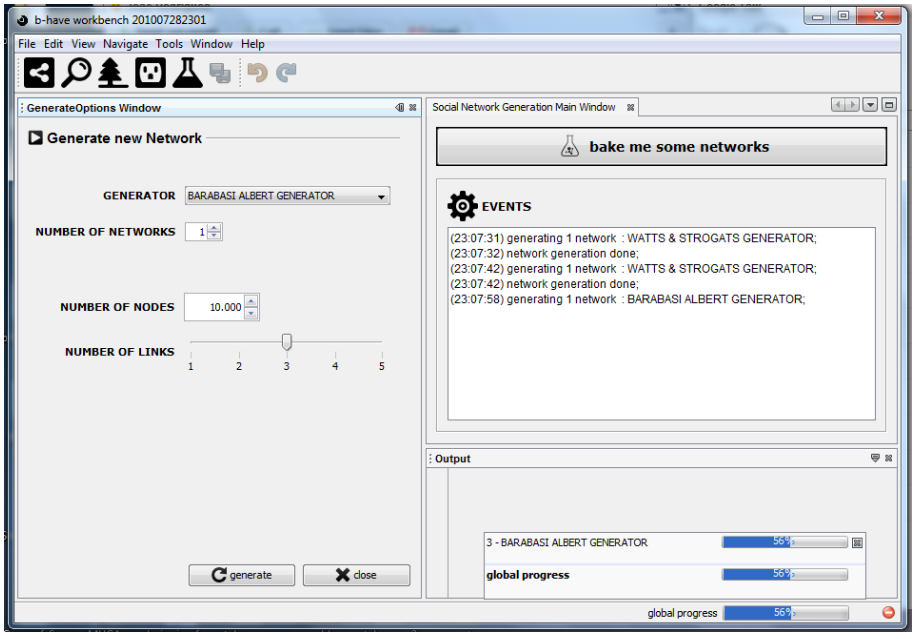


Figure 3.3: B-have workbench network generation tool.

3.2.1 The methodology behind the b-have project

As we are creating components to design social simulation experiments, we defend and adopt the KISS (keep it simple, stupid!") principle Axelrod (1997a). In some sense, we can say that Sloman's design principle starts off from this Sloman (1993),Sloman (1994). Simplicity and clear definition are then the major properties to achieve. The components to be created comprise the ability to take more shallow forms and become increasingly deepened due to their refactoring properties. This is useful because we can add more complexity to the components as we gain insight and understanding about the problem at hand. The idea is to explore the design of agents, interactions, environments, institutions and societies by making the initially simple particular notion used increasingly more complex, dynamic, and rooted in consubstantiated facts (see Antunes et al. (2007)).

The methodology behind the b-have workbench comprise then some important considerations. First of all, we propose that social simulation must incorporate separation of concerns between the modelling process and the simulation process. The partition generated between these two processes is very important because it allows us to decompose very complex models into separated components. Understanding a given component dynamics is easier than trying to decode the dynamic properties of a complete model. Second off all, given that the partition of the modelling space is well implemented, the components themselves must be specified in a clear and simple way without compromising flexibility or introduce dependence on a specific social theory, the motivation behind this desired properties are well defined in Edmonds (2003).

The successful partition of the social models allows us to arrive at better explanations for emergent processes on a given experiment. Moreover, the main rationale behind the conception of this project is that social scientists should focus on modelling processes rather than implementation issues.

We consider four main components for the clear partition of the social model space (see figure 3.4). Social networks as the means to represent the boundaries on the actor relations. Social agent models that comprise all their important properties while "freezing" every thing else, for the sake of simplicity. Environment models, which contextualize the agent interaction and help to construct more concise models of the reality and finally behaviour rules that must be flexible enough to represent both, changes in the environment, agents and interaction processes considering all the other components.

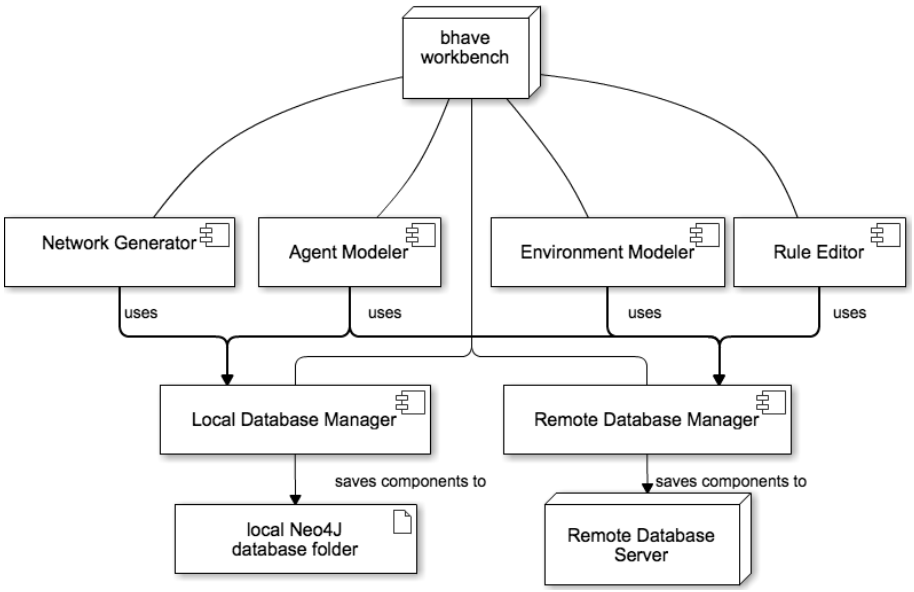


Figure 3.4: B-have workbench architecture overview.

3.2.2 Implementation considerations

To implement the b-have workbench, some considerations were made regarding what technologies to use and how those technologies provide means to accomplish the main objectives of the application. First of all, the b-have workbench was implemented using Java 6. This choice was based on the cross-platform properties of the language and easy maintenance. As for performance, virtual machines are becoming faster each day, so using this language does not bring performance issues in the computational tasks to be performed.

Using plain old java objects to build a application from scratch could bring numerous problems into this tool implementation. So, to avoid design problems and ensure that the tool was easy to extend in the future, the development process was made using the Netbeans Rich Client Platform framework (see figure 3.3). This way the tool can continue to grow in an easy and modular fashion.

3.2.3 B-have Network API

The *b-have workbench* incorporates a set of network generation algorithms in a rich-client application (a desktop application). In this thesis we created a *Java API*[2] (Application Programming Interface) that allows for the network generation algorithms to be used with other *Java* applications. This is useful to integrate the creation of social networks in frameworks like MASON (Luke et al., 2005) or Repast (Collier, 2003).

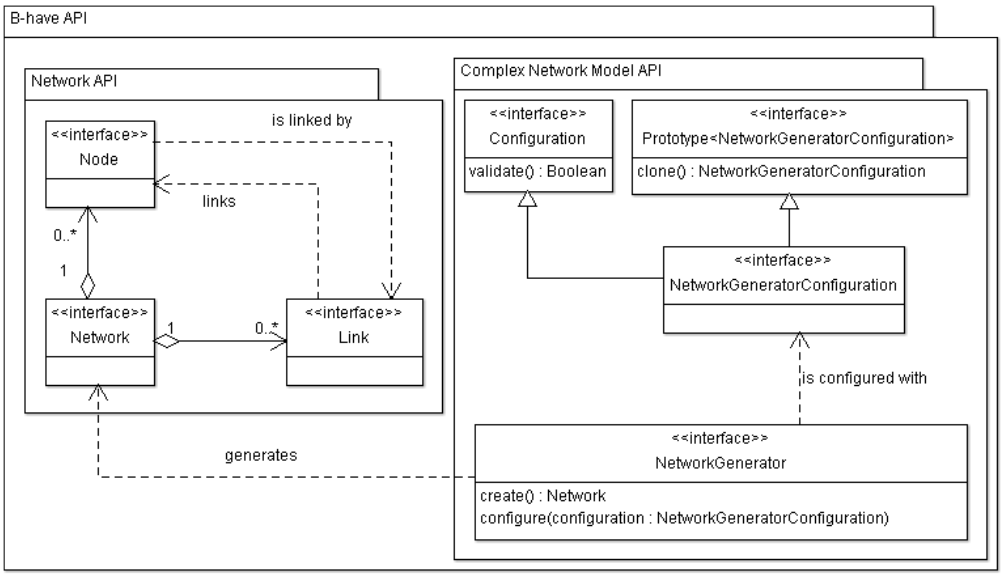


Figure 3.5: UML diagram for the *b-have workbench* network *API* basic interface structure.

Figure 3.5 presents a basic *UML* diagram for the interface structure of the developed *API*. This was designed so that the complex network models could be integrated in other simulation and modelling frameworks. In this thesis, we used the b-have workbench project in two ways. First, we used the existing application to pre-generate network models to be used to compare the prototyped model and the final model of experiments, "freezing"the network structures and focusing on the rest of the behaviour. This was done to ensure that the behaviour of the prototype was correctly reproduced in the final model. We then used the developed *API* to generate network on-the-fly and deploy them to our model in MASON.

[2]The developed *API* can be found at the *b-have* project website: http://bhaveproject.org

3.3 Distribution of Social Simulation Models in a Grid Environment

In the exploration of social simulation models we encounter a common problem which is deeply related to the analysis of the effects of different parameter combinations. The problem is that if a model parameter space is big enough, running simulations over that space is very demanding and takes a huge amount of time. The models are often executed on a single machine and the runs are executed sequentially. At most one can have a machine with a processor with multiple cores but the parallel execution of the simulations over the models is restricted to the number of the cores a processor has. In this section we show how one can eliminate such a problem by setting up a simple computer grid using multiple machines.

In this case, a grid is simply a set of loosely coupled networked computers acting together to perform very large tasks (Nabrzyski et al., 2004).

Considering a single machine as a processing unit, it is easy to see that more processing units can reduce the time necessary to run through the parameter space of a social simulation experiment. As an example, if one has ten processing units available in the grid, these can be used to process ten simulations over a model in the same time it would took to run a single simulation in a single processing unit, roughly in one tenth of the time. We say roughly because the performance gain is not linear (Sutter and Larus, 2005). This is specially true when we deal with grid systems as we will discuss later. The benefits of using a grid system are clearly expressed in (Walker, 2009).

When we have models that include a multiplicity of parameters, we want to explore them and analyse the results over the possible combinations of the parameter values. For simple models, the problem does not reside in the time consumed in the execution of a simulation run, but rather in the time consumed on the exploration of sometimes huge parameter spaces. Previously presented tools like NetLogo (Wilensky, 1999) allow for the execution of multiple runs in a single machine. Although this is an excellent tool for prototyping, it is limited by the number of cores a machine possesses. Other tools like MASON (Luke et al., 2005) are more efficient than NetLogo. In this case, MASON is optimised for running in a single thread efficiently, using one core of a machine (we can however run multiple simulations in multiple threads just like in NetLogo). We consider the usage of MASON for our large scale exploratory simulations, as it provides a fast and simple way to construct social simulation models easy to distribute.

We sustain that simple agent-based social simulation models that normally are executed on a single machine can benefit from parallel grid computing. In this section we show how it is possible to reduce the time taken in the exploration of the

simulation parameter space, executing different parameter configurations in parallel using a computer grid. To implement this, we use the discrete-event agent-based framework MASON (Luke et al., 2005) and Java Parallel Processing Framework (JPPF) (Cohen, 2005). We chose this set-up as both platforms are written in Java, making them cross-platform and extremely easy to install and use.

3.3.1 Parallel exploration of the parameter space

In this section, we describe how we can explore parameter spaces using a computer grid. We start by presenting an informal overview over the parallel exploration process and then formalise the concepts presented. Finally we present a comprehensive empirical analysis of the performance gains one can get from the usage of a grid system to execute social simulation experiments.

Parallel exploration process overview

A grid of computers executes working units called Jobs. Jobs have multiple independent tasks that can be executed separately. The job tasks can then be executed in parallel, by assigning them to different machines in the grid for execution.

So, we have to create agent-based model instances as tasks and create jobs by coupling multiple model instances (tasks). The next step is to submit the jobs to the grid and wait for results. Sending a job to the grid will distribute the execution of the tasks (the model instances) across the available grid machines. All of this is made using the JPPF framework to be discussed later on.

We can explain the parallel exploration process informally as follows (see figure 3.6):

1. Identify the social simulation experiment parameter space P (set of parameters considered and the respective domain for each parameter);

2. Take the parameter space and divide it into c unique configurations in which a configuration is a set of parameter values (one value for each parameter);

3. Construct grid jobs with r tasks. Each task is a configured agent-based model in which the model parameter values are drawn from the parameter space configurations. We consider the same configuration for one job, r is then the number of runs to be executed for each parameter configuration;

4. Submit the jobs to the grid.

5. Collect the results of the different simulation runs. The grid should be considered as a black box where we submit jobs and collect results when these are available;

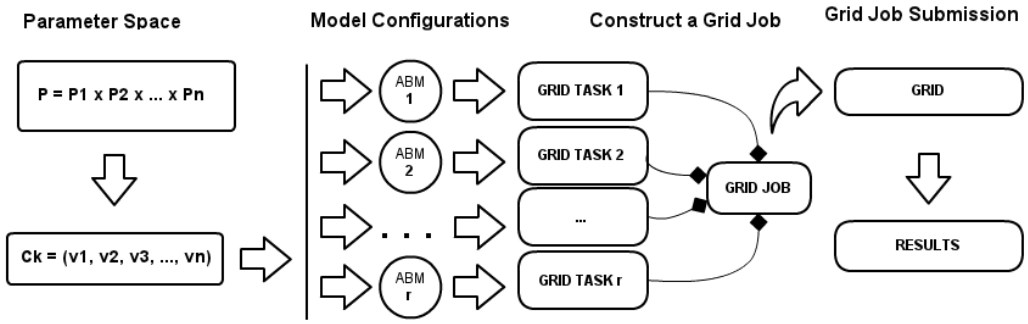


Figure 3.6: Parameter space parallel exploration process overview.

We propose the construction of grid jobs with r model instances with the same parameter configuration, where r is the number of runs we want to execute for each parameter configuration (as previously described in figure 3.6). This assures that we are coupling together model instances with the same expected execution time. The ideal number of jobs to be submitted at the same time to the grid depends on the machines available, there is no magic number for it as we can have an infinite number of possible grid configurations. We provide an empirical performance analysis for a small computer grid to make an assessment of the expected performance, the results can however be extrapolated for other grid configurations.

Formalising the parallel exploration process

We can now formally describe the simple parallelism process that can reduce the time taken in the exploration of social simulation models. One can construct an experiment and distribute its executions over different parameter configurations in the following manner. Consider an experiment $E = (M, P)$ where M is an executable social simulation model and P is the parameter space of the model. The parameter space P is of the form $P = P_1 \times P_2 \times \ldots \times P_n$ where P_k is the range of all possible values we want to consider for parameter k.

A model M behaves like a function $M : C \rightarrow M$, with $C \subseteq P$. M behaves like a function taking a set of parameter values which we will call configuration $C = (v_1, v_2, \ldots, v_n)$ where $v_k \in P_k$. The model execution over the configuration changes the state of this model which we can later analyse to extract results.

We define a grid task as a function $T_k : M \times C_k \rightarrow R_{T_k}$ where M is a simulation model and C_k is one of the $|P|$ possible configurations of the experiment's parameter space P. R_{T_k} is an entity representing the results produced by the task equivalent to executing the M over C_k.

A job is a work unit in a grid composed of tasks and it is defined as $J_k = \{T_{k1}, T_{k2}, \ldots, T_{kr}\}$. In this case a set of r tasks with the same configuration.

Consider now a grid $G = \{g_s, g_1, g_2, \ldots, g_m\}$ where m is the number of machines in that grid and g_s represents the grid server. We submit a job J_k to a grid server g_s, the grid server automatically decomposes J_k in its elementary r tasks (one task for each run over the parameter configuration C_k) and distributes each task T_k to each available machine $g \in G$ with $g \neq g_s$, balancing the workload between the different machines.

A grid job submission can then be denoted as $J_{\phi k} : J_k \times g_s \rightarrow R_k$ where J_k is the job being submitted to the grid, g_s is a grid server that receives the job submission and R_k is the set of results produced by executing the J_k job tasks over the same configuration C_k. In summary a job execution is the computation of r runs over the same parameter configuration. A job execution is done by submitting it to the grid, waiting for the task executions and collecting the results.

An experiment E is then distributed by creating a set of jobs J where $|J| = |P|$, being $|P|$ the number of configurations present in the parameter space P. From the total set of jobs J we create a set of $|J|$ job submissions J_ϕ and execute them to get our social simulation experiment results.

Performance Gains

The gains in performance one can get from using multiple processing units to execute instructions in parallel depend on the structure of the system. We can however have a general idea of how this process works.

To help us understand the general concept of parallel computation considering multiple processing units, we can refer to *Amdahl's Law* (Hill and Marty, 2008). Simply put, *Amdahl's Law* states that if you enhance a fraction of code f by a speed-up S, the overall speedup (or performance gain in terms of speed) is:

$$Speedup_{enhanced}(f, S) = \frac{1}{(1-f) + \frac{f}{S}} \tag{3.1}$$

Note that f is the portion of your code that can be executed in parallel and S is the speedup ratio analogous to the number of processing units available to distribute the code execution.

This law has also important corollaries that state that:

- When f is small, optimisations will have little effect.
- As S approaches infinity, speedup is bound by $1/(1-f)$.

When talking about a computer grid, the concept of speedup enhancement is similar but we have to take into account that the processing units are not centralised in the same machine but rather distributed over multiple heterogeneous machines and connected to the grid server by the means of an existing computer network. This configuration introduces communication overheads proportional to the number of machines contained within the grid.

Adding multiple machines would improve the time it takes to deploy a complete social simulation experiment exploration (as we are adding more processing units to the system) but adding more machines not only speeds up the exploration of a social simulation parameter space, but also adds more communication overhead to the system.

The speedup ratio we get from *Amdahl's Law* is an empirical measure of parallel performance. This can be described more generally as:

$$Speedup_{enhanced}(S) = \frac{\Theta_{E1}}{\Theta_{ES}} \tag{3.2}$$

where Θ_{E1} is the time it takes to run an entire experiment on a single processing unit and Θ_{ES} is the time it takes to run an entire experiment on S processing units. We reduce the time required to execute an entire experiment by running our tasks in parallel, distributing them across the S processing units.

As an example, consider a simple experiment with a parameter space consisting of exactly one configuration C_i (which is executed r times). With one parameter configuration, we perform a job submission $J_{\phi i}$ which submits a job J_i with r tasks. If we consider a task as the most basic unit that can be executed in parallel, we can say that our experiment can be totally executed in parallel. With $S = r$ we can execute every task concurrently. We can define our grid as $G = \{g_s, g_1, g_2, \ldots, g_r\}$

where g_s is the grid server and g_k is a grid node with $1 \leq k \leq r$. Given a job submission, the job leaves the grid when all de tasks are executed. Moreover, the time it takes to complete a job in the grid is equivalent to the maximum execution time of the tasks within that job (Epema et al., 2006). To calculate the speedup from the usage of a grid we then instantiate the terms from equation 3.2 as:

$$\Theta_{E1} = \sum_{k=1}^{r} \Theta_{T_{ik}}, \forall T_{ik} \in J_i \tag{3.3}$$

$$\Theta_{ES} = max(\Theta_{T_{ik}}), \forall T_{ik} \in J_i \tag{3.4}$$

where Θ_{E1} is the time we need to execute all the r tasks in a single processor, Θ_{ES} is the time it takes to execute all the tasks in the grid with S processing units available and $\Theta_{T_{ik}}$ is the time it takes to execute the task T_{ik} in a single processing unit.

The expression is not yet complete as we have to take into account the network communication overheads (as previously described). The overheads considered are:

- the job submission from the client g_c to the grid server g_s (denoted as $L_{g_c g_s}$);
- the task delivery from the grid server g_s to a grid node g_n ($L_{g_s g_n}$);
- the task result delivery from a grid node g_n to the server g_s ($L_{g_n g_s}$);
- the result delivery from the server g_s to the client g_c ($L_{g_s g_c}$).

We can now rewrite the previous term Θ_{ES} accordingly as:

$$\Theta_{ES} = L_{g_c g_s} + \Delta L + L_{g_s g_c} \tag{3.5}$$

with

$$\Delta L = max(L_{g_s g_k} + \Theta_{T_{ik}} + L_{g_k g_s}) \tag{3.6}$$

where $\Theta_{T_{ik}}$ is the time it takes to execute the task T_{ik} at the grid node g_k. Note that each task is executed at exactly one grid node.

Substituting the terms in equation 3.2 we get the speedup enhancement expression for the execution of a single grid job J_i with r tasks and r processing units available:

$$Speedup_{enhanced}(r) = \frac{\sum_{k=1}^{r} \Theta_{T_{ik}}}{L_{g_c g_s} + \Delta L + L_{g_s g_c}}, \forall T_{ik} \in J_i \tag{3.7}$$

In summary, the limits for the speedup gains in a given grid are closely related to the maximum time of execution of each model instance (which may not be constant) plus the communication overheads of job submission, task distribution and result collection. To have a better idea on how to analyse the performance of a computer grid, the reader should refer to (Epema et al., 2006).

Grid Performance Test

To help on the visualisation of the previously discussed grid performance gains, we perform a simple experiment. We consider the submission of a single job to the grid and vary the number of tasks within the job. We measure the job execution time at the client (this time includes the communication overheads) and observe how the number of tasks being executed at the grid affect the grid performance.

The experiment consists in creating "dummy" grid tasks that just wait 1000 milliseconds and then terminate. With all the tasks taking exactly one second to be executed, we submit a single job to the grid assigning an increasing number of tasks to this job. We use this to analyse the behaviour of the grid for a different numbers of tasks to be executed in parallel. Each job submission configuration is repeated for 50 independent runs.

In figure 3.7 we can see the average job execution time in the grid, versus the time it would take to execute all the job tasks sequentially. The grid used is composed of two 8-core and seven dual-core computers.

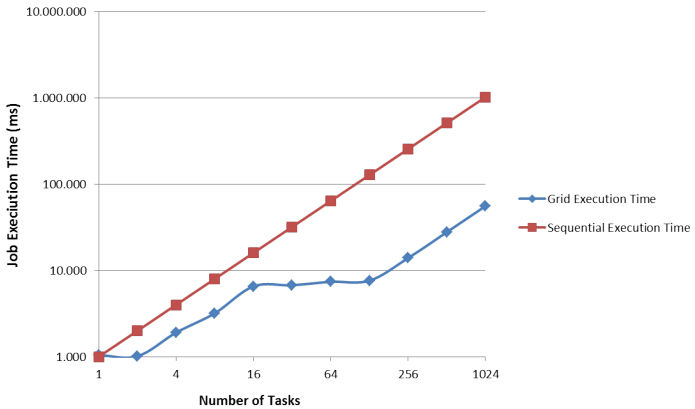


Figure 3.7: Average job execution time (in ms) for different numbers of tasks within this job. Each task takes exactly 1000 milliseconds to be executed. We measure the time it takes to execute the job in the grid (in blue) and also display the time it would take to execute all the tasks sequentially (in red).

As we can see in figure 3.7 and 3.8, if a job has few tasks, the execution time does not improve much. This is due to the load balance done by the grid server. The grid distributes the tasks in groups to avoid the excess of communication flooding the grid (which is particularly useful if this grid is a shared resource with multiple distinct clients submitting jobs at any given time). What this means is that the grid may choose for example to send a group of four tasks to a machine with only two processing units. As such, some tasks will be executed sequentially in this case. In figure 3.8 we can see the speedup ratio observed for this grid. The speedup is roughly optimal when we maintain 128 tasks in the grid. This experiment is

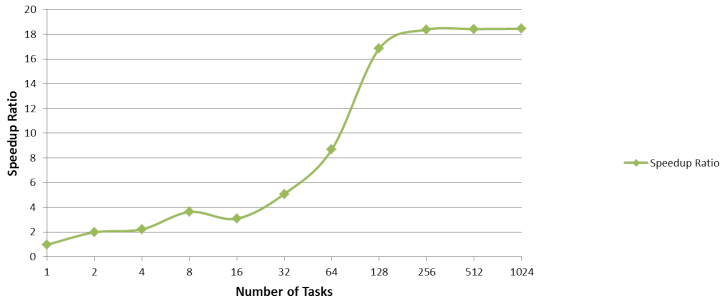


Figure 3.8: Speedup ratio observed for different numbers of tasks in a single job submission to a grid with 30 processing units. These processing units correspond to two 8-core and seven dual-core computers.

useful to observe the limitations of the grid. This basically dictates that for the submission of various grid jobs in parallel, with each job containing 50 runs for the same experiment configuration, the performance would not improve if we submit more than two / three jobs at the time (for the grid used in this example).

As we discussed previously, when dealing with the performance gains in a grid, the execution time of a job is bound by the maximum execution time of the tasks within that job. As the user has to perform multiple runs for each parameter configuration, the best approach is to organise the experiment with jobs that correspond to parameter configurations and tasks that represent runs of such configurations. Packing multiple runs for the same configuration in the same job is an elegant solution, as the expected task execution time is similar. Packing different configurations within the same job could cause situations like a group of very fast executing tasks being stalled by a long execution task. This is, tasks representing social simulation models that terminate very fast being stalled by other tasks containing a configuration that causes the enclosed model to take much more time to terminate its execution.

3.3.2 Combining MASON With JPPF

In this section we briefly describe the two technologies considered for the parallel execution of simulation runs. We also explain how to combine them. The resulting platform was the base for our final social simulation model. We used the work here discussed to conduct our experiments which will be discussed in the next chapter.

JPPF Overview

JPPF (Cohen, 2005), is an open-source, Java-based, framework for parallel computing. Basically it allows us to construct a grid with no effort. A grid is composed by one or more *grid servers* that handle job requests and manage the workload. Connected to those servers are the *grid nodes*. These are computers added to the system in a plug-and-play fashion. Finally, we have the *grid clients* which create and submit jobs to the grid servers. This framework provides facilities that enable us to deployment simple agent-based MASON models to be executed in parallel.

We focus on two basic elements: the first is a self-contained MASON agent-based model (by self-contained we mean that this model has everything that it needs to be executed anywhere on the grid once it is configured properly); the second is the JPPF grid client that allows us to submit a social simulation experiment to the grid.

Self-contained MASON Models

MASON is a multi-agent simulation toolkit designed to support large numbers of agents efficiently on a single machine (Luke et al., 2005). As MASON models are fully separated from visualisation, one can easily run a model without the graphical interface layer. MASON models are written in Java but with special attention to efficiency issues. This framework is elegant and simple enough to fit the purpose of this thesis: to show how one can use models that usually run in a single machine and submit multiple model instances to a grid.

MASON provides two essential building blocks for any model which is a *SimState* class that represents the discrete event simulator itself and a *Steppable interface* which we extend to create our agents (see figure 3.9). To create a self-contained simulation model we developed a simple *MASON* model by extending the referred building blocks and implement *Runnable* interface from *Java* (making the model suitable for execution in a thread, for instance) putting all the code necessary for the model to be executed in the "run" method. Finally, we want this model to be configurable prior to its deployment to the grid, so we create a method to accomplish that task and receive all the parameters necessary to the model prior to its execution. The basic *U*ML overview over the developed MASON model can be seen in figure 3.9.

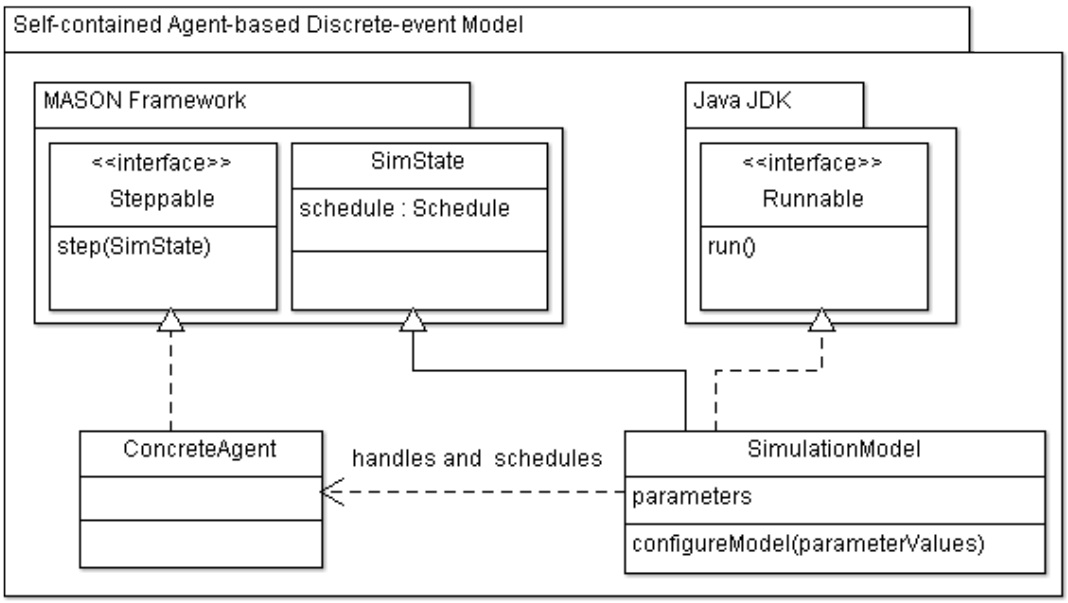


Figure 3.9: UML diagram describing the fundamental elements of the MASON framework. These elements are encapsulated in a *Java Runnable* to be submitted as a grid job, allowing the parallel exploration of social simulation parameter spaces.

Creating A Grid Client

To submit multiple jobs as described in the previous section we developed a JPPF grid client experiment runner. The experiment runner performs the following tasks:

1. scan through the parameter space;
2. create multiple MASON model instances with the various parameter values;
3. assign the model instances to grid tasks;
4. construct grid jobs containing those tasks;
5. submit the jobs to the grid;
6. collect the results;

Figure 3.10 shows the *UML* model for the developed grid client. This diagram depicts the fundamental elements for a JPPF grid client and how these are combined with the self-contained *MASON* model.

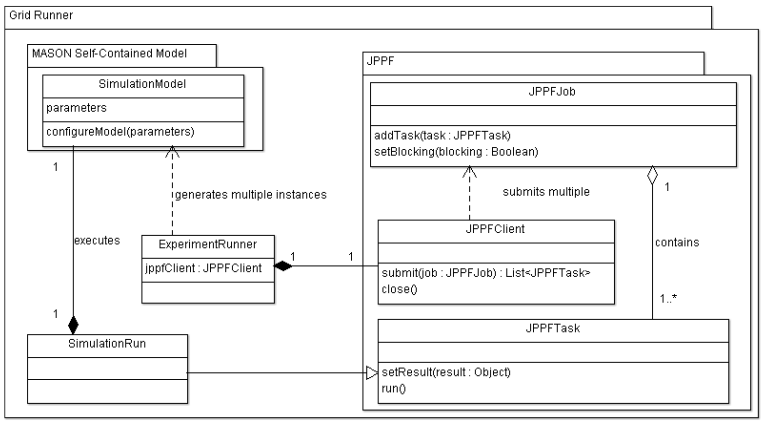


Figure 3.10: UML diagram depicting the fundamental elements that allow the creation of grid clients and how one can integrate the self-contained *MASON* model to submit simulation models as grid jobs.

3.3.3 Contributions

We have shown how one can use *MASON* and JPPF to take advantage of parallel computing technology to perform social simulation model parameter exploration. There is no requirement for advanced knowledge on parallel computing to easily implement a grid with the resources available. The approach we describe is perhaps not adequate for very complex agent-based models with need for scalability but it has proven advantages when dealing with simple exploration approaches.

We have described a simple approach to distribute social simulation experiments, executing tasks representing runs over the same model configuration in parallel. This method clusters tasks with similar expected execution times. This minimises the chances of creating jobs that enclose tasks with a high variance in the task execution time. Such a phenomenon would lead to jobs containing very fast tasks being stalled by long execution tasks.

The work presented in this section was selected for an oral presentation and published as a full paper (Nunes and Antunes, 2012b), after being reviewed by 3 anonymous referees.

Davide Nunes and Luis Antunes - "***Parallel Execution Of Social Simulation Models In A Grid Environment***", 13th International Workshop on Multi-Agent Based Simulation, MABS 2012

Chapter 4

A Model of Consensus by Segregation

As discussed in the previous chapters, in social simulation, not only is the structure of the social relations fundamental for the construction of plausible scenarios, but also the interaction processes are shaped by such structures. Each actor interacts in multiple social contexts located within multiple social relations that constitute their social space. In this chapter, we present a social simulation model to study the notion of *context segregation* (Nunes and Antunes, 2012a) by building on previous work about *context switching* (Antunes et al., 2009). In the developed model, the agents not only switch between social contexts, carrying with them their unique social identity, but also choose the contexts according to personal reasons. We apply the notion of context segregation to a simple game of consensus in which agents try to collectively achieve an essentially arbitrary consensus.

By introducing a segregation mechanism in the previous model, we explore the hypothesis that a society of agents converges to consensus more often and more rapidly due to local consensus group formation. The agents then avoid undesirable states of available social contexts. We conducted a set of experiments focused not only on the understanding of the segregation phenomenon, but also on the usage of different complex social networks to model abstract social relations and how they influence segregation.

The difference between context switching and context switching with segregation is in the change from one context to another. The segregation process introduces reasons to change whereas the context switching model (Antunes et al., 2009) only considers the probability of changing.

The chapter is organised as follows. In Section 4.1, we describe the methodology used to construct the social simulation model, modelling social contexts as social networks. We then describe the basic mechanism behind the context segregation model and its usefulness to social simulation. In Section 4.3, we briefly describe our model of experiments and what it is expected to occur. Finally, we analyse and

discuss the results on the context segregation model exploration and compare them to previous results of context switching.

4.1 Segregation Model

In (Antunes et al., 2009) it was suggested that the switching mechanism increases the speed of convergence to the global consensus, due to the formation of local consensus groups that, once formed, continually reinforce the cohesion of its members. Building on the previous model, we explore a mechanism designed from this hypothesis. Our goal is to analyse its validity. This is done by introducing a segregation mechanism (Schelling, 1971a, 1969; Nunes and Antunes, 2012a).

Whereas the previous context switching mechanism has only one parameter ζ_{C_i}, which is the probability of switching from a context C_i to another, the new mechanism encompasses a new parameter μ_{C_i}, which we call social tolerance of the context C_i. The social tolerance μ_{C_i} depicts the ratio of different opinions that a social agent tolerates in the context C_i. The mechanism can then be described in the following manner:

Let C_i be the social context in which a selected social agent is at the moment. Consider also μ_{C_i} as the social tolerance for the context C_i, and ζ_{C_i} the probability of switching from the context C_i to another selected relation.

1. Compute the ratio r of agents present in C_i with a choice opposite from the current agent's choice;

2. If $r > \mu_{C_i}$ (if the ratio is not within the social tolerance of the current context):

   switch to a randomly selected social context context C_j (see figure 4.1);

3. Otherwise

   switch to context C_j with a probability ζ_{C_i} (the probability of switching from the context C_i to the selected context C_j when the social tolerance requirements fail).

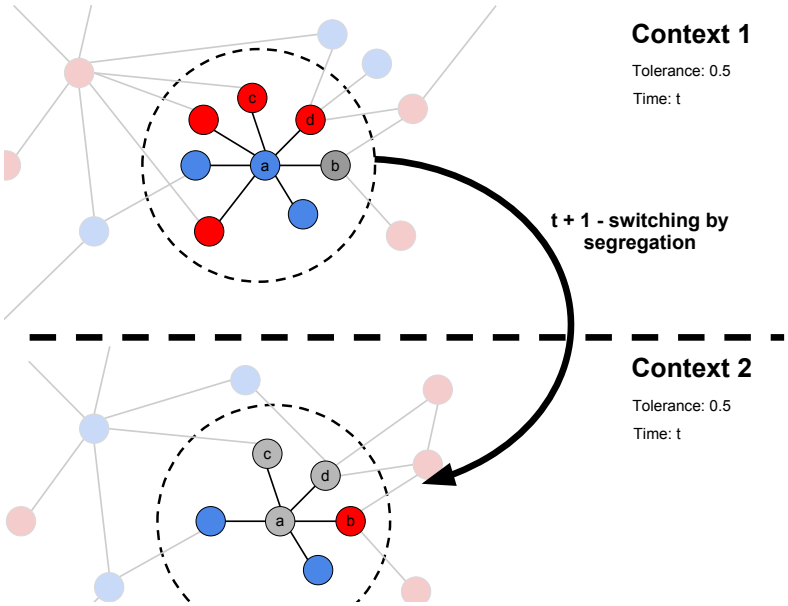


Figure 4.1: Example of context segregation. Agents inactive at a context are represented in grey. Agent choices are represented by other colours. At the end of the simulation iteration t, *agent a* (currently active at *context 1*) has to decide whether to switch context or not. The current context for *agent a* has a tolerance of $\mu_{C_1} = 0.5$. As the ratio of neighbours with an opposite choice is above the *tolerance threshold*, the agent will become active in *context 2* (in which the agent is currently inactive) at time $t + 1$.

In summary, low levels of tolerance in a social context make agents avoid that context if the neighbourhood is not in conformity with the agent's current choice. High values of tolerance promote the interaction with contexts even if the majority of the agents in a neighbourhood are not in agreement.

Context switching (Antunes et al., 2009) still plays a role in the dynamics of this model and contributes to the adjustment of the overall time spent in a social context. If an agent has a high tolerance level and chooses to stay in a neighbourhood, it may be still forced to switch out by the switching probability.

4.2 Model of Experiments

The experiments were developed in MASON (Luke et al., 2005) and executed in a grid environment described in (Nunes and Antunes, 2012b). Each experiment consists of 30 runs in which 300 agents interact until 3000 cycles pass, or total consensus is reached. In this set of experiments, our goal is to analyse the influence of the new parameter (the context tolerance μ_{C_i}) in the speed of convergence to global consensus.

We spanned the tolerance parameter between two contexts (μ_{C_i}, μ_{C_j}) from 0 to 1 in intervals of 0.05. We also vary the context switching parameter ζ_{C_i} between three values that were found to be interesting for the context switching mechanism (Antunes et al., 2009). Another source of variability is the network topology configuration for the various contexts.

In a first set of experiments, we focused on the usage of scale-free networks in two contexts, exploring the influence of the context tolerance parameter on the speed of convergence to global consensus. We then observed how different context switching probabilities affect the outcome of these experiments.

In a second series of experiments, we used different network topologies (scale-free and regular networks) in the construction of our scenario. We then explored not only the previously referred parameter variability, but also the consequences of the interplay between these distinct network shapes in the achievement of consensus.

We then analyse a set of experiments designed to understand the switching dynamics throughout a simulation run. We consider fixed tolerance values in an optimal consensus convergence region. We track down the number of agents that switch contexts using the switching probability, the context tolerance and the agents that do not switch and observe the switching trends over the course of the simulation.

Finally we vary the number of contexts agents have available and observe how it affects the convergence speed. We use this to compare the results with our previous work on context switching (Antunes et al., 2009).

The next section focuses on the details of the results.

4.3 Results and Discussion

In this section, we show how different values for context tolerance affect the speed of convergence to consensus. We explore the interplay between the switching and segregation mechanisms and the influence of different network topologies in the achievement of consensus. We also observe the influence of adding multiple social contexts to our model, observing its behaviour and comparing the results with previous work on context switching.

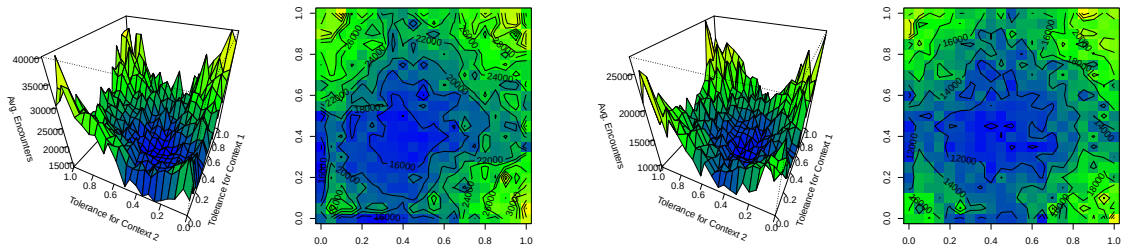
4.3.1 Context Tolerance Analysis

We now analyse and discuss the results on how different values for context tolerance affect the model behaviour. The next analysis also tries to show how the new tolerance parameter reacts to different values of switching probability (the parameter from the previous model (Antunes et al., 2009)).

Homogeneous social contexts with scale-free networks

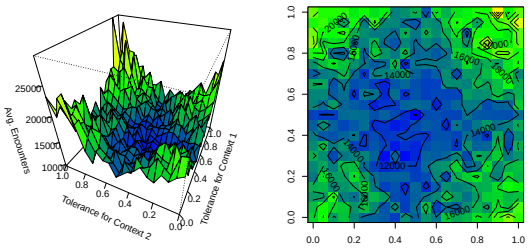
In our first set of experiments, we focus on a scenario with two social contexts, each one with a scale-free network. Figure 4.2 depicts the landscape for the span of tolerance parameter μ_{C_i} for both social contexts, maintaining homogeneous values of the switching probability ζ_{C_i} in both contexts ($\zeta_{C_i} = \{0.25, 0.5, 0.75\}$). The landscape represents the average number of encounters necessary to achieve global consensus (100%).

Focusing on the values of tolerance for which the number of encounters is smaller, we can extract right away a domain for which the tolerance parameter leads the speed of convergence to be faster. This happens for $\mu_{C_i} \in [0.2, 0.6]$.



(a) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.25, 0.25)$

(b) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.5, 0.5)$



(c) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.75, 0.75)$

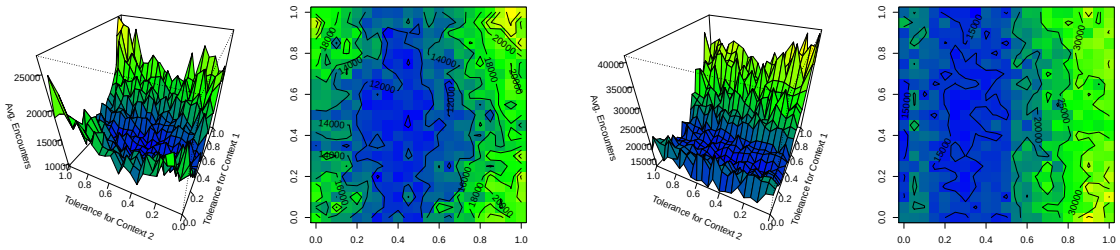
Figure 4.2: Average number of encounters to achieve global consensus with a context tolerance (μ_C) span of [0..1] for two social contexts with scale-free networks and homogeneous context switching values for each social context ($\zeta_{C_1} = \zeta_{C_2}$).

Another interesting result is that for smaller context switching values (for example $(\zeta_{C_1}, \zeta_{C_2}) = (0.25, 0.25)$), although globally the convergence is slowed down, the described values of moderate tolerance maintain an area of faster convergence. Even when the tolerance is 0 in one context, the convergence to consensus is still fast if the other context maintains a value of tolerance between 0.2 and 0.6. The context

tolerance is actually preserving the speed of convergence from the high instability caused by high context switching ratios.

Comparing these results with a model with only the context switching, we can see that the context segregation model actually leads the society of agents to consensus faster. We can see this by observing the number of encounters for the maximum tolerance in both contexts. According the segregation mechanism defined in section 4.1, when the tolerance is at its maximum value, only the switching mechanism is used, as "bad neighbourhoods" are not considered to exist.

In the next set of experiments we vary the context switching parameter and make it heterogeneous between contexts. Basically we adopt a low value of switching for one context and a high value of switching for the second. If agents only considered the context switching mechanism this would imply that they would switch from the context with the low switching probability less frequently (spending more time in that context) and spending less time in the context with a high switching value. Figure 4.3 presents the results for the described heterogeneous configuration. We



(a) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.25, 0.5)$

(b) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.25, 0.75)$

Figure 4.3: Average number of encounters to achieve global consensus with a context tolerance (μ_C) span of [0..1] for two social contexts with scale-free networks and non homogeneous context switching values. One of the social contexts has a lower value of switching than the other, we only present this configuration for one context because the results are symmetrical in this case.

obverse from figure 4.3 that maintaining a moderate tolerance value is extremely important in the context from which the agents switch less frequently. In this case if an agents spends more time in context C_1 and switches more frequently from context C_2, the convergence to local consensus achieved in C_1 can be delayed if they choose to stay there with an adverse neighbourhood. This happens because switching probability ensures that an agent spends more time at that context. In figure 4.8(a), we also see that very low tolerance values in context C_1 do not improve

the speed of convergence if this has a moderate switching ratio.

It is interesting to observe that on the limit of the heterogeneity ($(\zeta_{C_1}, \zeta_{C_2}) = (0.25, 0.75)$) (see figure 4.3(b)) the lowest values of tolerance for context C_1 also help to maintain optimal speed of convergence. This is coherent with our explanation for the phenomenon. As the agents switch more frequently from context 2 due to the switching probability ($\zeta_{C_2} = 0.75$), they establish local consensus groups within context 1 as long as the tolerance value for that context remains proximately within $\mu_{C_i} \leq 0.6$.

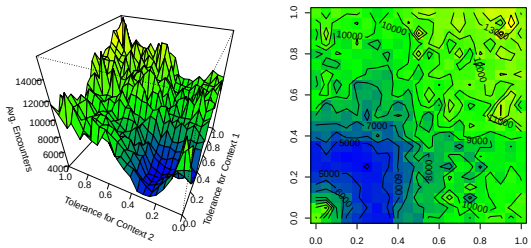
Homogeneous social contexts with regular networks

In another set of experiments, we analysed the usage of *k-regular networks* and how these affect the segregation mechanism. Although k-regular networks are not very good representations of real social networks, they provide a good baseline for the evaluation of our model of multiple social contexts. We think this is the case because the structure is the same for all the nodes in the network. In these networks. each node has the same number of connections ($2k$) which creates an highly clustered entity. This structure allows us to analyse the contribution of neighbourhood size for the segregation mechanism.

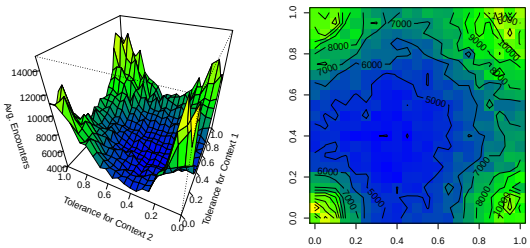
We first observe the influence of switching in the segregation tolerance response surface (see figure 4.4).

In figure 4.4 we see that for small values of k (in this case, each one of the 300 agents having 20 neighbours), the values of moderate context tolerance ($\mu_C \in [0.2, 0.4]$) is fundamental for achieving faster consensus with low values of switching probability $\zeta_C = 0.25$. When we raise the switching probabilities, the optimal tolerance region spans from 0.2 to 0.6 similarly to what happened in the homogeneous scale-free networks previously discussed. The number of encounters necessary to reach global consensus in the optimal tolerance region also raises with the switching probability but not drastically, this was also seen in the previous results for the scale-free networks 4.2. These results indicate that the segregation mechanism actuates with similar outcomes independently of the topologies being used. Scale-free networks require more encounters for consensus as they do not possess the high clustering features of these regular structures.

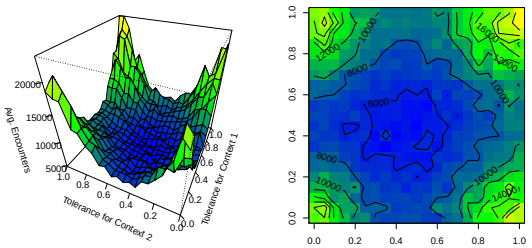
Our next experiment consisted in the observation of what would happen when bigger neighbourhoods are considered. In this case, figure 4.5 shows the results of such experiment. Here we can see that with larger neighbourhoods ($k = 30$ and $k = 50$) the tolerance parameter does not exert substantial impact in the convergence to consensus. It is although fundamental to keep the tolerance parameter roughly above the 0.2 threshold. This happens because large neighbourhoods present a structure in which low tolerances produce constant switching. This happens because



(a) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.25, 0.25)$ and $k = 10$



(b) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.5, 0.5)$ and $k = 10$

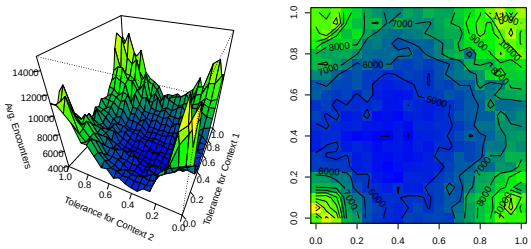


(c) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.75, 0.75)$ and $k = 10$

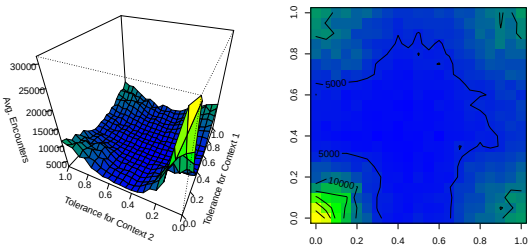
Figure 4.4: Average number of encounters to achieve global consensus with a context tolerance (μ_C) span of [0..1] for two social contexts with k-regular networks having a small k value and uniform context switching values for each social context ($\zeta_{C_1} = \zeta_{C_2}$).

there is an high probability of an agent encountering a neighbourhood in which more than 20% of its neighbours have different choices at the beginning of a simulation. This does not help the creation of local consensus at an early stage of a simulation as agents are constantly switching. This phenomenon was observed throughout all our experiments with homogeneous contexts (same topologies) allowing us to create two insights. The first is that under certain conditions, there is a minimum tolerance that agents must respect so that the segregation process is actually beneficial for the convergence to global consensus. The second, in consequence, being the fact that, for the optimal consensus regions, the segregation mechanism operates as a dominant switching strategy early in a simulation and, once the local consensus have been reached, agents switch less using this process.

Analysing the results of homogeneous *k-regular networks* with heterogeneous switching (see figure 4.6), this is, $\zeta_{C_1} \neq \zeta_{C_2}$, we can confirm a previously observed phenomena. In figure 4.6 we can see that similarly to what happened in the *scale-free networks* (see figure 4.3), to ensure faster convergence to consensus, we must ensure

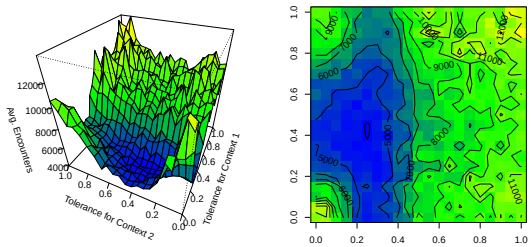


(a) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.50, 0.50)$ and $k = 10$

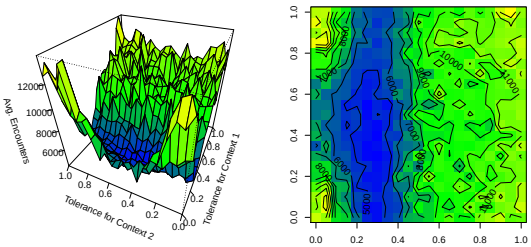


(b) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.50, 0.50)$ and $k = 30$

Figure 4.5: Average number of encounters to achieve global consensus with a context tolerance (μ_C) span of [0..1] for two social contexts with regular networks and uniform context switching values for each social context ($\zeta_{C_1} = \zeta_{C_2} = 0.5$). In this experiment, we vary the value of k to observe the influence of bigger neighbourhoods in the segregation mechanism. The results for $k = 50$ are practically identical to the results for $k = 30$.



(a) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.25, 0.50)$ and $k = 10$



(b) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.25, 0.75)$ and $k = 10$

Figure 4.6: Average number of encounters to achieve global consensus with a context tolerance (μ_C) span of [0..1] for two social contexts with k-regular networks having a small k value and heterogeneous context switching values for each social context ($\zeta_{C_1} \neq \zeta_{C_2}$).

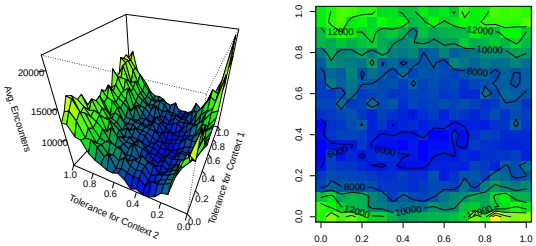
that the tolerance for the context from which the agent switches more frequently has moderate values.

In this case, the tolerance values, (like we observed in the previous experiment results depicted in figure 4.4) must be superior to a certain value (0.2) to allow for faster consensus. This is the aspect that differs from the scale-free topologies in what concerns to the segregation phenomenon. In scale free-networks, higher switching in one context makes that context tolerance not so relevant if there is another context

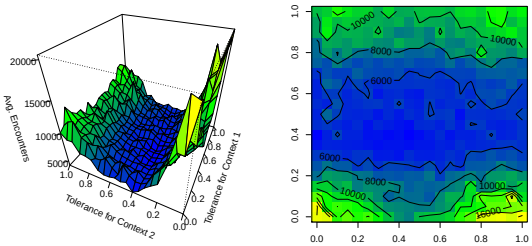
that allows agents to create the local consensus groups.

Heterogeneous social contexts

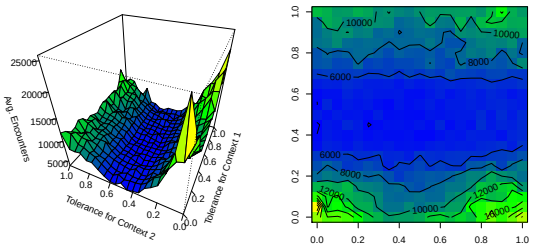
The second set of experiments involved a configuration of the two social contexts with heterogeneous networks, that is, each context has a different network topology. We focused on the interplay between k-regular networks and scale-free networks. As we previously exposed, regular networks offer an easy way to model highly clustered populations of agents. For this type of networks each node is connected with $2k$ other nodes, where k is a parameter of the generative procedure. We experimented with $k \in \{10, 30, 50\}$ being these reasonably small, medium, and large neighbourhood sizes.



(a) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.5, 0.5)$ and $k = 10$ for the k-regular network



(b) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.5, 0.5)$ and $k = 30$ for the k-regular network

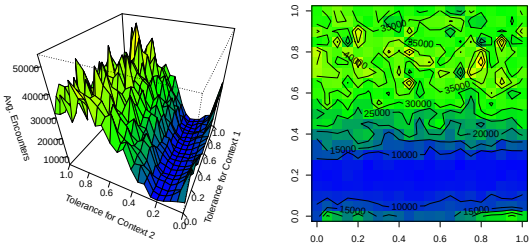


(c) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.5, 0.5)$ and $k = 50$ for the k-regular network

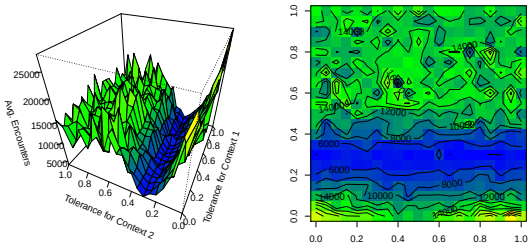
Figure 4.7: Average number of encounters to achieve global consensus with a context tolerance (μ_C) span of [0..1] for two social contexts, uniform context switching values $(\zeta_{C_1}, \zeta_{C_2}) = (0.5, 0.5)$ and heterogeneous networks (scale-free and regular, for context 1 and 2 respectively).

Figure 4.7 depicts the effect of having an heterogeneous setup for the social context networks. A social relation with a k-regular network displays interesting results when combined with a scale-free network. We see that it is worse to have high

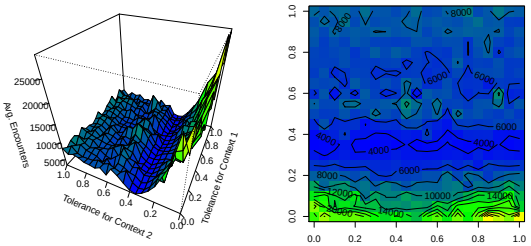
tolerance in regular networks for a small value of k than for a large one. Note that for smaller values of k, the neighbourhoods are smaller. Having higher tolerance in small sized social contexts implies that an agent has higher probability of interacting with a bad neighbourhood according to its personal choices in a given moment. For large values of k there is no need to have such a low tolerance value.



(a) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.25, 0.75)$ and $k = 10$ for the k-regular network



(b) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.25, 0.75)$ and $k = 30$ for the k-regular network



(c) Context switching with the values $(\zeta_{C_1}, \zeta_{C_2}) = (0.25, 0.75)$ and $k = 50$ for the k-regular network

Figure 4.8: Average number of encounters to achieve global consensus with a context tolerance (μ_C) span of [0..1] for two social contexts, heterogeneous context switching values $(\zeta_{C_1}, \zeta_{C_2}) = (0.25, 0.75)$ and heterogeneous networks (scale-free and regular, for context 1 and 2 respectively).

The last interesting results on the combination of scale-networks with k-regular networks are depicted in figure 4.8. The scale-free network has a context-switching value of 0.25 and the k-regular 0.75. This means that agents will switch less from the scale-free network and switch more frequently from the regular network.

We can see that low values of k promote the same effect we described in the previous figure 4.7, but the unbalance in the switching makes the phenomenon more evident. Another surprising result was that the landscape presented in figure 4.8 is very close to the shape of the landscape for homogeneous switching set to $(\zeta_{C_1}, \zeta_{C_2}) = (0.75, 0.75)$. What this means is that when combining a regular network with a scale-

free network, for medium or high levels of switching, very low to moderate values of tolerance are desirable to foster neighbourhood stability in the regular networks. For high levels of k, as we have seen before on figure 4.7, higher levels of tolerance also promote faster convergence.

Comparing the context segregation model with the context switching model (Antunes et al., 2009), for moderated values of tolerance, this model outperforms the former. We have also seen that the interplay between social network structures plays an important role in context dynamics, this is one of the key points discussed in the work of Antunes and colleagues (Antunes et al., 2008, 2009) but it is shown to be more evident in this exploration of the context segregation mechanism.

Although we only chose to analyse the context segregation model with scale-free and regular networks as the core structure for the abstract social relation structure, we have experimented with other topologies, also presenting interesting insights on the dynamics of this model and the influence of segregation processes in the achievement of global agreement. As an example, when using random networks (Erdős and Rényi, 1959) it is fundamental to model the social space with other concurrent social network topologies to avoid social isolation that occurs in this kinds of structures. Nevertheless, topologies and configurations that were not discussed in this thesis open up an interesting path for future work.

4.3.2 On Switching Dynamics

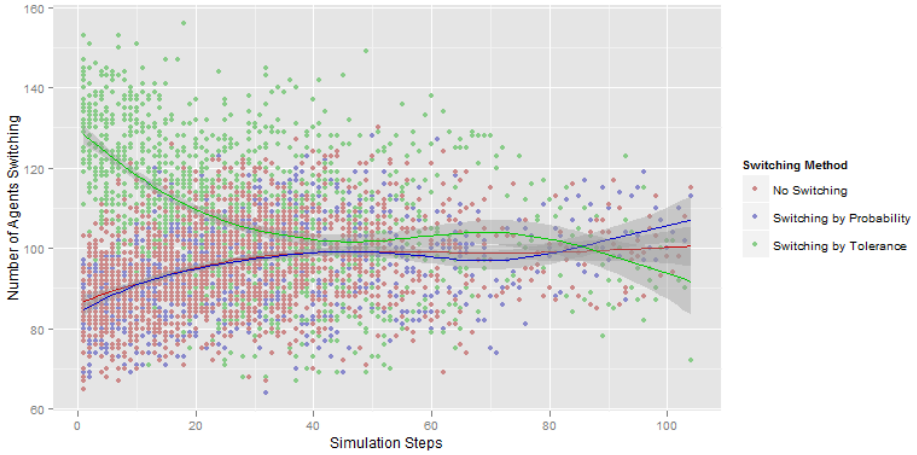
In this section, we analyse the results for our experiments regarding the switching mechanism trends throughout a simulation run. In these experiments we take a fixed context tolerance value for an optimal consensus convergence region (see figure 4.2). We then vary the switching probability within the values $\zeta_{C_i} = \{0.25, 0.5, 0.75\}$ and observe how this affects the switching mechanisms during the simulation. The objective is to observe how the segregation mechanism regulates the context switching mechanism in order to achieve faster consensus. We also want to analyse the interplay between the switching probability from our previous model of context switching and the segregation strategy throughout a simulation.

Switching Dynamics with scale-free networks

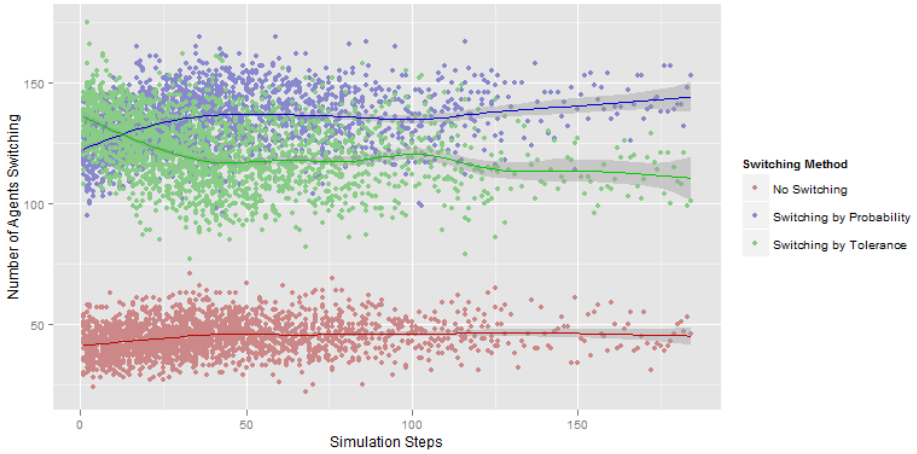
We experimented with two homogeneous scale-free networks, fixing the tolerance parameter in $(\mu_{C_1}, \mu_{C_2}) = (0.4, 0.4)$. This is one example of tolerance values within an optimal consensus convergence region as we can see in figure 4.2 from section 4.3.1.

In figure 4.9 we see that the segregation mechanism is more active at the beginning of the simulation while the neighbourhoods are not yet stable and is taken

(a) Switching probability $(\zeta_{C_1}, \zeta_{C_2}) = (0.25, 0.25)$



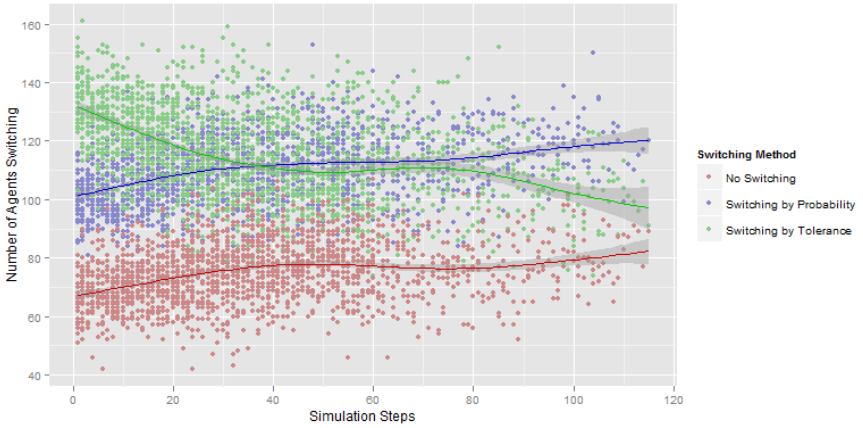
(b) Switching probability $(\zeta_{C_1}, \zeta_{C_2}) = (0.5, 0.5)$



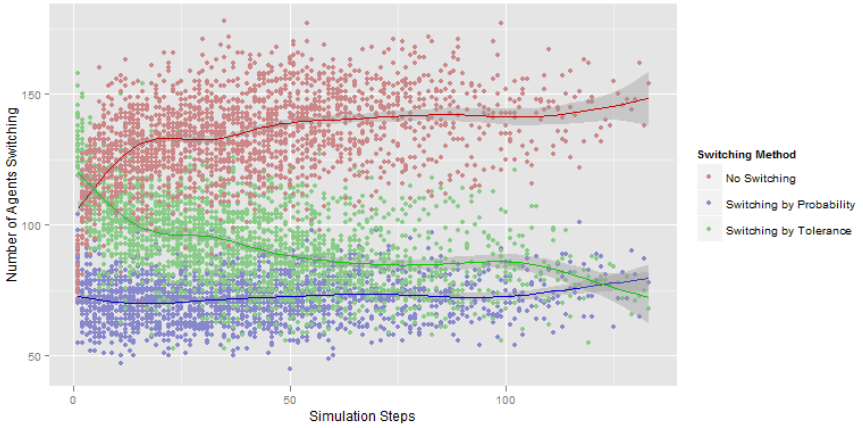
(c) Switching probability $(\zeta_{C_1}, \zeta_{C_2}) = (0.75, 0.75)$

Figure 4.9: Global switching trends depicting the number of agents choosing segregation, switching probability or no switching at the end of each interaction throughout a simulation run. We use scale-free networks in both social relation planes, homogeneous switching probability values and 300 agents interact until global consensus is reached. The tolerance parameter is fixed in $(\mu_{C_1}, \mu_{C_2}) = (0.4, 0.4)$.

over by the switching probability. The later acts as the main switching mechanism throughout a simulation. The global behaviour observed is that the number of agents that switch by segregation decreases more rapidly at the beginning of the simulation, gradually going down throughout the rest of the simulation. We can also see that the switching by probability mechanism is not disrupted by the segregation mechanism. The number of agents switching by probability is still strongly correlated to the switching probability parameter ζ_C.



(a) Switching probability $(\zeta_{C_1}, \zeta_{C_2}) = (0.75, 0.50)$



(b) Switching probability $(\zeta_{C_1}, \zeta_{C_2}) = (0.75, 0.25)$

Figure 4.10: Global switching trends for the context segregation model depicting the number of agents choosing segregation, switching probability or no switching at the end of each interaction throughout a simulation run. In this experiment we use scale-free networks in both social relation planes, heterogeneous switching probability values and 300 agents interact until global consensus is reached. The tolerance parameter is fixed in the optimal region $(\mu_{C_1}, \mu_{C_2}) = (0.4, 0.4)$.

The results for heterogeneous switching probability values (see figure 4.10) are intriguing. The switching by segregation trend is similar to the previous results (figure 4.9). For switching probability values of $(\zeta_{C_1}, \zeta_{C_2}) = (0.75, 0.50)$ (figure 4.10(a)) we see that the number of agents switching by probability and not switching resembles the trends we see in the previous experiment depicted in figures 4.9(b) and 4.9(c). With a greater disparity in the switching values ($(\zeta_{C_1}, \zeta_{C_2}) = (0.75, 0.25)$) we see that the agents choose not to switch as a dominant behaviour. A possible explanation is that the presence of a context from which the agents switch less frequently, in conjunction with the moderate tolerance value, creates the necessary conditions to foster an early convergence in that context. This results can be related to the discussion of figures 4.3, 4.8 and 4.6.

4.3.3 The influence of multiple social contexts

In this section we discuss a new experiment consisting on adding more social relation planes to the social space of our model. We use the results to make a comparison with our previous model of contexts switching (Antunes et al., 2009). We analyse the results for the *scale-free networks*. In the context switching model, these show the worst results in terms of speed of convergence. These models are especially interesting to us as most of the well known real-world complex network structures display scale-free properties (Barabási and Albert, 1999; Caci et al., 2011).

In this experiment, we consider the usage of multiple *scale-free* networks fixing the tolerance parameters within the optimal region (see figure 4.2). The tolerance parameter is fixed with the value $\mu_C = 0.4$. We vary the switching values homogeneously across the various contexts to compare our results with the context switching model.

Table 4.1: The influence of adding more social layers the achievement of consensus in terms of number of encounters necessary and using one optimal tolerance value ($\mu_C = 0.4$) observed in figure 4.2. In this experiment we used 100 agents to compare the results with the precious model of context switching.

	switching prob.					
	$\zeta_C = .25$		$\zeta_C = .5$		$\zeta_C = .75$	
num. planes	avg	st dev	avg	st dev	avg	st dev
1	-	-	-	-	-	-
2	3322	1657	2401	1159	2317	1276
3	3514	1722	2812	1557	2966	1845
4	3783	1730	3477	1622	3516	1912
5	4452	2302	4235	2213	5160	3011

As we can see in table 4.1, our initial conjecture that some tolerance values create an optimal zone for consensus achievement was confirmed. Adding more social relation planes to the social space makes so that we need more encounters to reach global consensus but this increment is not significant. Also, when we only have a single network in which the agents interact, consensus is never reached. This was also verified in our previous work (Antunes et al., 2008, 2009).

Table 4.2: Comparison of the number of encounters necessary to achieve consensus between the *context switching* (Antunes et al., 2009) and the *context segregation* (Nunes and Antunes, 2012a) models with *scale-free* networks. In these experiments, 100 agents interact to a maximum of 3000 cycles. The tolerance parameter for the segregation model is set to a value of $\mu_C = 0.4$.

		switching prob.					
		$\zeta_C = .25$		$\zeta_C = .5$		$\zeta_C = .75$	
num. planes	model	avg	st dev	avg	st dev	avg	st dev
1	segregation	-	-	-	-	-	-
	switching	-	-	-	-	-	-
2	segregation	3322	1657	2401	1159	2317	1276
	switching	10341	6386	5600	3844	4660	3547
3	segregation	3514	1722	2812	1557	2966	1845
	switching	15163	8666	8805	5785	7729	5604
4	segregation	3783	1730	3477	1622	3516	1912
	switching	18775	10807	13224	8192	14309	10901

The comparison between the results of simple context switching versus the model with the segregation mechanisms (see table 4.2) confirms some previously observed facts and present an interesting insight. In this case, with more social relation planes available, the segregation mechanism always presents better results than the simple switching mechanism. This is due to the fact that the switching is no longer uninformed and the agents switch according to their personal preferences. Switching from undesirable neighbourhoods, creating a segregation between multiple social contexts, promotes an early formation of local consensus groups.

We can see that adding more relational planes, not only does not interfere significantly in the convergence to consensus, but also presents another curious insight: the fact that the number of encounters observed for the segregation model (although this requires much less encounters) follow a similar trend to the results for the context switching model, meaning that the original switching mechanism influence is preserved.

4.4 Contributions

In this chapter, we showed that not only contexts are important for dissemination phenomena in structured societies, but also social segregation allows for convergence speed-up of such processes given that the right conditions are met. We built on previous work regarding context switching to construct our model of context segregation. We also presented a set of experiments that explored the segregation dynamics and its role to the achievement of a global consensus. The basis for our current work can be found on the following publication (Antunes et al., 2009):

Luís Antunes, Davide Nunes, Helder Coelho, João Balsa, Paulo Urbano – "***Context Switching versus Context Permeability in Multiple Social Networks***", 14th Portuguese Conference on Artificial Intelligence, EPIA 2009

In this previous work, we showed that context switching promotes faster convergences to consensus in comparison to context permeability (Antunes et al., 2008). The model of context segregation introduced in this dissertation shows even better results. Our mechanism of segregation actually regulates the context switching process and fosters early consensus group creation. This phenomenon helps to speed up the dissemination process, achieving a global consensus in less encounters.

Our results showed that social segregation mechanisms not only allows faster convergence to consensus, but also actuates independently from the different social structure configurations used in our experiments. Certain tolerance values can maintain the speed of convergence more or less independent both from the networks used and model configurations that included a larger number of concomitant social relations. Moreover, segregation also did not disrupted the existing mechanism of context switching by probability. We observed that the global switching trend for this mechanism is still highly correlated to the probability of changing from each context.

The segregation process here discussed considers that the tolerance values are fixed for each social context. In real world scenarios, this mechanism can evolve and the tolerance values can change over time. Even so, we experimented with static tolerance parameters to understand its influence in the segregation process and the achievement of consensus. The construction of a model that considers the evolution of tolerance contextualised with a real world scenario can be an interesting follow-up for future work.

The work presented in this section was selected for an oral presentation and published as a full paper (Nunes and Antunes, 2012a):

Davide Nunes and Luis Antunes - "***Consensus by segregation - the formation of local consensus within context switching dynamics***", 4th World Congress on Social Simulation, WCSS 2012, 2012

Chapter 5

Conclusions

In this final chapter, we conclude our dissertation by summarising our contributions, accomplished goals and our major findings.

We start by providing a more detailed summary of the contributions made by this dissertation, contextualising our findings and achievements. In the following sections, we reflect upon the possibilities for future work and consider the available research directions. Finally, we wrap-up and draw some conclusions about the work developed in this thesis.

5.1 Summary of Contributions

In this section, we summarise the contributions made by this dissertation.

5.1.1 Reviewing the state-of-the-art

This thesis provides an extensive review over simulation methods, agent-based modelling techniques and recent advances in social simulation that contemplate the usage of network structures to construct social space scenarios. We also provide an extensive review over complex network generative models and the theory behind such structures. To support the understanding of these models, we presented introductory notions on graph theory and social network analysis. These notions serve as the building blocks used to conceive and comprehend complex social network models. Our review also included previous work on multi-context models, introducing the notions of context permeability and context switching.

Although our review on social network models is very extensive, it still lacks a connection with real-world network systems. Our contribution was important for social simulation, as the majority of abstract simulation models are focused on the usage of network generation algorithms, but we still need a bridge that allows for the models to be instantiated with more concrete scenarios. This connection with

reality can help to understand and contextualise the abstract macro phenomena observed and avoids the conception of naive explanations for simulation results.

5.1.2 B-have project and simulation infrastructures

Another contribution was the continuous on the *b-have workbench* (Nunes and Antunes, 2011). The advance on this project was the implementation of an API suitable for social network model integration in Java-based social simulation environments. The network generated by the API, similarly to the workbench application, can generate social network structures. In this case, the network generation algorithms can be integrated in Java-based projects. We used this to include the generative algorithms in our MASON simulation model. This API also includes basic network export mechanisms, allowing the generated networks to be physically stored. The stored networks can then be imported by the API and restored to their JAVA object-based format.

Although this advancement was made in the b-have project, there is still one problem that needs to be addressed. Since the b-have project does not yet contemplate a fully fledged simulation platform, the integration of the generated components in existing platforms is a necessity. This integration is not yet provided out-of-the-box. The Java API provides a first effort in this direction, nevertheless, importing other network formats to existing applications should be a priority in the project and the development of libraries that allow this feature should be done in the future.

We then presented a social simulation infrastructure (Nunes and Antunes, 2012b) that allows for the usage of network generation algorithms, a multi-agent system simulation model and the distribution of experiments in a grid environment. This environment is described both formally and informally in what regards to its implementation. We combined the MASON simulation platform with our network generation API to create a simulation model. We then used the JPPF framework to distribute self-contained simulation models in a computer grid environment. This approach is particularly useful to the social simulation community as most of the experiments don't require the models to be distributed themselves but rather an environment where the simulation runs can be executed more efficiently. By efficiency we mean the parallel execution of experiment runs using the available resources, without putting much effort in the grid configuration itself. With this work, we extended the state-of-the-art by providing an implementation that can serve as a working example to create similar models and distribute the experiments using this technology. Our contribution was presented to the scientific community in (Nunes and Antunes, 2012b), after being peer-reviewed by 3 anonymous referees and presented at MABS 2012, a top-level workshop in this area.

5.1.3 Our model of multiple social relations with segregation

Finally, we contributed to the advancement of the state-of-the-art by providing a model of context segregation (Nunes and Antunes, 2012a) based on previous work on context switching (Antunes et al., 2009) and a set of experiments that explore the segregation dynamics and its role to the achievement of a global consensus. Our work provides some interesting insights not only on the new segregation mechanism but also on our previous work. Our results showed that not only do contexts play an important role in dissemination phenomena in structured societies, but also social segregation allows for convergence speed-up of such processes, given that the right conditions are met. Considering that *scale-free* networks seem to be present in a significant amount of real social relations, the mechanism of context switching with segregation with several concurrent networks can have a decisive role in enhancing the conditions for achieving global consensus. In the segregation process adopted, the tolerance values were fixed for each social context. In real world scenarios, this mechanism can evolve and the tolerance values can change over time. Nonetheless, this experiment configuration allowed us to understand the segregation influence in the achievement of consensus.

Considering the interaction behaviour of our agents, this is governed by a simple binary game of consensus in which agents follow the majority. We choose this simple game because we are interested in exploring the properties of our multi-relational model and the influence of different network topologies in dissemination processes and not really the interaction processes themselves. With this, we constructed and contextualised the basis for a consistent multi-context modelling approach that, although being currently treated as abstract, has the potential to be applied to more complex real-world social scenarios. Our results indicated that the segregation mechanism provided a regulatory process that speeds up the convergence to consensus but a clear explanation for this result is still very difficult to find. The question we should ask in future, is how dependent is the segregation mechanism from the consensus game used in our experiments. Answering this question can help on the contextualisation of our work within real-word scenarios and sociological theories that can support it.

Regarding the network models used in our simulation, one thing that could be improved is the fact that the network typologies are taken as fixed structures. Social systems are generally open and agents are free to join and leave the system. One thing that should be observed is the relationship between network properties and the dynamic processes that make usage of such structures. This could be achieved by using models with tunable properties such as clustering coefficient, transitivity, etc. We found that our segregation mechanism was relatively adaptive to differ-

ent network configurations, but observing this adaptability with a fine grain can be fundamental to understand how the mechanism behaves for different network structures.

The idea that segregation can foster faster auto-organisation can be tracked down from Schellings's work on residential segregation (Schelling, 1969, 1971b). We found that this remains valid in our model of multiple social relations. We were also curious to find that there are some conditions for segregation that not only help to foster faster consensus, but also actuate as a control mechanism transversal to different social structure configurations. With this we mean that under the right tolerance values for each social context, segregation actually helps to maintain the speed of convergence more or less independent from the network topologies used. The convergence speed was also independent from the social space dimension (the number of social relation planes) considered.

This work was presented in a paper (see (Nunes and Antunes, 2012a)) which was also peer-reviewed by 3 anonymous referees and presented in WCSS 2012, the most important world conference in social simulation.

5.2 Future Work

In this section, we discuss the possible future work for the subjects presented in this thesis. We start by looking at future work regarding our revisions over social networks and social simulation models. We then overview the research directions for our model of multiple social relations. We then discuss the needs for simulation analysis tools and some ideas that could be pursued in this domain. Finally, we present a possible application for some notions discussed on this thesis. Specifically, the usage of consensus games and knowledge about social network structures to construct a multi-agent system capable of extracting contextualised ontologies from Web documents.

5.2.1 Revision over social network models and social simulation

Our extensive review over network models opens up the possibility for a re-work of the presented topics into two different state-of-the-art reports. The first report being a complete revision over social network models. This revision should including not only the generative models presented in this dissertation, but also the work related to their application. The second report can be focused on the current state of social simulation, extending the presented work taking into account real-world scenarios where these model can be applied. These reports are a natural follow-up of the work

here presented as we noticed that the literature around these topics is very scarce, dispersed and incomplete.

In a similar way, our work on social simulation deployment can be further extended to create a tutorial on grid computing for multi-agent social simulation. This requires the analysis of different simulation and grid platforms, especially adequate for social simulation with working examples for each configuration. Also, in this future work, one thing that should be made clear is what kind of distribution can be made in a social simulation model and what are the different types of models according to this target question.

In this dissertation our agents are homogeneously configured and do not possess cognitive capabilities. This was the case because we were interested in the network structure properties and the auto-organisation properties of the system rather than complex agent architectures and decision making processes. Nevertheless cognitive agents still pose as a problem in social simulation as it is very difficult to create scalable models that can simulate artificial societies with a large number of agents.

As for the b-have workbench, future work should be focused on following the design direction presented in (Nunes and Antunes, 2011). As this tool was created in a modular fashion, the objective is to continue to add some key components such as more network generation algorithms from the literature and component export formats. An interesting possibility would be the creation of a Web service for network generation on demand with the possibility of network model instance storage and indexation. This would promote the reproducibility of experiments using networks generated by this service.

5.2.2 Our multi-relational approach to context permeability

For the new model of segregation presented in chapter 4, future work includes the analysis of the evolution of local consensus groups throughout an experiment, while tracking down the dynamics of the creation of these groups. To do this, we will investigate the usage of different segregation measures designed to be applied to social network structures, similarly to what is done in model presented in (Fagiolo et al., 2007). This ongoing work also focuses on the analysis of different network topologies and the comparison of experimental results with phenomena taking place in real network structures. As we create an understanding of multi-context models, an interesting line of research would be the composition of the different context mechanisms into one, more complex model. This would include the notions of: context permeability, switching and segregation (and other possible context dynamics) and consolidate these notions in a single model, comparing it with a real-world case study.

Our model of multiple social relations (Antunes et al., 2007, 2009; Nunes and

Antunes, 2012a) should be the consolidation from a sociological theory point of view. Although still very abstract, the present work on this multi-relation modelling methodology should be contextualised within the existing research on social identity (Ellemers et al., 2002), multiple concomitant social relations, and the embedding of actors in these structures. Moreover, we should focus on the influence of this context permeability phenomenon in the perception a social actor forms of its social space (Roccas and Brewer, 2002) and its importance for other interaction macro processes such as cooperation (or collusion) (Spagnolo, 1999).

Another interesting idea to further explore our model is the investigation of the relationship between abstract interaction spaces and the overlaid network of social relations. The interaction between these components has been suggested in (Nunes and Antunes, 2011) but has not been further developed. This line of research can also contribute to the completion of another objective related to the b-have project, the creation of reusable abstract environment models. The relation between social interaction and physical space is explored in architecture. Different designs for an environment can influence how people interact with each other through the social ordering of space (Shah and Kesan, 2007).

5.2.3 Social simulation analysis tools and methodologies

Analysing a social simulation experiment is often not a trivial task. The need for analysis tools is a current reality in the social simulation community. Designing appropriate experiments for abstract simulation models becomes extremely difficult, especially because there are no tools specifically designed to analyse these experiments. Experiment design and result analysis processes thus become dependent on the experience of the modeller. One idea that follows from our model of multiple social relations is that the formal conceptual representation of our multiplex social network structures can represent a leap forward in the construction of the needed tools.

We propose the creation of an ontology (Gruber, 1993) for multiple social network scenarios. This promotes clear understanding over our social space model and can be easily connected to real world scenarios. Moreover, the development of such ontology can benefit both the semantic web movement and the conception of tools to analyse social simulation models that make use of such network structures.

An idea that follows is the creation of a simulation ontology and its usage to make inferences about simulation experiments. This structure should be able to capture the temporal dimension of a simulation run and help on the discovery of relationships between simulated events. Ontologies are already applied to describe agent-based models in social simulation (see (Livet et al., 2010)) but their operationalisation can be taken further. These ontologies could be embedded in the simulation platforms

themselves and instantiated with experiment data. The next step would be to capitalise these structures to construct simulation automated analysis tools.

Finally, a subject that was not discussed in this dissertation was the clear definition of way the experiments are designed. In this case, our simulation experiments were designed around the exploration of some key parameters of our model. The way we design the experiments was based on a clear methodology (see Antunes et al. (2007)). The idea is to explore the design of agents, interactions, environments, institutions and societies by using initially simple notions and then increasingly deepening the simulations in terms of complexity, dynamism, and grounding in substantiated facts. The experiment design itself was informally exposed and this can make the task of experiment reproduction slightly less intuitive. In future work, the formal design of experiments using approaches like Lorscheid et al. (2012). We can consider another line of work consisting on the development of tools to support this approach and integrate it with existing simulation platforms.

5.2.4 Applications

Another promising line of research lies on the application of multi-agent systems and consensus games to domains like the semantic Web movement.

The semantic Web movement promotes the embedding of semantic content into Web documents. Unfortunately the Web entities do not find enough motivation to do their part in the construction of a semantically richer Web environment. This technology has however promising applications in domains like healthcare and life sciences which have been one of the major promoters behind it, driven by the need of common research vocabulary and knowledge management applications in healthcare.

A multi-agent adaptive application of context permeability and switching could be used to build and maintain rich contextual ontologies describing Web resources. Agent cooperation and social behaviour evolution are keystones for the creation of such ontologies, which will provide a basis for fostering auto-catalytic developments and adoption of semantic Web standards. The consensus formation mechanisms discussed in this dissertation could be capitalised to engineer an efficient solution to the ontology creation problem, as well as maintenance and updating. Consensus games and segregation regulatory mechanisms could be applied to create a decentralised auto-organised system capable of creating small-contextualised ontologies from sets of documents from which concepts can be extracted.

This system could also capitalise from the fact that web documents are organised according to scale-free network structures Barabási and Albert (1999). This could be an interesting starting point to study the relationship between the structure of Web document hyper-links and conceptual content present in those documents.

5.3 Conclusion

In this dissertation we made a leap forward in social simulation and modelling state-of-the-art. We uncovered the dynamics of segregation processes in our model of multiple social contexts. This process can be seen as an abstraction to strategic social neighbourhood selection. The segregation mechanism allowed us to validate an hypothesis created in our previous work on context switching. Models of context switching introduce a temporal component in the interaction processes that allows global consensus to be reached faster and more often. This is possible due to the creation of local consensus groups that create reinforced social contexts. These cohesive context structures are at the basis of faster dissemination in structured societies. In fact, social segregation mechanisms can not only allow faster convergence to consensus, but also actuate independently from the different social structure configurations used in our experiments. Certain tolerance values can maintain the speed of convergence more or less independent both from the networks used and model configurations that included a larger number of concomitant social relations.

Our findings create research directions that range from the understanding of social decision-making processes to promising applications in real-world scenarios involving auto-organisation of dissemination processes.

We still need to conceive an explanation for the regulatory process created by segregation in the light of sociological theories in order to create an instantiation of our models with more concrete scenarios. Regarding the concept of permeability between social contexts, this needs to be tracked down in real scenarios and carefully categorised. We are just scratching the surface of multi-relational models in social simulation with the next step being the connection of our model to the reality. This will allow our results to be validated and explained.

Bibliography

Allan, R. (2009). Survey of agent based modelling and simulation tools. *Computational Science and Engineering Department, STFC Daresbury Laboratory, Warrington WA4 4AD*, (October).

Antunes, L., Balsa, J., Urbano, P., and Coelho, H. (2008). Exploring context permeability in multiple social networks. In *Proceedings of the World Congress on Social Simulation*.

Antunes, L., Coelho, H., Balsa, J., and Respício, A. (2007). E*plore v.0: Principia for strategic exploration of social simulation experiments design space. Kyoto.

Antunes, L., Nunes, D., Coelho, H., Balsa, J. a., and Urbano, P. (2009). Context Switching versus Context Permeability in Multiple Social Networks. In *Progress in Artificial Intelligence*, volume 5816 of *Lecture Notes in Computer Science*, pages 547–559.

Axelrod, R. (1997a). Advancing the art of simulation in the social sciences. *Complexity*, 3(2):16–22.

Axelrod, R. (1997b). The Dissemination of Culture. *Journal of Conflict Resolution*, 41(2):203–226.

Barabási, A. and Albert, R. (1999). Emergence of scaling in random networks. *science*, 286(5439):509.

Bardón, M. (2009). *Mathematical Models and Simulations of Complex Social Systems*. PhD thesis.

Blau, P. M. (1977). A Macrosociological Theory of Social Structure '. *American Journal of Sociology*, 83(1):26–54.

Boguñá, M., Pastor-Satorras, R., Díaz-Guilera, A., and Arenas, A. (2004). Models of social networks based on social distance attachment. *Physical Review E*, 70(5):1–8.

Borgatti, S. P., Mehra, A., Brass, D. J., and Labianca, G. (2009). Network analysis in the social sciences. *Science (New York, N.Y.)*, 323(5916):892–5.

Caci, B., Cardaci, M., and Tabacchi, M. (2011). Facebook: topology to personality and back - an actor-based simulation.

Castelló, X., Eguíluz, V. M., Miguel, M. S., Loureiro-Porto, L., Toivonen, R., Saramäki, J., and Kaski, K. (2008). Modelling Language Competition: Bilingualism and Complex Social Networks. *The Evolution of Language - Proceedings of the 7th International Conference (EVOLANG7)*, pages 59–66.

Chen, X., Fu, F., and Wang, L. (2009). Social tolerance allows cooperation to prevail in an adaptive environment. *Physical Review E - Statistical, Nonlinear and Soft Matter Physics*, 80(5 Pt 1):051104.

Cohen, L. (2005). Java parallel processing framework.

Collier, N. (2003). Repast: An extensible framework for agent simulation. 36.

Conway, D. (2009). Structurally Induced Random Graph Model of Social Networks. *Retrieved April*, 20:2010.

Costa, L. D. F., Rodrigues, F. a., Travieso, G., and Villas Boas, P. R. (2007). Characterization of complex networks: A survey of measurements. *Advances in Physics*, 56(1):167–242.

Crooks, A. (2007). The repast simulation/modelling system for geospatial simulation. *Agent-Based Models for Spatial Systems in Social*, 44(0).

Davidsson, P., Holmgren, J., Kyhlbäck, H., Mengistu, D., and Persson, M. (2007). Applications of Agent Based Simulation. *MultiAgentBased Simulation VII*, 4442:15–27.

de Silva, E. and Stumpf, M. P. H. (2005). Complex networks and simple models in biology. *Journal of the Royal Society, Interface / the Royal Society*, 2(5):419–30.

Deng, K., Hu, K., and Tang, Y. (2011). Evolving network models under a dynamic growth rule. *Arxiv preprint arXiv:1108.1597*, pages 17–21.

Desjardins, M. E. G. M. (2005). Agent-organized networks for dynamic team formation. In *In Proceedings of clustering diameter the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*.

Diestel, R. (2006). *Graph Theory*. Springer, 3rd edition.

Dubiel, B. and Tsimhoni, O. (2005). Integrating agent based modeling into a discrete event simulation. In *Proceedings of the 37th conference on Winter simulation*, number 1, pages 1029–1037. Winter Simulation Conference.

Edmonds, B. (2003). Towards an ideal social simulation language. *Multiagentbased simulation II*, 2581:50–53.

Edmonds, B. (2010). Bootstrapping knowledge about social phenomena using simulation models. *Journal of artificial societies and social simulation*, 13(1):8.

El Sheikh, A., Al Ajeeli, A. T., and Abu-taieh, E. M. (2007). *Simulation and Modeling: Current Technologies and Applications*. IGI Publishing, Hershey, PA, USA.

Ellemers, N., Spears, R., and Doosje, B. (2002). Self and social identity*. *Annual Review of Psychology*, 53(1):161–186.

Epema, D. H. J., Franke, C., Iosup, A., Papaspirou, A., Schley, L., Song, B., and Yahyapour, R. (2006). On grid performance evaluation using synthetic workloads.

Epstein, J. (1999). Agent-based computational models and generative social science. *Generative Social Science: Studies in Agent-Based Computational Modeling*, 4(5):4–46.

Epstein, J. M. and Axtell, R. L. (1996). *Growing Artificial Societies: Social Science from the Bottom Up (Complex Adaptive Systems)*. The MIT Press, 1st printing edition.

Erdős, P. and Rényi, A. (1959). On random graphs. *Publicationes Mathematicae Debrecen*, 6:290–297.

Fagiolo, G., Valente, M., and Vriend, N. (2007). Segregation in networks. *Journal of Economic Behavior & Organization*, 64(3-4):316–336.

Fan, Z. (2005). A Fitness-Based Model for Complex Networks.

Fowler, J. H., Dawes, C. T., and Christakis, N. a. (2009). Model of genetic variation in human social networks. *Proceedings of the National Academy of Sciences of the United States of America*, 106(6):1720–4.

Gilbert, N. and Troitzsch, K. (2005). *Simulation for the Social Scientist*. Open University Press.

Girvan, M. and Newman, M. E. J. (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences of the United States of America*, 99(12):7821–6.

Goldspink, C. (2000). Modelling social systems as complex : Towards a social simulation. *Journal of Artificial Societies and Social Simulation*, 3(2):1–21.

Granovetter, M. S., American, T., May, N., and Granovetter, M. S. (1973). The Strength of Weak Ties. *The American Journal of Sociology*, 78(6):1360–1380.

Gruber, T. R. (1993). Toward principles for the design of ontologies used for knowledge sharing. In *Formal Ontology, Conceptual Analysis And Knowledge Representation*. Kluwer Academic Publishers.

Hamill, J. (2006). Analysis of layered social networks. Technical report, DTIC Document.

Herrera, C. and Zufiria, P. (2011). Generating scale-free networks with adjustable clustering coefficient via random walks. In *Network Science Workshop (NSW), 2011 IEEE*, pages 167–172. IEEE.

Hill, M. and Marty, M. (2008). Amdahl's law in the multicore era. *Computer*, 41(7):33 –38.

Hintze, A. and Adami, C. (2008). Evolution of complex modular biological networks. *PLoS computational biology*, 4(2):e23.

Jackson, M. and Rogers, B. (2007). Meeting strangers and friends of friends: How random are social networks? *The American economic review*, 97(3):890–915.

Jager, W. and Amblard, F. (2008). FreqNet : a new approach in formalizing social networks. *World Congress on Social Simulation 2008*.

Jennings, N. (2000). On agent-based software engineering. *Artificial intelligence*, 117(2):277–296.

Johnson, S. (1967). Hierarchical clustering schemes. *Psychometrika*, 32:241–254. 10.1007/BF02289588.

Kempe, D. and Kleinberg, J. Influential Nodes in a Diffusion Model for Social Networks. *Computer*.

Law, A. M. and Kelton, W. D. (2000). *Simulation Modeling and Analysis*. McGraw-Hill, New York, NY, third edition.

Leskovec, J., Chakrabarti, D., Kleinberg, J., Faloutsos, C., and Ghahramani, Z. (2010). Kronecker graphs: An approach to modeling networks. *The Journal of Machine Learning Research*, 11:985–1042.

Leskovec, J. and Faloutsos, C. (2007). Scalable modeling of real graphs using kronecker multiplication. In *Proceedings of the 24th international conference on Machine learning*, pages 497–504, New York, New York, USA. ACM.

Lewis, T. G. (2009). *Network science : theory and practice*. John Wiley & Sons, Hoboken, N.J.

Livet, P., Muller, J.-P., Phan, D., and Sanders, L. (2010). Ontology, a mediator for agent-based modeling in social science. *Journal of Artificial Societies and Social Simulation*, 13(1):3.

Lorscheid, I., Heine, B.-O., and Meyer, M. (2012). Opening the 'black box' of simulations: increased transparency and effective communication through the systematic design of experiments. *Comput. Math. Organ. Theory*, 18(1):22–62.

Luke, S., Cioffi-Revilla, C., Panait, L., Sullivan, K., and Balan, G. (2005). MASON: A Multiagent Simulation Environment. *Simulation*, 81(7):517–527.

Macal, C. M. and North, M. J. (2005). Tutorial on agent-based modeling and simulation. In *Proceedings of the 37th conference on Winter simulation*, WSC '05, pages 2–15. Winter Simulation Conference.

Macal, C. M. and North, M. J. (2008). Agent-based modeling and simulation: ABMS examples. In *Proceedings of the 40th Conference on Winter Simulation*, WSC '08, pages 101–112. Winter Simulation Conference.

Molloy, M. and Reed, B. (1995). A critical point for random graphs with a given degree sequence. *Random Structures & Algorithms*, 6(2-3):161–180.

Murata, T., Arikawa, H., Morishita, S.-i., and Maeda, T. (2007). A design of problem solving environments for policy making assistance using MAS-Based social simulation. In *e-Science and Grid Computing, IEEE International Conference on*, pages 521 –528.

Nabrzyski, J., Schopf, J. M., and Weglarz, J., editors (2004). *Grid resource management: state of the art and future trends*. Kluwer Academic Publishers, Norwell, MA, USA.

Nekovee, M., Moreno, Y., Bianconi, G., and Marsili, M. (2007). Theory of rumour spreading in complex social networks. *Physica A: Statistical Mechanics and its Applications*, 374(1):457–470.

Newman, M. (2003a). Properties of highly clustered networks. *Physical Review E*, 68(2):026121.

Newman, M. (2003b). The structure and function of complex networks. *SIAM review*, 45(2):167–256.

Newman, M. E. J., Watts, D. J., and Strogatz, S. H. (2002). Random graph models of social networks. *Proceedings of the National Academy of Sciences of the United States of America*, 99 Suppl 1(March):2566–72.

Nunes, D. and Antunes, L. (2011). Introducing the b-have workbench - creating reusable components for social simulation experiments. In *Proceedings of the 7th European Social Simulation Association Conference*.

Nunes, D. and Antunes, L. (2012a). Consensus by segregation - the formation of local consensus within context switching dynamics. *Proceedings of the 4th World Congress on Social Simulation, WCSS 2012*.

Nunes, D. and Antunes, L. (2012b). Parallel execution of social simulation models in a grid environment. In *Proceedings of the 13th International Workshop on Multi-Agent Based Simulation, MABS 2012*.

Pancs, R. and Vriend, N. J. (2007). Schelling's spatial proximity model of segregation revisited. *Journal of Public Economics*, 91(1-2):1–24.

Pidd, M. (2004). *Computer Simulation in Management Science*. Wiley, 5 edition.

Pitt, M. (2008). Simulation for strategic planning in health care - the state of the art. *Health Care*, (February).

Pujol, J., Flache, A., Delgado, J., and Sangüesa, R. (2005). How can social networks ever become complex? modelling the emergence of complex networks from local social exchanges. *Journal of Artificial Societies and Social Simulation*, 8(4):1–18.

Railsback, S. F., Lytinen, S. L., and Jackson, S. K. (2006). Agent-based Simulation Platforms: Review and Development Recommendations. *Simulation*, 82(9):609–623.

Ravasz, E. and Barabási, A. (2003). Hierarchical organization in complex networks. *Physical Review E*, 67(2):026112.

Roccas, S. and Brewer, M. B. (2002). Social identity complexity. *Personality and Social Psychology Review*, 6(2):88–106.

Rosvall, M. and Sneppen, K. (2007). Dynamics of Opinions and Social Structures. *Journal of Consumer Research*, physics.so:4.

Sansores, C. and Pavón, J. (2006). Agent-Based Modeling of Social Complex Systems. *Computer*, pages 99 – 102.

Schelling, T. C. (1969). Models of segregation. *The American Economic Review*, 59(2):pp. 488–493.

Schelling, T. C. (1971a). Dynamic models of segregation. *Journal of Mathematical Sociology*, 1(May):143–186.

Schelling, T. C. (1971b). On the ecology of micromotives. *The public interest*, 25(1971):61–98.

Schelling, T. C. (1978). *Micromotives and Macrobehavior*. Fels lectures on public policy analysis. Norton.

Schnettler, S. (2009). A structured overview of 50 years of small-world research. *Social Networks*, 31(3):165–178.

Schriber, T. and Brunner, D. (2006). Inside discrete-event simulation software: how it works and why it matters. In *Proceeding of the 2001 Winter Simulation Conference (Cat. No.01CH37304)*, pages 158–168. Ieee.

Shah, R. C. and Kesan, J. P. (2007). How architecture regulates. *Journal of architectural and planning research*, 24:350.

Simon, H. A. (1996). *The sciences of the artificial (3rd ed.)*. MIT Press, Cambridge, MA, USA.

Sloman, A. (1993). Prospects for ai as the general science of intelligence. In Sloman, A., Hogg, D., Humphreys, G., Ramsay, A., and Partridge, D., editors, *Prospects for Artificial Intelligence: Proc. of AISB-93*, pages 1–10. IOS Press, Amsterdam.

Sloman, A. (1994). Exploration in design space. In *ECAI*, pages 578–584.

Sokolowski, J. and Banks, C. (2010). *Modeling and simulation fundamentals: theoretical underpinnings and practical domains*. Wiley.

Spagnolo, G. (1999). Social Relations and Cooperation in Organizations. *Ssrn Electronic Journal*.

Steinhaeuser, K. and Chawla, N. V. Community Detection in a Large Real-World Social Network. *Methods*.

Suleiman, R., Troitzsch, K. G., and Gilbert, N. E. (2000). Tools and techniques for social science simulation.

Sutter, H. and Larus, J. (2005). Software and the concurrency revolution. *Queue*, 3:54–62.

Tassier, T. and Menczer, F. (2008). Social network structure, segregation, and equality in a labor market with referral hiring. *Journal of Economic Behavior & Organization*, 66(3-4):514–528.

Toivonen, R., Kovanen, L., Kivelä, M., Onnela, J.-P., Saramäki, J., and Kaski, K. (2009). A comparative study of social network models: Network evolution models and nodal attribute models. *Social Networks*, 31(4):240–254.

Travers, J. and Milgram, S. (1969). An experimental study of the small world problem. *Sociometry*, 32(4):pp. 425–443.

Troitzsch, K. G. (2009). Perspectives and challenges of agent-based simulation as a tool for economics and other social sciences. *International Conference on Autonomous Agents*, pages 35–42.

Tsvetovat, M. and Kathleen M. Carley (2005). Generation of realistic social network datasets for testing of analysis and simulation tools. *CMU ISRI Technical Reports Series*.

Uhrmacher, A. and Swartout, W. (2003). *Agent-oriented simulation*, pages 215–239. Kluwer Academic Publishers, Norwell, MA, USA.

Vazquez, F., González-Avella, J., Egúiluz, V., and Miguel, M. (2009). Collective Phenomena in Complex Social Networks. *Applications of Nonlinear Dynamics*, pages 189–199.

Wainer, Gabriel A. and Mosterman, P. (2010). *Discrete-event Modeling and Simulation - Theory and Applications*. Taylor and Francis.

Walker, L. (2009). Implementing a large scale social simulation using the New Zealand BeSTGRID computer network: A case study. *18th World IMACS / MODSIM Congress*, (July):1073–1079.

Wasserman, S. and Faust, K. (1994). *Social Network Analysis: Methods and Applications*, volume 24 of *Structural analysis in the social sciences*. Cambridge University Press.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442.

Weisbuch, G. (2004). Bounded confidence and social networks. *The European Physical Journal B-Condensed Matter and Complex Systems*, 38(2):339–343.

Wilensky, U. (1999). Netlogo.