Social networks and emergent behaviour: understanding group segregation in complex social networks

Dominic Aits
dla17@doc.ic.ac.uk

Supervisor: Alex Carver
Course: CO530, Imperial College London

September 2, 2018

Abstract

We study the emergence of group segregation in social network formation, where individuals choose to create or sever links in accordance with maximising their own private interests. By introducing the idea of an individual having a degree of 'tolerance' for others not of their own type, we can generate social networks that exhibit segregated behaviour by groups. We further show that this framework allows for intra-group segregation as well as inter-group segregation. Moreover, we design an algorithm for dynamic social network formation according to this framework. We find, through simulations, that group segregation is an endemic feature, and that as the cost of linking increases, networks that converge to a stable state begin to exhibit common characteristics with growing certainty.

Submitted in partial fulfillment of the requirements for the MSc degree in Computing Science of Imperial College London

Contents

1 Introduction

Networks are a recurring feature in many facets of life, ranging from societies to economics, and from biology to computer science. A significant aspect of real-life networks, especially social ones, is that they exhibit segregation - in other words, individuals who share common characteristics routinely are more closely connected and associated to one another relative to others who do not. Understanding the formation of group segregation in social networks is an important undertaking, namely for policy making.

There is an extensive literature in the economics, sociology and computer science disciplines regarding network formation. Analysing how social networks are formed has led to many schools of thought and approaches. One prominent field is that contributed by the Economics discipline. A game theoretic approach allows for a micro-founded network formation framework, where individuals (who are nodes in such a network) form and sever links according to their self-interests. In this approach, network formation is therefore a game among agents. Jackson and Wolinsky (1996) [22], in their seminal paper, introduce the 'connections model', that achieves just this, albeit limited to homogeneous agents in a static setting, and focused primarily on the stability and efficiency of such networks.

However, the literature on segregation in social networks is not as developed. In particular, there is limited literature that takes a game theoretic approach to network formation using connections between nodes with the aim in mind of generating group segregation in networks, despite a surprising need for such an analysis.

In real-life social networks, individuals themselves are responsible for creating and destroying their own links with others, in line with their own preferences (defined as 'types' henceforth). This paper therefore generalises the 'connections model' introduced by Jackson and Wolinksy (1996) [22] by introducing heterogeneity among individuals' types. Furthermore, our key contribution is the introduction of 'tolerance'. Two different agents, of the same type, may have different degrees of tolerance for other agents of other types. Thus, we introduce heterogeneity among agents of the same type. We are able to show that these two forms of heterogeneity are able to generate segregation among types (both inter-type and intra-type) during the formation of a network.

We moreover provide intuition for our results. There is a disparity between what is good for individuals (i.e. maximising their own utility), and what is good for society as a whole (i.e. maximising the cumulated welfare of everyone in the network). Often private interests do not coincide with the public, or social, interest. This distinction in the economics discipline is known as an externality, and serves as a key candidate for being able to generate segregation during network formation. We highlight this tension at play in social networks generated with two and three distinct types among agents.

Finally, we extend our analysis to a dynamic setting. We design an algorithmic procedure to induce social network formation in line with our theoretical framework. Simulations of generated networks, over a variety of parameter distributions, highlight that social network segregation is a common characteristic, even with a very low cost of linking. Statistical analysis over the generated network simulations yields surprising insights. We observe that most networks directly converge to a stable state without running into a cycle. However, there is a clear parabolic relationship between the cost of linking/connecting and the probability that a network will directly converge to a stable state. Moreover, we find that as the cost of linking increases, pairwise stable networks will share common statistical properties with increasing certainty. This means that while we might not be able to predict the actual structure of a social network ex-ante, we can more accurately predict its features (at high costs of linking).

The paper proceeds as follows. Section 2 is a literature review covering the developments in the field up until this paper. In Section 3, we introduce our theoretical framework, and in

Section 4, we showcase stable social networks generated from this theory in networks where agents have two and three distinct types. Section 5 introduces our algorithm for generating social networks according to our theoretical framework, and the details we took to implement this. The results from our simulations are discussed in Section 6. In Section 7, we discuss the social, legal and ethical considerations of our work. Finally, in Section 8, we conclude the paper.

2 Literature Review

In this section, we review the literature regarding network formation across multiple disciplines. We segment the literature into two broad thematic areas: 'micro-founded' and 'mechanical' network formation models, relating each to the literature that currently exists on segregation for each. Before that, we discuss the properties that a good model of network formation should be able to generate in Section 3.

2.1 Observations on Segregation in Social Networks

The sociology literature was one of the first disciplines to explore the role of segregation in social networks. Segregation, by definition, is the setting apart of groups of individuals according to some trait or structural reason. A key feature leading to segregation in social networks is the concept of homophily - that individuals with similar characteristics tend to bunch together (McPherson et al. (2001) [28]), and have a higher rate of contact than those who are more dissimilar.

The research into the literature regarding homophily, and its role in creating segregation in social networks is expansive. Lazarsfield and Merton (1954) [25] characterised two types of homophily: *status homophily* and *value homophily*. Status homophily accounts for sociodemographic traits that lead to divisions in society - such as race, ethnicity, sex, age, occupation - any traits that lead to a status being attributed to an individual, whether formal or informal. Value homophily, on the other hand, refers to traits specific to an individual, such as attitudes and beliefs. Both of these types can be intertwined and affect one another. The purpose of noting this is that segregation and homophily can occur across many dimensions and facets of life. A suitable model to explain segregation therefore has to be *agnostic* to the dimensions by which groups are formed. If not, then such a model is not generalisable in explaining human behaviour.

There has been a long empirical focus on measuring and identifying segregation across the literature. For example, in the United States, segregation by race and ethnicity has unfortunately persisted even into modern history. Kalmijn (1998) [24] finds that strong homophily via race and ethnicity occurs in marriage. Louch (2000) [26] notes a more surprising observation - that people are much more likely to report that their close friends are connected to one another if they are of the same ethnicity. Homophily across a different social dimension - gender - can occur from the earliest interactions between individuals. Eder and Hallinan (1978) [8] observe how children form and delete links in forming social networks at school. They focus on intransitivity - i.e. if A wants to connect to B and B wants to connect to C, what occurs between A and C. In their paper they find that boys are more likely to add an additional link to resolve the intransitivity between A and C while girls will delete the link with B. Homophily therefore appears to be an endemic feature of social networks - and if it occurs at the earliest stages of network formation - then a suitable model to explain it should be based on a set of simple connection-based rules to generate such behaviour.

A natural question is to then ask what other properties such networks tend to have overall. Girvan and Newman (2002) [15] and Gallo (2009) [14] identify five main properties that a majority of social networks exhibit:

1. *Short average distance:* the average distance between two nodes in the network is low.

2. *High clustering coefficient:* nodes that have a high level level of connections are more likely to be connected to one another

3. *Segregation patterns:* nodes tend to group together with few links between communities.

4. *Brokers:* there exist nodes who connect across communities.

5. *Hubs:* there exist nodes with a very high number of connections.

The first two characteristics combined lead to a small-worlds property that some networks tend to exhibit. The other three can co-exist alongside small-worlds, and may enhance their construction. A small-world network may sport segregated communities, which are hubs, connected by a few brokers. Goyal et al. (2006) [16] provide a real-world example of this in play. They analyse the networks formed by academic economists, and find that the social distances between economists between 1970 and 2000 has declined significantly despite a doubling in the numbers of economists. The discipline itself is a small-world network, formed by connections of inter-linked stars (i.e. using both hubs and brokers).

A useful model therefore is one that has the ability to generate a rich set of equilibrium networks, according for the five observations above. Such a theoretical framework will make strong use of homophily in network formation processes, thus leading to segregation.

In choosing the right approach, we consider the bifurcation in the modeling approaches Jackson (2005) [18] highlights. In particular there is a difference between mechanical (or stochastic) models and economic equilibrium models. The former focus on random graphs, where links are formed/severed through a stochastic process, or via some exogenous algorithm. They treat the formation process as a 'black box', and whilst providing useful information as to *how* social networks form, they are less useful in explaining *why*. Meanwhile economic equilibrium models focus on the micro-foundations of the formation process. Nodes make or destroy links according to maximising their own private interests. Whilst these models are excellent at explaining the *why* of social network formation, they are less adept at explaining the *how* - in particular, in providing the predictive power over what kinds of networks form and the characteristics that they have.

In this paper, we are interested in exploring why group segregation forms in social networks as they develop. For that purpose our modeling approach of choice is the economics equilibrium framework.

2.2 Micro-founded Models

In this section we survey the literature regarding models that explain network formation from individual actions (i.e. they are 'micro-founded') - as in the economic equilibrium approach. We make a key distinction between planar (space dependent models) and connections based models (links between nodes).

2.2.1 Connection-based Models

The literature on connections based models are expansive, so we survey the key contributions made so far. We define connections based models as those where relationships are captured by links between nodes. The links can represent any connection between two different agents according to any dimension of preference, whether that be economic, socio-demographic or otherwise. In that sense, they are ideal in capturing the breadth required from a model which can explain both status and value homophily.

The seminal paper in the space was provided by Jackson and Wolinksy (1996) [22]. In a static setting they provided conditions in an undirected graph framework with two-way benefit flows for networks to be stable. They introduced the equilibrium concept of *pairwise stability* - where two agents can form a link based on joint consent, but can unilaterally and individually delete the link if they so choose to. Whilst the analysis is limited to the case with homogeneous agents, it was useful in showcasing that under certain costs of linking, the resulting set of equilibrium networks can be restricted. The main focus of such a paper though was on the tension of stability and efficiency - in other words, the likelihood that the networks formed from private interests tend not to coincide with the network structures that are optimal for society as a whole.

From here, the literature developed in two different streams. The first is to generate variations of the original Jackson and Wolinksy (JW) model to account for more general behaviour, and to allow the equilibrium networks generated to better match those seen in real-life. That is, to generate models that better capture the properties of small-world structures, such as homophily. The second stream is more focused on designing network formation processes under different equilibrium concepts.

There have been many variations on the original JW model. Johnson and Gilles (2000) [23] introduce cost heterogeneity with a spatial cost topology, and find that the range of equilibrium networks generated are rich and diverse. Jackson and Rogers (2005) [20] expand the JW model to generate small-world properties. They cap the benefits that an individual may get from the wider network at a specific path length, and are able to generate communities in the network structure as a result. Unfortunately, such a modeling choice is arbitrary and not justified from a micro-foundation perspective, requiring the search for a better framework.

Our paper is probably most closely associated with Gallo (2009) [14]. In his formulation he is able to generate pairwise stable networks that exhibit segregation and small-world properties. However, his approach has a few drawbacks. His results are driven by cost heterogeneity and by information asymmetry. Gallo introduces heterogeneity into the *knowledge* of the network - some agents are fully aware of the whole structure of the network, whilst others are less knowledgeable. This is unappealing since segregation may not necessarily be a function of the information that an agent may have about the nature of the network. Our paper therefore is a noted improvement in this regard. Not only do all agents have perfect information regarding the nature of the network, but we are able to generate segregation by introducing heterogeneity into the *benefits* that an agent may get from his connections to others, as opposed to cost heterogeneity, which appears to be the common route taken in the literature.

Others have sought to expand the equilibrium concepts used to generate networks. A key paper in this stream is that provided by Bala and Goyal (2000) [1]. They opted for a more traditional Nash equilibrium approach in network formation, where agents choose the optimal strategy they can *given* the strategies of all other agents. An equilibrium occurs in this setting when no agent wants to deviate from their current strategy given the strategies of all others. This necessarily leads to a different type of network formation since agents can form and sever links independently from all others - no mutual consent is needed. Moreover, it is appropriate in directed graphs, where benefits flow in one direction from one node to another, as opposed to two-way flows as per the original JW model.

Other key papers in the literature include the following. Galeotti et al. (2006) [13] expand the original Bala and Goyal model from a homogeneous setting to one with heterogeneous players and characterise the set of equilibria when the costs of connections can vary between different types of agents. Celis and Mousavifar (2016) [6] generalise the original Bala and Goyal model to allow for network formation in bi-directed networks. Currarini et al. (2009) [7] focus on homophily, minorities and segregation using a traditional utility maximisation approach with steady state equilibria. Their main results focus primarily on how the size of groups interacts with the groups' relations in the broader network - larger groups, for example, tend to form more connections per capita, and more of those links are with individuals of the same types. In a separate paper, Jackson and Rogers (2007) [19] present a model where individuals form links in two ways - either at random, or by searching the local structure of the network around where they reside. In their approach they are able to generate networks that have short average distances and high clustering.

There has been a brief literature on dynamic network formation. Watts (1999) [33] was the first to expand the JW model to a dynamic setting and analyse potential network structures that can occur, with Jackson and Watts (2002) [21] providing a more theoretical paper afterwards. Again the focus was primarily on the tension between pairwise stability and efficiency. Unfortunately these papers tend to take a very theoretical view as opposed to a combinatorial/simulation based approach, which limits their ability to make predictions over the types of networks that may occur in parameter ranges where multiple equilibria are likely (from a given initial condition). In this sense, our paper makes a novel contribution since we generate multiple networks given our model with the aim of distilling overarching statistics about the nature of such equilibrium networks.

2.2.2 Planar Models

Planar models have traditionally been used for explaining segregation in networks, and we briefly touch upon them here. The seminal paper in this field is by Schelling (1969) [31]. The model, in its original formulation, took the form of a one-dimensional linear array. Agents may belong to one of two groups - "stars" ("+") and "zeros" ("0"). Each agent has a tolerance for the fraction of agents of the same type in their neighbourhood - defined as the four nearest cells on either side of the agent's position. If the actual fraction of agents of the same type to all agents is below the tolerance measure, the agent chooses to relocate (and in doing so, shifts all other agents' positions). The key contribution of the paper is the global segregation can occur rapidly in the case where the tolerance ratio of agents is 0.5.

In a follow up paper, Schelling (1971) [30] expands the model into a two-dimensional space. The key difference between this paper and the previous one is that empty cells are introduced for agents to move to, if that is their preference. Agents' neighbourhood ranges are now defined as the local 3×3 spaces around them. Moreover, agents retain the same tolerance measure as before - if they are dissatisfied with the fraction of agents of their own type, they move to the nearest empty cell that removes that dissatisfaction. Schelling finds that in this framework, if the tolerance fraction is below a third, a randomly initiated sequence of agents' positions leads to random outcomes - if it its higher, then the model converges to global segregation.

The literature since has grown rapidly, so we briefly touch upon major contributions to the field. Of particular interest is models where integration and segregation can jointly occur. Benenson and Hatna (2011) [4] (and later in 2015 [3]) were the first to showcase this in a Schelling framework by adjusting the size of the two groups. In particular, joint integration and segregation can occur at low tolerance threshold levels. Moreover, other variations of the model can also induce more varied segregation behaviour. Hatna and Benenson (2012) [5] show that similar results can occur when further heterogeneity is introduced into the model - in particular when each type of agent has its own tolerance threshold.

A natural limitation of the planar approach is that it limits the means by which agents can deviate from their current position if they are incentivised to do so. In Schelling (1971) [30], agents move to the nearest empty cell where the fraction of agents of the same type in their local neighbourhood is greater than their tolerance threshold. Agents are not permissible however to move beyond these confines. This makes an analysis of segregation for other socio-demographic dimensions, such as friendships and political beliefs, much more difficult to model as we constrain the means by which agents can deviate. A more general framework, where connections between agents matter as opposed to their position in some Euclidean space, would allow for a richer, more applicable, model.

2.3 Mechanical Models

Despite the paper being focused on connection based models, we briefly touch upon the alternative modeling approach - using mechanical and/or stochastic processes to generate networks. The main approach taken here is by using random graphs. Erdös and Rényi (1959 [9], 1960 [10], 1961 [11]) were some of the first to use a simple Bernoulli distribution for link formation - each link has a p probability that it belongs to a graph, and a $1 - p$ probability that it does not. Their key insight is in identifying a set of "phase" transitions. When p is small (less than $\frac{1}{n}$), the graph contains disjoint and relatively small components - when it is larger, we are more likely to get one large component.

Wasserman and Pattison (1996) [32] introduce a generalisation of Bernoulli random graphs, termed Markov graphs by Frank and Strauss (1986) [12]. In short, they allow for dependencies in the link formation process. In a pure Bernoulli process, links are formed independently of one another, which tends to lead to low clustering. Instead, these authors introduced conditional dependence between links - link ik may be more or less likely to form if links ij and ik are already present. Barabási and Albert (1999) [2] and Price (1976) [29] introduce a different variation via preferential attachment. In their framework, nodes are allocated new links depending on how many links they already have. This process yields degree distributions that are more in line with what is found in the empirical evidence.

Despite the success of replicating the structure of observed networks, a key failure in this framework is its inability to explain *why* links form. The link formation process is exogenously determined. Therefore, to adequately explain group segregation in social network formation, we need a framework that allows for micro-founded behaviour, in other words, connection-based models.

3 Theoretical Framework

In this section, we introduce our theoretical framework to generate segregation in social networks during network formation. This section is structured as follows. We first introduce some definitions which will be used throughout the remainder of the paper (Section 3.1). In Section 3.2 we introduce Jackson and Wolinsky's (1996) seminal model to establish a benchmark model where all agents are homogeneous with regard to their types. The choice of equilibrium concept in this paper is pairwise stability. Then, in Section 3.3, we introduce heterogeneity into the model, in the spirit of Galeotti et al. (2006) [13]. Here, we argue that this framework is able to generate group segregation, but that it holds a few unrealistic properties during a network formation process. Therefore in Section 3.4, we introduce our concept of 'tolerance'. This allows us to introduce heterogeneity within types of agents, and makes for a rich theoretical framework capable of generating both inter- and intra-type segregation.

3.1 Definitions

In this subsection we specify some definitions that we will be using throughout the specification of the theory. Let $\phi = \{1, ..., N\}$ represent a finite set of agents. Each agent represents a node on a graph. The edges between the nodes in the graph represents a connection between any two agents. The definitions are in line with Jackson and Wolinksy (1996) [22].

3.1.1 Graph Structure

As is customary in the literature, and following Jackson and Wolinsky (1996) [22], we define the complete graph as g^{ϕ}, to be the set of all subsets of ϕ, each with a size of 2. From this, we can define any possible graph on ϕ as belonging to the set $\{g \,|\, g \subset g^{\phi}\}$. To specify a connection between any two distinct nodes $i, j \in \phi$, we denote ij to represent the link between nodes i and j in the relevant subset of ϕ. Therefore, if $ij \in g$, then nodes i and j are directly connected; and if $ij \notin g$, then nodes i and j do not have a direct connection between them.

Notice that this formulation means that we are considering graphs where connections are not directed. The main focus in this paper is on joint confirmation of linking - that is, both individuals need to consent to form a connection between them. Either individual may choose to sever an existing link between them if they so choose to. In contrast, directed graphs, or di-graphs as they may be known, have uni-directional links, meaning that an individual may choose to form (or 'perceive' a link) to another without their express consent.

The above allows us to formalise the creation and destruction of links. Let $g + ij$ yield the graph generated by taking the original graph g, and adding the link ij. Similarly, let $g - ij$ yield the graph generated by taking the original graph g, and removing the link ij. In other words, $g + ij = g \cup \{ij\}$; and $g - ij = g \backslash \{ij\}$.

Let $N(g) = \{i \,|\, \exists\, j \;\; s.t. \;\; ij \in g\}$ represent the set of nodes which have at least one neighbour (i.e. direct connection), and $n(g) = |N(g)|$ to be the cardinality of this set. We can further define a path in this graph. For a graph g, and nodes i_1 and i_m, if there exists a set of distinct nodes $\{i_1, i_2, ..., i_m\} \subset N(g)$, and $\{i_1 i_2, i_2 i_3, ..., i_{m-1} i_m\} \subset g$, then there exists a path between i_1 and i_m in g.

Jackson and Wolinksy (1996) [22] also provide a definition of what constitutes a component in this graph, which we adopt for this paper as well. Let $g' \subset g$. If, for all $i \in N(g')$ and $j \in N(g')$ where $i \neq j$, there exists a path in g between i and j - and, for any $i \in N(g')$ and $j \in N(g')$, it is such that $ij \in g$ implies that $ij \in g'$ - then we can say that g' is a component of g.

To this paper, we also add a further definition that will be of use later in Section 5. We define the adjacency matrix of a graph g to be $adj(g)$ to represent a symmetric $|\phi| \times |\phi|$ matrix retaining information about all current links in g. Let g_{ij} represent a potential link between nodes i and j. If $g_{ij} = 1$, then $ij \in g$; otherwise if $g_{ij} = 0$, then $ij \notin g$. Each element of $adj(g)$ can be accessed by $adj(g)(ij)$ where $adj(g)(ij) = g_{ij}$. Naturally, $adj(g)(ii) = g_{ii} = 0$ for the diagonal elements.

3.1.2 Utility Functions

In strategic network formation, individuals choose to form or sever links between each other whilst maximising their own private interests. Hence, each individual needs to have a utility, or payoff function, ascribing the value that individuals gain from the current state of a graph g. As before, we follow in the spirit of Jackson and Wolinksy (1996) [22] when it comes to definitions.

We first define the *value* attributable to an *entire graph* to be $v : \{g \,|\, g \subset g^{\phi}\} \to \mathbb{R}$, and for V to be the set of all possible functions v. For example, in the true utilitarian sense, a suitable candidate for v may be $v(g) = \sum_i u_i(g)$ where $u_i : \{g \,|\, g \subset g^{\phi}\} \to \mathbb{R}$.

We further need to define an *allocation rule*, $U : \{g \,|\, g \subset g^{\phi}\} \to \mathbb{R}^{\mathbb{N}}$. This allocation rule determines how the value attributing to the whole graph is allocated to the individual agents (i.e. nodes) in the graph. Hence, the function $U_i(g, v)$ represents the utility/payoff an individual i receives from a graph g with value function v. For example, in the case above where $v(g) = \sum_i u_i(g)$, a suitable candidate for $U_i(g, v)$ is $U_i(g, v) = u_i(g)$ for all $i \in \phi$.

3.1.3 Pairwise Stability

Having defined the necessary components, we can now focus on identifying an equilibrium concept for network formation. This is a critical part of any analysis as it enables us to determine when any dynamic network formation process is to end. There are a variety of approaches that the literature has formed over the years, as eluded to in Section 2.2.1, with the two key candidates being *pairwise stability* and *Nash equilibrium*.

We opt for the equilibrium notion of pairwise stability, as per Jackson and Wolinsky (1996) [22]. Pairwise stability requires individual agents to jointly consent for a link to be formed between them, yet allows each individual to destroy any existing links they may have independently of anyone else. We believe that this is more indicative of how real-life social networks are formed. For example, creating an (honest) friendship between two individuals requires both of their consent - however, it is possible for either individual to sever such a relationship independently of the other if that individual so chooses to. The difference between pairwise stability and Nash equilibrium is in the formation of links. In Nash equilibrium, an agent chooses an optimal strategy *given* the strategies of all other agents - in other words, the agent acts non-cooperatively with other agents. Hence, agents may form links with others without the other agent's express consent. This may be appropriate in a directed graph setting, but not in an undirected one (i.e. two agents may alternate between creating and destroying links between them, hence never leading to a stable network).

We now formalise the notion of pairwise stability. A graph g is pairwise stable, given a value function v and an allocation rule U if two conditions hold:

$$\forall ij \in g, \;\; U_i(g, v) \geq U_i(g - ij, v) \quad and \quad U_j(g, v) \geq U_j(g - ij, v) \tag{1}$$

$$\forall ij \notin g, \;\; if \;\; U_i(g, v) < U_i(g + ij, v) \quad then \quad U_j(g, v) > U_j(g + ij, v) \tag{2}$$

Following Jackson and Wolinsky (1996) [22], we furthermore state that a graph g is *defeated* by g' if $g' = g - ij$ and (1) does not hold for ij under g, or if $g' = g + ij$ and (2) does not hold for ij under g.

Intuitively, if a link exists in a graph between two agents, then they must both benefit individually from such a link. The second condition states further that if agent is strictly better off adding a link to another, then this link can only not exist if the other agent is strictly worse off. In other words, if one agent is strictly better off with a link, and the other is weakly better off, then the link will be formed. Additionally, no two agents can have a link if either of them are strictly worse off since this violates condition (1).

An interesting case emerges if both agents are indifferent about maintaining a link or not. If $ij \in g$, and agents i and j are indifferent between maintaining the link or not, then condition (1) states that the link will be kept. If $ij \notin g$, then condition (2) is not violated either. Therefore, if two agents are indifferent between a link or not, then that link may, or may not, exist. For ease of simulation in Section 5, we stipulate an additional assumption:

Assumption 1. *If two distinct agents $i, j \in \phi$ are indifferent between $g_{ij} = 1$ and $g_{ij} = 0$, then form the link such that $g_{ij} = 1$.*

So now links are formed when both agents are weakly better off with the link than without it. Equivalently, we can re-state the definition of pairwise stability, weakening condition (2) to allow for this:

$$\forall ij \in g, \;\; U_i(g, v) \geq U_i(g - ij, v) \quad and \quad U_j(g, v) \geq U_j(g - ij, v) \tag{3}$$

$$\forall ij \notin g, \;\; if \;\; U_i(g, v) \leq U_i(g + ij, v) \quad then \quad U_j(g, v) > U_j(g + ij, v) \tag{4}$$

Notice that the definition of pairwise stability is a static concept. There is nothing in the definition that alludes to how networks are formed dynamically. It instead represents a 'snapshot' of the stability of a given network at any point in time, and is thus independent of any formation process that the network is following during its construction. Hence, the point of pairwise stability is mainly to determine whether a dynamic process should terminate - if no agent wants to sever a link, or if no pair of agents wants to create a new link, then the network is pairwise stable, and there is nothing to be gained from continuing a dynamic process.

There are a few further observations to be made about pairwise stability as an equilibrium concept. First, only single links are formed or severed at any one point in time. Pairwise stability does not allow for simultaneous link creation or destruction, either by a single agent, or through a coalition. Second, agents are myopic. This means that any given point in time, when considering a potential link to create or destroy, agents do not anticipate future potential actions by either themselves or by others. For example, if an agent were not myopic, he/she may opt to not form a particular link with another, expecting that then they might be excluded from joining a different component that would yield far more utility. In this regard pairwise stability is a simple and perhaps naive equilibrium concept, but one that is particularly effective in establishing a benchmark when considering social network formation behaviour.

3.2 A Homogeneous Connections Model

In this section, we present the connections model popularised by Jackson and Wolinksy (1996) [22]. The aim is to establish a benchmark restricted model with constraints that we can relax to generate richer results. Jackson and Wolinsky proposed a homogeneous agent model of network formation. Every agent has the same 'type', and therefore gains the same benefits from connecting to another, irrespective of who they are.

They established their model as one mimicking communications between agents. Individuals gain utility from directly connecting to others via link creation. In addition, they also gain benefits from indirect communication - from who their direct connections are linked to, and so on. These indirect benefits are decayed according to the shortest path length between any two agents. Moreover, there is a cost to forming links. Individuals therefore have to balance the benefits, both direct and indirect, against the cost of communication, when deciding on what actions to pursue.

A motivation of such a model is as follows. Social networks tend to be small worlds. Most individuals have a small group of 'close' friends (i.e. direct links), and wider circles of acquaintances, to whom an individual spends less time with, and whom they get 'access' to via their direct connections (and so on). In effect, an individual gains higher utility the closer they are connected to those they know. At the same time, there is a cost to forming friendships, either through the time value associated with forming the links, or through some other (perhaps pecuniary) cost. Such a view of network formation is in line with the connections model highlighted here.

We now proceed to formalise the model. We first define a *general* utility function for an agent i. Let $U_i(g)$ represent the utility that an agent i gains from graph g, and suppose further that u_{ij} denotes the utility that an agent i receives when directly connecting to agent j. Moreover, let δ be the decay rate at which the utility from an indirect connection between agents i and j is discounted by, raised by the geodesic distance (i.e. shortest path length) between these two agents, denoted by $dist(i, j)$. Naturally $0 < \delta < 1$. Finally, we introduce a cost of direct communication between any two agents i and j, denoted by c_{ij}. Therefore, the general utility function can be written as:

$$U_i(g) = \sum_{k \in \phi} \delta^{dist(i,k)} u_{ik} - \sum_{k:\ ik \in g} c_{ik} \tag{5}$$

Notice that equation (5) embodies both *value* and *cost* heterogeneity in its current formulation. u_{ik} can differ among different pairings of i and k, allowing for heterogeneity among the benefits that individuals can gain from connections, either direct or indirect. Similarly, c_{ik} can differ among pairings of i and k as well, representing that there may be different costs of connection for different individuals. Further, note that every individual receives a benefit u_{ii}, without any discount, irrespective of what connections they have. If there is no path between i and j, then $dist(i, j) \to \infty$, with the result that no benefit is attained to either node from a lack of such a path.

Jackson and Wolinsky then place a series of constraints on equation (5) to generate a homogeneous agent model. They assume symmetry, with $c_{ik} = c$ for all ij (hence cost homogeneity), and $u_{ik} = 1$ for all ij (hence value homogeneity).

They proceed to prove the following proposition for a characterisation of pairwise stable networks. Below we state the proposition, without proof, but with intuition below:

Proposition 1. *(Jackson and Wolinksy (1996) [22]). Assume a symmetric connections model with $U_i(g) = u_i(g)$:*

(i) A pairwise stable network has at most one (non-empty) component.

(ii) For $c < \delta - \delta^2$, the unique pairwise stable network is the complete graph, ϕ^N.

(iii) For $\delta - \delta^2 < c < \delta$, a star encompassing all players is pairwise stable, but not necessarily the unique pairwise stable graph.

(iv) For $\delta < c$, any pairwise stable network which is non-empty is such that each player has at least two links and thus is inefficient.

We provide some intuition to the result above. The first result, that any pairwise stable network has at most one non-empty component can be understood by the following. Consider a pairwise stable network g, which has two or more non-trivial components. Pick one component, say which has nodes i and j with $g_{ij} = 1$. Note the incremental gain to both of their utilities which the link ij yields them (by definition of a link, it must be weakly positive). Now consider a separate component, with nodes k and l and link kl. If we connect k to i, then not only does k gain a direct benefit from connecting to i, but it also gets an additional benefit, albeit with a value of δ^2, of an indirect connection to j through i. Hence, there can be at most one non-empty component.

For (ii), we can apply the result from (i). For $c < \delta - \delta^2$, any individual can choose to improve upon an indirect connection through directly linking to another agent. This yields a benefit of δ, at a cost of $c+\delta^2$, since the individual will, at worse, forego an indirect connection of δ^2 when substituting it for a direct connection.

For (iii), there are multiple equilibria, one of which is a star. No peripheral nodes at the edge of a star have an incentive to connect to another peripheral node. This is because the benefit of doing so (δ) is outweighed by the cost of $c + \delta^2$. Similarly, the centre node at the middle of the star has no incentive to delete links since $\delta > c$ (and same for periphery nodes connecting to the centre).

Finally, for (iv), it is intuitive to see that any pairwise stable network that is non-empty must contain nodes that have at least two links, since to have only one link would yield a negative payoff. In such a case, an individual could improve by deleting its single link and being a singleton. It is also true that the empty network here is also pairwise stable as no individual has an incentive to connect to any other.

Whilst the homogeneous model is powerful in the sense that it is capable of generating a relatively small subset of pairwise stable networks over large cost ranges, it is incapable of generating segregation. This is because all agents belong to the same type. Therefore, it is necessary to relax the constraints on the model, which we now proceed to do.

3.3 A Heterogeneous Connections Model

In generating segregation in social network formation, we need to introduce the idea that agents have a 'type' associated with their existence. This allows us to introduce heterogeneity into the model. There are two (non-exclusive) avenues to do this. The first is to introduce value heterogeneity into the model - in other words, that an agent of one type gains a different level of utility from a connection, either direct or indirect, to another agent of a different type than their own. The second is cost heterogeneity. Here, we can specify that an agent of one type, when connecting directly to an agent of a different type, is penalised differently than if they connected to an agent of the same type.

Introducing agent heterogeneity into a model of strategic network formation is not new. Galeotti et al. (2006) [13] introduced both value and cost heterogeneity into their models. However, there are some clear differences between their approach and ours. First, they characterise stable networks using Nash equilibrium as the stability concept of choice. Second, they do not conduct a general study of decay in the network. Rather, they focus primarily on networks where the path length does not matter in the magnitude of benefits. Third, they focus primarily on cost heterogeneity over value heterogeneity.

Other literature has also explored heterogeneity in social network formation. Johnson and Gilles (2000) [23] extend Jackson and Wolinksy's (1996) network formation model by introducing spatial cost topology. Individuals are distributed across a real number line (i.e. to represent their individual varying characteristics), and the cost of directly connecting between individuals depends on the 'length' between them. Hojman and Szeidl (2003) [17] also focus on cost heterogeneity, focusing specifically on core-periphery star based structures, using a refinement of the pairwise stability equilibrium concept. McBride (2004) [27] takes an entirely new approach focusing on imperfect monitoring between individuals to generate an equivalent of agent heterogeneity.

However, to our best knowledge, no paper has so far analysed exclusively the role of value heterogeneity in a pairwise stable network setting with decay. This is our first theoretical contribution to the field. We allow for value heterogeneity among agents as it is more intuitive to introduce a disutility of connections between two individuals of differing types than through adjusting individual costs of connection. Moreover, value heterogeneity also impacts an individual's payoff from a network through indirect connections as well, whilst cost heterogeneity applies to an individual's direct connections. Thus, we believe value heterogeneity is more realistic in that it captures an individual's preference over not only their direct connections, but the connections of their closest connections, and so on. Our model therefore exhibits value heterogeneity but cost homogeneity.

We proceed by developing the model presented in Section 3.2. Suppose that 'types' that agents belong to are distributed along the $[0, 1]$ interval. An agent i, at the point of existence, is allocated a type $t_i \in [0, 1]$. We redefine the payoff function that allocates a utility value to agent i with type t_i from a direct connection to another agent j with type t_j under graph g as $u_i(t_i, t_j, g)$, and specify that the allocation rule generates a utility for the agent i under the standard rule:

$$U_i(g) = u_i(t_i, t_j, g) \tag{6}$$

We further assume that $u_i(t_i, t_j, g)$ is continuous and twice-differentiable. Moreover, our formulation allows for some reasonable assumptions to be placed upon the nature of this utility function. It is assumed that every agent would prefer to connect with individuals who have a type closer to his/her own. Thus:

$$\left.\frac{\partial u_i(t_i, t_j, g)}{\partial t_j}\right|_{t_j<t_i} \geq 0\,; \quad \left.\frac{\partial u_i(t_i, t_j, g)}{\partial t_j}\right|_{t_j>t_i} \leq 0\,; \quad \left.\frac{\partial u_i(t_i, t_j, g)}{\partial t_j}\right|_{t_j=t_i} = 0 \tag{7}$$

In other words, given agent i's type as t_i, and another agent j with type t_j, agent i gets a lower marginal utility from a connection with j as $|t_i - t_j|$ increases. We also restate equation (5) in light of this, where $d_i(g)$ is the *degree* of node i.

$$U_i(g) = \sum_{k\in\phi} \delta^{dist(i,k)} u_i(t_i, t_k, g) - d_i(g).c \tag{8}$$

To showcase a usage of this model, we now proceed to state and prove some propositions. We first deduce an upper bound for the number of non-empty components. Second, we characterise a set of pairwise stable networks under a set of parameter values.

Proposition 2. *Assume a heterogeneous connections model on a graph g with the types of agents drawn from the [0,1] interval. Further, let $T = \{T_1, ..., T_m\}$ be the set of available types, with each agent $i \in \phi$ having a type $t_i \in T$. The number of types is given by $|T|$. Then, a pairwise stable network has at most $|T|$ (non-empty) non-trivial components. If a component has an agent of a specific type, then all agents of the same type belong to the same component.*

Proof. This is a proof by contradiction. Suppose that the graph g is pairwise stable. Further assume that there exists at least two non-trivial components.

Let u^{ij} be the utility attributed to agent i from a connection with agent j. So if $ij \notin g$, then $u^{ij} = u_i(t_i, t_j, g + ij) - u_i(t_i, t_j, g)$; and if $ij \in g$, then $u^{ij} = u_i(t_i, t_j, g) - u_i(t_i, t_j, g - ij)$.

Consider first graphs g for which non-trivial components consist of agents all of the same type. Assume further that $u_i(t_i, t_j, g) < 0$ for $t_i \neq t_j$ and $i \neq j$. Take a type T_1 without loss of generality, and all components for which all agents within them are all of the type T_1. Consider $ij \in g$ in a component of type $T_1 \in T$. Since g is pairwise stable, then $u^{ij} \geq 0$. Let $kl \in g$ belong to a different component of type T_1. Since i is in a component with j, both being of type T_1, and k is not in this component, then it is the case that $u^{kj} > u^{ij} \geq 0$, since k receives the indirect benefits of i discounted by δ not included in u^{ij} as well as the direct benefit of j. Similarly, $u^{jk} > u^{ij} \geq 0$. Therefore, this contradicts pairwise stability since $jk \in g$. Hence, all agents, for a given type, are connected by at most one non-empty component. Furthermore, consider a component of a different type, say, $T_2 \in T$. No agent with type T_1 belonging to a component of all agents with type T_1 will choose to connect with a component of agents with type T_2 since $u_i(T_1, T_2, g) < 0$ for any $i \in \phi$ with type T_1. Thus, a pairwise stable network under these assumptions can have at most $|T|$ non-trivial components.

Now consider all graphs g where non-trivial components consist of agents of potentially different types. Place no restriction on $u_i(t_i, t_j, g)$. If g is pairwise stable, and all non-trivial components consist of agents of the same type, then from before, there are at most $|T|$ non-trivial components. Now consider g such that a non-trivial component contains node i of type T_1 and node j with type T_2 w.l.o.g. with link ij. Then if g is pairwise stable, $u^{ij} \geq 0$. Consider another component with node k with type T_1 connected to another node l with, w.l.o.g, type T_3. Then, as before, $u^{ki} > u^{ij} \geq 0$ and $u^{ik} > u^{ij} \geq 0$, contradicting the assumption that g is pairwise stable. Then all agents of type T_1 will choose to link to this component. It does not matter subsequently whether l wants to change its connections - at worst, it becomes a singleton node. The same can be argued for j and type T_2. Thus, if a component shares agents of different types, all agents of the same type will be present in the component. Hence, the number of non-trivial components is bounded upwards by the number of types $|T|$. □

Proposition (2) is a generalisation of Proposition (1)(i), utilising much of the same logic of Jackson and Wolinsky (1996) [22]. It is a useful proposition as it restricts the set of all possible pairwise stable graphs g.

We now consider some results relating to the characterisation of pairwise stable networks. For simplicity, we consider a network with only two types T_1 and T_2.

Proposition 3. *Assume a heterogeneous connections model with types drawn from the interval [0, 1]. Let there be types $T = \{T_1, T_2\}$ such there are only two types. Further w.l.o.g, assume that $\delta u(T_1, T_2, g) \geq \delta u(T_2, T_1, g) > c$. Then:*

(i) For $c < (\delta - \delta^2).u(T_1, T_2, g) \leq (\delta - \delta^2).u(T_2, T_1, g)$, the complete network is the unique pairwise stable network.

(ii) For $(\delta - \delta^2).u(T_1, T_2, g) < c < (\delta - \delta^2).u(T_1, T_1, g)$ and $(\delta - \delta^2).u(T_2, T_1, g) < c < (\delta - \delta^2).u(T_2, T_2, g)$, there is a unique pairwise stable equilibrium. Each agent of a particular type is connected to one another, so there are complete connections between members of the same type. Each member of a type has a unique, and single, connection to a member of the opposite type, if there is one available.

(iii) If $(\delta - \delta^2).u(T_1, T_1, g) < c < \delta.u(T_1, T_2, g)$ and $(\delta - \delta^2).u(T_2, T_2, g) < c < \delta.u(T_2, T_1, g)$, a star, with the centre of any type, is a pairwise stable network, but it may not be unique.

Proof. (i) By $c < (\delta - \delta^2).u(T_1, T_2, g) \leq (\delta - \delta^2).u(T_2, T_1, g)$, we have that $c < (\delta - \delta^2).u(T_1, T_1, g)$ and $c < (\delta - \delta^2).u(T_2, T_2, g)$. By Proposition (1)(ii), all agents of the same type are all directly connected to one another. Now consider a graph g in which an agent i of type T_1 is not directly connected to an agent j of type T_2. For agent i, if we form the link ij, then i gets a benefit of at worst $(\delta - \delta^2).u(T_1, T_2, g) - c$ (in giving up an indirect connection to j). Therefore $u^{ij} = u(T_1, T_2, g + ij) - u(T_1, T_2, g)$ is strictly positive. Similarly the same applies to j with $u^{ji} > 0$. Therefore there cannot be any pairwise stable graph g which is not completely connected.

(ii) By $c < (\delta - \delta^2).u(T_1, T_1, g)$, all agents of type T_1 are directly connected to one another. The same applies to agents of type T_2 since $c < (\delta - \delta^2).u(T_2, T_2, g)$. Assume that there exists a pairwise stable graph g where two agents of the same type, say k and l of type T_1 w.l.o.g, both have a direct connection to an agent of type T_2, say m. From before k and l are directly connected. If k chooses to delete his link with m, then he gets a marginal payoff of $c - (\delta - \delta^2)u(T_1, T_2, g) > 0$, so does so. If there are more than two agents of type T_1 connected to the same agent of type T_2, the same argument applies. Equally, if k and l were of type T_2, and m of type T_1, the same argument applies. Thus, there can be at most one direct connection between agents of different types. Moreover, no agent of a single type will have more than one connection to the component consisting of agents of the other type, since generating the second connection will yield a marginal utility of $(\delta - \delta^2).u(T_1, T_2, g) - c < 0$ or $(\delta - \delta^2).u(T_2, T_1, g) - c < 0$ as the direct connection will forego the benefits of an indirect connection valued at δ^2. Finally, we show that a graph g consisting of two separate complete networks of the individual types, non-overlapping, is not pairwise stable. This is because, w.l.o.g, an agent i, of type T_1, can directly connect to an agent of the other component of type T_2 and gain a minimum of $\delta u(T_1, T_2, g) - c > 0$. The same applies for agents of the other type. Thus, the unique pairwise stable network is with all agents of the same type directly connected to one another, and where every agent of this type, if a connection is available, will have a unique direct connection to an agent of the other type.

(iii) Consider a graph g that is a star, i.e. one agent is the centre node, and all other agents have a direct connection to the central agent. W.l.o.g say that the central agent is of type T_1. The central agent has no incentive to delete links to either agents of the other type (since $c - \delta u(T_1, T_2, g) < 0$), or to agents of the same type (since $c - \delta u(T_1, T_1, g) < c - \delta u(T_1, T_2, g) < 0$). Consider a periphery agent of type T_1. This agent has no incentive to connect to another periphery agent of either type, since $(\delta - \delta^2)u(T_1, T_2, g) - c < 0$ and $(\delta - \delta^2)u(T_1, T_1, g) - c < 0$. Obviously, a periphery agent will not want to delete the link with the centre agent irrespective of its type since $c < \delta.u(T_1, T_2, g) < \delta.u(T_1, T_1, g)$. Therefore, the star is a pairwise stable network.

□

It is interesting to still find unique pairwise equilibria for some parameter value ranges when we introduce heterogeneity. In particular, we are able to showcase that complete integration of types is still possible under heterogeneity, albeit under more and more restrictive cost ranges.

Proposition (3)(iii) allows for multiple pairwise stable equilibria. The star, with either type at its centre is a pairwise stable equilibrium. What this analysis highlights is that in networks with different types, and where types are in communication with one another (i.e. there is a degree of integration), the resulting network structure may exhibit a degree of centrality. These networks may have particular agents who are 'gatekeepers' to other individuals of the same type, and all others have to communicate through these gatekeepers to gain access to others. This feature is indicative of a small-worlds property common in many social networks.

Proposition (3) is also interesting in that no agent receives any disutility from connections, either direct or indirect, with agents of the other type. Regardless, we still do not get complete integration between communities. This is because the overriding feature is that present in the homogeneous connections model. Agents attempt to economise on their direct connections if the cost of connecting is too high. Instead they rely on the positive benefits (i.e. externalities) that their indirect connections provide.

For completeness, we also consider the case where a direct connection to a member of the other type yields a cost instead of a net gain. Notice that there is no disutility from an indirect connection to an agent of the other type.

Proposition 4. *Assume a heterogeneous connections model with types drawn from the interval [0, 1]. Let there be types* $T = \{T_1, T_2\}$ *such there are only two types. Further w.l.o.g, assume that* $c > \delta u(T_1, T_2, g) \geq \delta u(T_2, T_1, g)$. *Then for any agent of a given type, if he/she has a direct connection to a second agent of another type, then this second agent must have at least two links in any pairwise stable network.*

Proof. This is a complementary proposition to Proposition (1)(iv). Suppose a non-empty network g is pairwise stable whereby an agent i, w.l.o.g, of type T_1 has a direct connection to an agent j of type T_2. Suppose further that j's only direct connection is with i. Then i can strictly benefit by deleting its link with j and receive $c - \delta u(T_1, T_2, g) > 0$. Hence g is not pairwise stable. □

It is entirely possible to have pairwise stable networks under such assumptions where agents of different types are in the same components. The net cost of a direct connection between agents of different types may be outweighed if an agent of a specific type brings enough indirect connections to be worth connecting to. However, the driver of such network formation lies primarily in the cost of connection, and impacts all types equally.

The true value of heterogeneous models is to assess situations where an agent actually gains disutility from a connection, either direct or indirect, to an agent of a differing type. In particular, are all non-empty pairwise stable networks truly segregated in the sense that agents of a given type reside in their own distinct components? The following proposition characterises the nature of pairwise stable networks under this framework.

Proposition 5. *Assume a heterogeneous connections model with types drawn from the interval [0, 1] with a finite number of agents. Let there be types* $T = \{T_1, T_2\}$ *such there are only two types. Further w.l.o.g, assume that* $0 > u(T_1, T_2, g) \geq u(T_2, T_1, g)$. *Then:*

(i) Any non-empty acyclic pairwise stable network consists of at most two separate components. There is no path in either component which contains agents of both types. Each component is homogeneous in terms of types.

(ii) If $c < (\delta - \delta^2)u(T_1, T_1, g) \leq (\delta - \delta^2)u(T_2, T_2, g)$, *the unique acyclic pairwise stable network consists of two separate components, each only consisting of agents of the same type, completely connected.*

Proof. (i) Suppose an acyclic pairwise stable network g has at least one non-trivial component. Further suppose that one of these components contains agents of types T_1 and T_2. By definition, an agent i of type T_1 must have a direct connection to an agent j of type T_2. For the link ij to occur in a pairwise stable network, then i must be gaining an indirect benefit from its connection to j since the direct connection to j yields a disutility of $u_i(T_1, T_2, g) - c < 0$ (and similarly for j as well). This indirect utility is only attainable from a component called B_1 accessed through the shortest path via j (otherwise i would gain the indirect benefit from another path). Denote the utility of the component B_1 as U_{B_1} via j. Then the utility for i from link ij is $\delta^2 U_{B_1} + \delta u_i(T_1, T_2, g) - c > 0$ since the component B_1 is at a path length of 2 away from i.

The component B_1 must contain at least one agent of type T_1, say k_1, that has at least one direct connection to an agent of type T_2. By Proposition 4, k_1 must have at least two links. An agent of type T_2 will only want to directly connect to k_1 if there is a component accessible only through k_1 which contains agents of type T_2. Call this component B_2, and let an agent of type T_2 in B_2 be m_2. Moreover, $B_2 \subset B_1$ since g is acyclic. Similarly k_1 will only want to maintain a connection to component B_2 if it contains agents of type T_1 that are directly connected to m_2. These agents of type T_1 themselves will only want to directly connect to m_2 if there is a component $B_3 \subset B_2$, accessible only through m_2, which contains agents of type T_1.

Thus the problem repeats itself over and over again, without termination. As the graph has a finite number of agents, at some point an agent of, say w.l.o.g, type T_1 will directly connect to a component (or singleton) only consisting of type T_2. Thus any links between these agents will be severed, and links between types T_1 and T_2 will be severed recursively. Thus g cannot be pairwise stable where there is a path containing agents of different types.

(ii) Proposition (5)(i) highlights that agents of the same type will all reside in the same component with no other agents of the other type for any acyclic graph. If $c < (\delta - \delta^2)u(T_1, T_1, g) \leq (\delta - \delta^2)u(T_2, T_2, g)$, then two applications of Proposition (1)(ii) to the two components individually will generate completely connected components. □

Proposition (5) is strong in its prediction. It predicts that if there is any disutility between two different types in a heterogeneous connections model (with only two types), then complete segregation between the two types, for any cost range, will be a feature in acyclic non-empty non-trivial pairwise stable networks.

Note however that in cyclic graphs, agents of either types can still have a direct connection. Suppose for example that there are 6 agents, 3 of type T_1 and 3 of type T_2. Moreover, suppose that each agent has two connections to agents of the other type only (hence a cycle of alternating types). Consider an agent i of type T_1. i will not want to delete one of its two links if $c - \delta u_i(T_1, T_2, g) - \delta^2 u_i(T_1, T_1, g) + \delta^4 u_i(T_1, T_1, g) + \delta^5 u_i(T_1, T_2, g) < 0$. i will further not choose to link with another agent of type T_1 if $c > \delta u_i(T_1, T_1, g) - \delta^2 u_i(T_1, T_1, g) + \delta^2 u_i(T_1, T_2, g) - \delta^3 u_i(T_1, T_2, g)$. These provide bounds on c. Some tedious algebra then yields the inequality $u_i(T_1, T_1, g)(2\delta - 1 - \delta^3) > u_i(T_1, T_2, g)(\delta(1 - \delta) - (1 - \delta^4))$. The RHS of this inequality is positive as $u_i(T_1, T_2, g) < 0$ and $1 - \delta^4 = (1 - \delta^2)(1 + \delta^2) = (1 - \delta)(1 + \delta)(1 + \delta^2)$. Thus $(1 - \delta)(1 + \delta)(1 + \delta^2) > (1 - \delta) > \delta(1 - \delta)$. Suppose that $u_i(T_1, T_2, g)$ is extremely close to 0. Consider $2\delta - 1 - \delta^3$. In the positive domain for δ this expression has real roots at $\delta = 1$ and $\delta = \frac{\sqrt{5}}{2} - \frac{1}{2}$ with the expression itself being positive between these two real roots. Thus the cost range for c is valid, and the cycle is pairwise stable (the same analysis can be conducted for agents of the other type as the problem is symmetric in approach).

The reason for why the proof for Proposition (5) does not carry through to cyclic graphs is due to the refining of component sizes. For two agents that provide disutility to one another to directly connect, there must be components linked to these agents that provide both of them with utility. Our proof above shows that, with only two types in an acyclic network, there is no stable component that is not a singleton that allows this to occur. We reduce the size of these components since the graph is acyclic as the proof iterates until there are no more agents left in the network. If the graph is cyclic, this argument does not hold as the counter example above shows.

A key question therefore arises - is it possible to connect types, which generate disutility to one another, in a heterogeneous acyclic network with more than two types? We first state a useful definition below for brevity of exposition.

Definition 1. (*Gross disutility*) An agent i with type T_1 is said to provide gross disutility to agent j with type T_2 if $u_j(T_2, T_1, g) < 0$.

We now state a general theorem which enables the existence of acyclic pairwise stable networks to occur where agents of different types who provide gross disutility to one another are directly connected.

Theorem 1. *Let the graph g be acyclic. Assume a heterogeneous connections model with types given by $T = \{T_1, ..., T_m\}$, drawn from the [0, 1] interval. Consider two types, where either or both provides gross disutility to one another. Suppose further that there is a direct connection between agents of these two types, say, w.l.o.g, between agents i of type T_1 and j of type T_2. Let A be the component accessible via i, and B be the component accessible via j. Then in any pairwise stable network, if i faces gross disutility from its connection to j, then B must contain at least one agent of a third type, say T_3, that provides utility to i. Similarly, if j faces gross disutility from its connection to i, then A must contain at least one agent of a third type that provides utility to j.*

Proof. This is a repeated application of Proposition (5). Consider an agent i of type T_1 directly connected to agent j of type T_2. Let A be the component accessible via i, and B be the component accessible via j. Suppose further that i gets gross disutility from its connection to j. If B contains only agents of type T_2, then i will sever its connection to j. If B only contains agents of type T_1, then j will want to severe its connection to B. If B contains agents of types T_1 and T_2, then by Proposition (5), g cannot be pairwise stable. Therefore, B must contain at least one agent of a separate type T_3 that provides utility to i. □

To show an example of this in action, consider a graph g, with three types T_1, T_2, and T_3. There are 4 agents: a and b of type T_3, c of type T_1 and d of type T_2. c and d provide gross disutility to one another. Consider a chain where a is directly connected to c, c to d, and d to b. Assume for simplicity that $u_a(T_3, T_1, g) = u_b(T_3, T_1, g) = u_a(T_3, T_2, g) = u_b(T_3, T_2, g)$ and symmetry via $u_a(T_3, T_3, g) = u_b(T_3, T_3, g)$. Further, assume $u_c(T_1, T_1, g) = u_d(T_2, T_2, g)$ - and $u_c(T_1, T_2, g) = u_d(T_2, T_1, g)$. Finally, assume $u_c(T_1, T_3, g) = u_d(T_2, T_3, g)$.

Let us first consider agent a (analysis is the same for b). a will not delete its link if:

$$c - \delta u_a(T_3, T_1, g) - \delta^2 u_a(T_3, T_2, g) - \delta^3 u_a(T_3, T_3, g) < 0 \tag{9}$$

a will not add a link to d if:

$$\delta u_a(T_3, T_2, g) + \delta^2 u_a(T_3, T_3, g) - \delta^2 u_a(T_3, T_2, g) - \delta^3 u_a(T_3, T_3, g) - c < 0 \tag{10}$$

a will not add a link to b if:

$$\delta u_a(T_3, T_3, g) - \delta^3 u_a(T_3, T_3, g) - c < 0 \tag{11}$$

Equation (11) dominates equation (10) since $u_a(T_3, T_3, g) > u_a(T_3, T_2, g)$. Equations (9) and (11) are consistent if $\frac{u_a(T_3,T_1,g)}{u_a(T_3,T_3,g)} > \frac{1-2\delta^2}{1+\delta}$.

Now consider c (the analysis is the same for d). c will not delete its link with a if:

$$c - \delta u_c(T_1, T_3, g) < 0 \tag{12}$$

c will not delete its link with d if:

$$c - \delta u_c(T_1, T_2, g) - \delta^2 u_c(T_1, T_3, g) < 0 \tag{13}$$

c will not connect with b if:

$$\delta u_c(T_1, T_3, g) - \delta^2 u_c(T_1, T_3, g) - c < 0 \tag{14}$$

Equation (13) dominates equation (12) since $\delta < 1$ and $u_c(T_1, T_2, g) < 0$. Equations (13) and (14) are consistent if $2\delta - 1 > -\frac{u_c(T_1,T_2,g)}{u_c(T_1,T_3,g)} > 0$. This completes the stability analysis by symmetry. For example, such a network is pairwise stable if the gross disutility between c and d is very small, and the utility that a and b get from c (and therefore d) is similar to the direct utility they would get by directly connecting themselves.

3.4 A Heterogeneous Connections Model with Tolerance

There are natural limitations to the heterogeneous connections model. In real world examples we see networks bifurcated with two distinct parties still communicating with one another. For example, two political parties may be opposed from an official policy point of view, yet some members of either party may occasionally speak to one another. Naturally these networks do not always feature cycles. We therefore need a richer framework than the one we have currently designed so far.

Our solution is to introduce heterogeneity between members of the same type. Consider two agents of a given type. One such member may choose to shun other types due to the gross disutility associated with a direct connection. However, the other member may be tolerant of some other types, and so might be willing to connect to some other types. Therefore, we introduce the idea that members of the same types may have varying degrees of tolerance for members of other types.

The way this can be achieved is by adjusting the utility function that dictates the utility agent i gets from its connection to agent j. In the previous section we defined this as $u_i(T_1, T_2, g)$ where T_1 is the type of i and T_2 is the type of j. If we want to introduce heterogeneity within types, we need an additional parameter for this functional form. We define the *degree of tolerance* as σ. Thus, the utility function is now defined as $u_i(T_1, T_2, \sigma_i, g)$.

We further posit the following relational features for the utility functions with respect to the degree of tolerance, that says that an agent's utility from a connection to another is weakly increasing as the agent's degree of tolerance increases:

$$\frac{\partial u_i(t_i, t_j, \sigma_i, g)}{\partial \sigma_i} \geq 0 \tag{15}$$

The theoretical framework is very rich and general now. The scope for generating general propositions is therefore limited, and so the paper from now on will be focused on exploring interesting network structures, analysing the intuition behind such results, and on simulating statistical features of such networks.

The remainder of this section will now be focused on definitions and explaining some intuition behind the means of network formation.

Definition 2. (*Active tolerance/active intolerance*). Consider an agent i with type T_i. Let there be another type T_j with $T_i \neq T_j$. Agent i is said to be *actively tolerant* of type T_j if there exists a degree of tolerance σ_i such that $u_i(T_i, T_j, \sigma_i, g) > 0$. i is *actively intolerant* of T_j if $u_i(T_i, T_j, \sigma_i, g) < 0$.

This allows for an interesting trade-off to be analysed when agents make links. We have seen before that an agent of a given type may choose to link with another agent who provides them with gross disutility, because the indirect benefits from such a connection outweigh any costs. Now we are able to explore the possibilities that only certain agents of a given type may want to make such connections. They may have a larger degree of tolerance for the type they are directly connecting to. They may have a larger degree of intolerance for some of these indirect connections. In fact, as we will see, it is possible for some agents to shun members of their own type due to the connections that they hold.

We also define some tolerance bounds for utility functions for the analysis of general behaviour of networks:

Definition 3. (*Minimum absolute tolerance (MAT)*). Let the set of degrees of tolerance, for which an agent i of type T_i will have a non-negative utility from a connection to any other agent, be $\beta_i = \{\ \sigma_i \mid u_i(T_i, T_j, \sigma_i, g) \geq 0, \forall T_j \in T\ \}$. The *minimum absolute tolerance* ${\sigma_i}^{MAT}$ for agent i is the lowest degree of tolerance in the set β_i for which this is true. In other words, ${\sigma_i}^{MAT} = inf(\beta_i)$.

Definition 4. (*Minimum absolute intolerance (MAI)*). Let the set of degrees of tolerance, for which an agent i of type T_i will have a negative utility from a connection to any other agent other than to agents of its own type T_i be:

$$\alpha_i = \left\{ \sigma_i \;\middle|\; \begin{Bmatrix} u_i(T_i, T_i, \sigma_i, g) \geq 0 \\ u_i(T_i, T_j, \sigma_i, g) < 0 \; if \; T_i \neq T_j \end{Bmatrix} \right\}$$

The *minimum absolute intolerance* ${\sigma_i}^{MAI}$ for agent i is the highest degree of tolerance in the set α_i for which this is true. In other words, ${\sigma_i}^{MAI} = sup(\alpha_i)$.

To provide some clarity to this, we introduce a functional form that we use throughout the rest of the paper. We denote $u_i(T_i, T_j, \sigma_i, g)$ to be:

$$u_i(T_i, T_j, \sigma_i, g) = exp\Bigg(-\frac{1}{2}\bigg(\frac{T_i - T_j}{\sigma_i}\bigg)^2 \Bigg) - \bar{c} \tag{16}$$

The functional form above is similar to the Normal distribution probability density function. Since the exponent itself is always non-negative, we can introduce disutility between types by subtracting a constant $\bar{c}$. The utility function itself has some promising features. First, it is symmetric around an agent's given type T_i. The larger the distance of another agent's type (say T_j), the lower the utility an agent of type T_i receives. The highest utility an agent of type T_i can receive is through a connection of an agent of the same type. Second, it is intuitive to add a degree of tolerance to the function. If $\sigma_i = 0$, the function becomes degenerate. As $\sigma_i \to \infty$, agent i will have absolute tolerance for all other individuals. As σ_i increases, agent i becomes more tolerant of other agents.

Applications for the heterogeneous connections model with tolerance can be found in Section 4 where we explore the role of intra- and inter-group segregation.

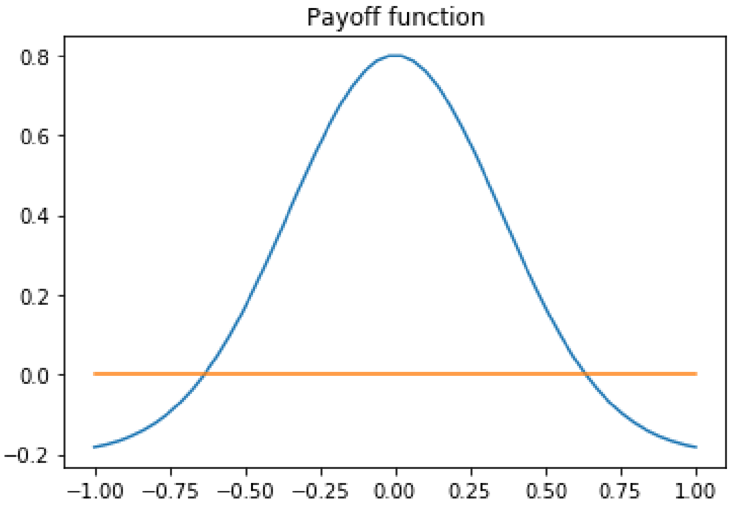


Figure 1: Utility function used throughout as per Equation (16) as we vary T_j between -1 and 1. For illustration, $T_i = 0$, $\sigma_i = \sqrt[2]{8}$, and $\bar{c} = 0.2$.

4 Applications to Social Networks

In this section, we apply our theoretical framework to a few test cases to showcase its richness in being able to generate social networks that exhibit group segregation. In particular, we construct two worlds and analyse pairwise stable networks in both. Section 4.1 concerns pairwise stable networks in a world where there are two distinct types of agents (i.e. the 'Left' and the 'Right', according to a political spectrum). Meanwhile Section 4.2 analyses pairwise stable networks when there are three distinct types of agents (i.e. the 'Left', the 'Centre', and the 'Right').

4.1 Two Party State

We assume a heterogeneous connections model with tolerance, and N agents each of two types, T_1 and T_2. We proceed to showcase the following network structures to highlight the scope of the model to generate segregation in social networks:

1. Complete inter-group segregation with intra-group integration
2. Both intra-group and inter-group segregation

Complete inter-group segregation with intra-group integration. Our theoretical framework allows for integration and segregation at the same time. Let each agent of types T_1 and T_2 have a degree of tolerance denoted by the minimum absolute intolerance level (MAI). As seen in Section 3, a cycle of alternating types is pairwise stable, but any acyclic graphs cannot have the types directly connected. In the latter case, agents of type T_1 and T_2 reside in separate components. In other words, there is complete segregation of the two types. Therefore, the network can be considered to be separable between types, and each component may be pairwise stable according to the relevant cost parameters. For example if $c < (\delta - \delta^2)u_i(T_1, T_1, {\sigma_i}^{MAI}, g) \leq (\delta - \delta^2)u_i(T_2, T_2, {\sigma_i}^{MAI}, g)$, then two separate components where both are completely connected is pairwise stable. This is the case where the pairwise stable network exhibits complete segregation between types and complete integration between agents of the same type.

If the cost of connection rises, then the separate components can be pairwise stable as per Proposition (1). For example, if $(\delta - \delta^2)u_i(T_1, T_1, {\sigma_i}^{MAI}, g) < c < \delta u_i(T_1, T_1, {\sigma_i}^{MAI}, g)$

and $c < (\delta - \delta^2)u_i(T_2, T_2, {\sigma_i}^{MAI}, g)$, then a completely connected component of type T_2 and a separate star of type T_1 is a pairwise stable network. An interesting observation can be noted here, namely the interaction between the cost of connection and value heterogeneity. In the example above, there is complete inter-group segregation and incomplete intra-group integration (i.e. type T_1 is a star and so all nodes are not perfectly connected). The segregation between groups is driven in our case by *value* heterogeneity with each type having gross disutility for the other. The intra-group integration, or lack thereof, is driven by the cost of connection as per the original Jackson and Wolinksy (1996) [22] model.

The cost of connection intersects with positive externalities to generate networks where nodes are not completely connected. Positive externalities, broadly speaking, occur when a third party's actions positively benefit another; and negative externalities if such actions have a negative impact. In our framework, indirect connections, depending on whether there is gross disutility or not, will yield positive or negative externalities. For agents of the same type, there are always positive externalities between them. Therefore, there is a tension between positive externalities and the cost of connection. Depending on the cost of connection, agents may choose to reduce their number of connections since the positive externalities outweigh the costs of a direct connection. Thus the cost of connection drives intra-group segregation.

For inter-group segregation, value heterogeneity amplifies the cost of connection. In the case where there is gross disutility between types, the negative externality enhances the cost of connection. Therefore, two types which have a gross distutility to one another will only connect if they both gain indirect positive externalities to make it worth the while. The externalities from indirect connections can also affect whether two agents of different types who provide an (albeit small) positive direct connection to one another will form a link. If an agent is connected to a component of types who provide gross disutility to others, then this agent may be 'shunned' by other agents, despite the fact that he may provide a positive direct connection.

Both intra-group and inter-group segregation. We are also able to generate inter-group and intra-group segregation concurrently. This is achieved by using differing levels of tolerance between types and also within a type. Suppose that all agents of type T_2 have a high degree of tolerance, say the minimum absolute tolerance level, MAT. Further suppose that $(N-1)$ agents of type T_1 have a degree of tolerance equal to the minimum absolute intolerance level, MAI, with the remaining agent of type T_1 having a degree of tolerance of MAT. Say that the cost of connection is zero. Finally assume that $|N\delta u_i(T_1, T_2, {\sigma_i}^{MAI}, g)| > |u_i(T_1, T_1, {\sigma_i}^{MAI}, g)|$. Recall that the definition of MAI means that agents of a specific type gets disutility from its connections with an agent of a separate type. Then a pairwise stable network is one where all nodes of type T_2 are completely connected, and are also completely connected with the one node of type T_1. The remaining nodes of type T_1 are in their own separate component, completely connected. They do not want to connect to the lone agent of type T_1 since the indirect connections to agents of type T_2 generates enough disutility such that they would rather not connect.

4.2 Three Party State

We can further show these forces at work in a three party state model. Assume a heterogeneous model with tolerance, with N agents each of three distinct types, T_1, T_2 and T_3. We proceed to show the following features that can generated under such a framework:

1. Complete inter-group segregation with intra-group integration

2. Both intra-group and inter-group segregation

3. 'Type capture' - where type A's connections to type B stop any connections between type C and type B
4. 'Compromise' - where two types, A and B which provide gross disutility to one another, are indirectly connected via a third type, say type C.

Complete inter-group segregation with intra-group integration. As we increase the number of types we can still generate complete inter-group segregation and intra-group integration. Let there be N agents of types $T_1, T_2, ..., T_N$, each with a degree of tolerance denoted by the minimum absolute intolerance level (MAI). Thus no agents of differing types will connect to one another. Further suppose that $c < (\delta - \delta^2)u_i(T_1, T_1, {\sigma_i}^{MAI}, g) \leq (\delta - \delta^2)u_i(T_2, T_2, {\sigma_i}^{MAI}, g) \leq (\delta - \delta^2)u_i(T_N, T_N, {\sigma_i}^{MAI}, g)$. Thus each agent of a given type will be completely connected to agents of the same type, and no agents of differing types will connect to one another. Thus we have complete inter-group segregation and complete intra-group integration. This applies to any number of types, including for three types.

Both intra-group and inter-group segregation. We can also show that intra-group segregation and inter-group segregation occurs in a three party model. One effective way is to reduce the problem to effectively a two-state model. Assume that there are three types with the following degrees of tolerance: $(N-1)$ type A agents with MAI (minimum absolute intolerance), one type A agent with MAT (minimum absolute tolerance); and N type B and C agents respectively, each with MAT. Further assume that the cost of connection is zero. Thus every agent of type B and type C will want to connect to every other agent. In any pairwise stable network all agents of types B and C will be completely connected. Equally, the lone type A agent with a MAT degree of tolerance will want to connect with every other agent, and in a pairwise stable network will connect with every agent of types B and C. Further assume that $|\delta N u_i(T_A, T_B, {\sigma_i}^{MAI}, g) + \delta N u_i(T_A, T_C, {\sigma_i}^{MAI}, g)| > |u_i(T_A, T_A, {\sigma_i}^{MAI}, g)|$ where the definition of MAI means that an agent of a given type gets disutility from connecting with an agent of a different type. The $(N-1)$ agents of type A only want to connect to agents of type A. They will not only refuse to connect with any agent of types B or C, but also the lone agent of type A with a MAT degree of tolerance. This is because the disutility gained from the $2N$ indirect connections to agents of types B and C will outweigh the single benefit (from a direct connection) that an agent A offers. Thus our framework is robust enough for generating intra- and inter-group segregation even with more than two types.

'Type capture' - where type A's connections to type B stop any connections between type C and type B. Our framework is rich enough to capture interesting scenarios where type A is able to deny type B access to type C. Say that the cost of connection is zero. Assume, w.l.o.g. that there are N agents each of types A, B, and C. Agents of type B are assumed to have a MAT degree of tolerance, and so all agents of type B will be completely connected to one another. Suppose that $|\delta u_i(T_A, T_C, {\sigma_i}^A, g)| > |u_i(T_A, T_B, {\sigma_i}^A, g) + (N-1)\delta u_i(T_A, T_B, {\sigma_i}^A, g)|$ and $|\delta u_i(T_C, T_A, {\sigma_i}^C, g)| > |u_i(T_C, T_B, {\sigma_i}^C, g) + (N-1)\delta u_i(T_C, T_B, {\sigma_i}^C, g)|$. Finally, assume that $u_i(T_A, T_B, {\sigma_i}^A, g) \geq u_i(T_C, T_B, {\sigma_i}^C, g) > 0$ and $u_i(T_A, T_C, {\sigma_i}^A, g) \leq u_i(T_C, T_A, {\sigma_i}^C, g) < 0$. Thus if there were no agents of type C, types A and B would be all completely connected, and vice versa if there were no agents of type A (for types C and B). However, if all three types exist, one pairwise stable network is that all agents of types A and B will be completely connected and agents of type C will reside in a separate fully connected component. No agent of type C would be willing to make a direct connection to any agent of type B since one indirect connection to one agent of type A would provide sufficient disutility to offset all of the benefits that type B could provide (1 direct connection and $(N-1)$ indirect ones). There is also another pairwise stable network where all agents of types B and C are fully connected, and agents of type A reside in their own separate fully connected component.

The real-world equivalent of this is 'guilt by association'. Agents of a given type A may not want to connect with another type B simply due to the connections that agents of type B have. This is despite the fact that, were those connections not present, agents of types A and B would mutually want to form links with one another. Thus, the tension between costs via indirect connections and benefits from direct connections can explain scenarios why some parts of social networks shun other parts due to the associations that they hold. In effect, certain groups' appeal for tolerance (i.e. type B in our example with MAT degree of tolerance) might in fact lead to fewer connections overall being made (i.e. in the case where agents of type B and A are connected, but there are more agents of type C who shun type B agents).

'Compromise' - where two types, A and B which provide gross disutility to one another, are indirectly connected via a third type, say type C. The tension between costs via indirect connections and benefits from direct connections can play out in reverse. Consider three types as before (A, B, and C), each with N agents. Agents of type B have a MAT degree of tolerance so will connect with one another fully, assuming that the cost of linking is zero. Assume that $u_i(T_C, T_B, {\sigma_i}^C, g) \geq u_i(T_A, T_B, {\sigma_i}^A, g) > 0$, so agents of types A and C would connect to all agents of type B ignoring any considerations of indirect connections. Finally, assume that $|u_i(T_C, T_B, {\sigma_i}^C, g) + (N-1)\delta u_i(T_C, T_B, {\sigma_i}^C, g)| > |N\delta u_i(T_C, T_A, {\sigma_i}^C, g)|$ and $|u_i(T_A, T_B, {\sigma_i}^A, g) + (N-1)\delta u_i(T_A, T_B, {\sigma_i}^A, g)| > |N\delta u_i(T_A, T_C, {\sigma_i}^A, g)|$ where $u_i(T_A, T_C, {\sigma_i}^A, g) \leq u_i(T_C, T_A, {\sigma_i}^C, g) < 0$ w.l.o.g. Therefore, an agent of type A will be willing to connect to an agent of type B, given that type B agent is connected to all other agents of type B and C. The direct (and indirect) benefits of the connections to B outweigh the indirect association with type C. In this regard, an agent of type A compromises on its connections to type B agents despite the disutility from type C agents. The same analysis applies vice versa with agents of type C.

5 Designing a Generative Algorithm

We proceed to dynamically generate social networks according to the theoretical framework we have thus designed. To do so requires the creation of an algorithmic procedure which is the focus of this Section. We first discuss the necessary terminating conditions for such an algorithm in Section 5.1. Then we highlight some issues with such a theoretical approach from a computational perspective (Section 5.2). Finally, we present our algorithm in Section 5.3 and the method it was implemented here.

5.1 Terminating Conditions

In any dynamic network formation process we need to identify a means by which the process terminates. There has been a little discussion on the means of identifying terminating conditions in the literature (Watts (2001) [33], Jackson and Watts (2002) [21]). In particular, Jackson and Watts (2002) [21] provide a useful proposition relating to terminating conditions in social network formation in the context where pairwise stability is the equilibrium concept. We first provide some definitions before stating their proposition and providing an intuitive explanation behind it. Recall that v corresponds to the value function of a network, and U to the allocation rule, as in Section 3.1.

Definition 5. (*Improving Path* from *Jackson and Watts (2002) [21]*). An *improving path* from a network g to a network g' is a finite sequence of adjacent networks $g_1, ..., g_k$ with $g_1 = g$ and $g_K = g'$ such that for any key $k \in \{1, ..., K-1\}$ either:

$$(i)\ \ g_{k+1} = g_k - ij\ for\ some\ ij\ such\ that\ U_i(g_k - ij, v) > U_i(g_k, v)$$
$$(ii)\ \ g_{k+1} = g_k + ij\ for\ some\ ij\ such\ that\ U_j(g_k + ij, v) \geq U_j(g_k, v)\ and\ U_i(g_k + ij, v) \geq U_i(g_k, v)$$

We have slightly adjusted Definition (5) to allow for the creation of a link if both agents are indifferent to doing so (for computational ease). The reasoning behind Definition (5) is intuitive. At a given time, an agent may choose another to directly connect to, and if both agents consent to the connection, then it is formed. Equally, at any given time, an agent may choose to delete a connection unilaterally without consulting the other agent. This approach enshrines myopic behaviour - agents may choose to delete a connection, not foreseeing that it may cause other links to be formed/deleted elsewhere in the network, with the end result that their own utility will be worse off. However in networks where agents have limited information over the total network structure, this may be an appropriate approximation for such situations.

We further need a definition regarding the nature of cycles:

Definition 6. (*Cycles* from *Jackson and Watts (2002)* [21]). A set of networks C form a *cycle* if for any $g \in C$ and $g' \in C$ there exists an improving path connecting g to g'. Moreover a cycle C is a *maximal cycle* if it is not a proper subset of a cycle. Finally a cycle C is a *closed cycle* if no network in C lies on an improving path leading to a network that is not in C. A closed cycle has to be a maximal cycle.

We are therefore now able to state the lemma by Jackson and Watts (2002) [21] without proof regarding the termination of network formation processes:

Lemma 1. *(Jackson and Watts (2002) [21]). For any v and U there exists at least one pairwise stable network or closed cycle of networks.*

The proof for this lemma is intuitive and we sketch an understanding here. Naturally, a pairwise stable network cannot lie on an improving path to any other network. So if we take a network which is pairwise stable, then any algorithm should terminate. If the current state of the network is not pairwise stable, and a change is made to a link, then the resulting network is either pairwise stable (in which case we terminate the formation process), or not. If we never find a pairwise stable network on an improving path, then the path itself must be a cycle. If we consider only cycles, due to the finite number of networks there must be a maximal cycle in which there are no paths leaving the maximal cycle. In other words it must be a closed cycle.

The next natural question that then occurs is how we capture the possible existence of a cycle in an improving path. We need to store the history of a formation process. To do this, we follow the suggestion of Jackson and Watts (2002) [21], and introduce the concept of a *g-tree*.

Definition 7. (*g-tree* from *Jackson and Watts (2002) [21]*). Given a network g, a *g-tree* is a directed graph which has as vertices all networks and has a unique directed path leading from each g' to g.

A *g-tree* in effect is a network of networks, with each node recording the state of the social network at each change of state. In this paper, we make use of g-trees to identify when the social network formation process should terminate.

Lemma (1) states that we need to identify when a social network is either pairwise stable, or when the formation process has yielded a state of the social network that we have previously found at a prior iteration (and thus a cycle). At each stage of the formation process, if there is a change to the state of the social network (i.e. a link is formed or severed), we add a node to the g-tree representing the state of the social network. Each node will be associated with an adjacency matrix to represent the state of the social network. At the beginning of each iteration of the algorithm, we need to consult the g-tree - if there is a cycle, then we should terminate the algorithm. It is important to stress that the identification of a cycle is not a sufficient condition to say that a network formation process will never end - an algorithm may branch off from a cycle in the g-tree when a node is revisited. However, there will remain a positive probability, however slight, that the algorithm may not terminate, and for those reasons, at the first detection of a cycle in the g-tree, we will terminate the social network formation process.

There is another terminating condition that we need to account for when assessing when a social network formation process should terminate. We need to identify when a network is pairwise stable. We cannot use the g-tree since the existence of a unique directed path is not sufficient to identify whether the last node at the end of the path is pairwise stable or not. Instead, we need to consult the social network itself directly. A key consideration of pairwise stability is that no node wants to either add (with consent of another node) or delete a link. Therefore, it is sufficient to check every *possible* edge that can occur in a graph. For a graph g with N nodes, this is equivalent to $\frac{N(N-1)}{2}$ checks. We can store all potential edges in an array - at each iteration, we will have to check each potential link. If there is no change to the potential link in the network (i.e. if it previously existed, it still exists; and if it did not previously exist, it still does not), then we remove the potential link from the array. If a change to the network is warranted, then we commit the change to the social network, add its state as a new node to the g-tree, and reinitialise the potential edges array to now contain all potential links that may exist in a network.

It is important to note that all of the above is independent of the means by which network formation occurs - i.e. how links are chosen, whether links are assessed in groups or individually, etc. We now proceed to discuss further details on this in the section forthcoming.

5.2 Further Considerations

There are a few further considerations that need to be discussed in light with any network formation process. We first discuss the means by which potential links (or groups of potential links) are chosen to be assessed in relation to the pairwise stability conditions. Second, we discuss the initial conditions at the initialisation of the social network formation process, and the implications of such a choice on the range of social networks formed.

5.2.1 The Choice of Potential Links

A priori we will need to decide upon the means by which potential links are analysed to be in accordance with pairwise stability or not. There are multiple ways in which this can be achieved. At the outset, potential links can analysed in groups or individually. Watts (2001) [33] allowed agents to simultaneously delete the links they wanted to at the same time, whilst only allowing them to create one individual link. An alternative is to take each potential link in a network, and assess whether a link should be there or not, as per Jackson and Watts (2002) [21].

The former has the advantage that it can speed up computation time in the sense that a bulk number of links can be removed at once. However, it in effect changes the pairwise stability concept, as adding a link currently not in the network is now additionally associated with deleting any links that an agent no longer wants (the reader is directed to Watts (2001) [33] for further details).

The latter approach is more suitable, and is what is followed here in the paper. When assessing whether the state of a social network is pairwise stable or not, we have to check all potential edges in a network. As explained before, if the network is not pairwise stable, we make a change to a link, and re-start the process again with all potential edges. Otherwise, we will check each potential edge, and if no more are to be checked, then the network is pairwise stable and we can terminate the algorithm. We can randomly select these edges one at a time. If an edge is consistent with a pairwise stable network, then it can be discarded and another edge is randomly chosen from those that remain. This approach has the advantage of mirroring some real-life aspects. Individuals in real-world networks often meet sporadically and randomly, and thus this process reflects this reality. Unlike Jackson and Watts (2002) [21], we do not add a perturbation to the network - that is, where a link is either randomly created or destroyed - as we want to focus exclusively on the results that dynamic network formation generates without creating additional uncertainty.

5.2.2 Initial Conditions

It should be clear by now that there are many multiple equilibria that can arise from a network formation process. Moreover, the initial starting conditions of a network can have a critical impact on the resulting network structure after a network formation process.

The quickest way to see this is to consider the original homogeneous connections model where $c > \delta$. If the network begins with no connections (i.e. the empty network), then there will be no connections made at all, and the empty network is the pairwise stable network. Notice that it is not the unique pairwise stable network for this cost range. Indeed, we can initialise the network with a cycle of 5 agents, and no agent would want to add or sever a link. Hence, we can see that initial starting conditions can have a marked effect on the resulting pairwise stable networks that are generated.

There are two issues with focusing on initialising networks with connections already in place between agents. The first is that there is no systematic way aside from randomly assigning connections between nodes at initialisation. The second is the justification of such a measure. This paper is focused on understanding how and why social networks form. If we were to exogenously connect nodes (that otherwise would not be connected), and then proceed to run a network formation process, we would be undermining the very framework that we are trying to understand. We would be at risk of tampering with the end results of pairwise stable networks due to a priori adding connections to a network without a micro-founded framework. For that reason we initialise the network to be the empty network (i.e. no connections) at the outset before any network formation process takes place (as in Watts (2001) [33]).

Given our choice of initial condition, we can already generate some propositions on the resulting pairwise stable networks that our generative algorithmic process will generate. The first, on pairwise stable networks when costs are very high, is complementary to Watt (2001) [33].

Proposition 6. *Let there be a type set $T = \{T_1, ..., T_m\}$. Further suppose that there a set of agents $\phi = \{1, ..., N\}$. Assume that $c > \delta u_i(T_1, T_1, \sigma, g) \geq ... \geq \delta u_i(T_m, T_m, \sigma, g)$ for all $i \in \phi$ and for all σ. Then a network formation process beginning with the empty network g will only generate the empty network as the unique pairwise stable network.*

Proof. Consider any two nodes $i, j \in \phi$ of any two types T_1 and T_2 respectively (w.l.o.g). At the beginning of the network formation process, $ij \notin g$. i and j will not want to form a link since $c > \delta u_i(T_1, T_1, \sigma_i, g) \geq \delta u_i(T_1, T_2, \sigma_i, g)$ and $c > \delta u_j(T_2, T_2, \sigma_j, g) \geq \delta u_j(T_2, T_1, \sigma_j, g)$. Therefore no link will be formed. This holds for any two nodes in the network, so the empty network is the only pairwise stable network. □

Proposition 7. *Let there be a type set $T = \{T_1, ..., T_m\}$. Further suppose that there is a set of agents $\phi = \{1, ..., N\}$. For all $i \in \phi$, let the degree of tolerance be the minimum absolute tolerance given by $\sigma_i = {\sigma_i}^{MAT}$. For each $i \in \phi$, let T_i be i's type and let $\overline{T_i} \in T$ be the type that provides the least amount of utility to i. If for all $i \in \phi$, $c < (\delta - \delta^2)u_i(T_i, \overline{T_i}, \sigma_i, g)$, then the unique pairwise stable network is the complete network.*

Proof. Since for all $i \in \phi$, $\sigma_i = {\sigma_i}^{MAT}$, then no agent provides gross disutility to another, i.e. $u_i(T_i, \overline{T_i}, \sigma_i, g) \geq 0$. Therefore, every agent has an incentive to directly connect to one another since at worst a direct connection gives any agent i utility of $(\delta - \delta^2)u_i(T_i, \overline{T_i}, \sigma_i, g) - c > 0$. Equally, no agent would want to sever an existing link as this would yield a negative marginal utility. Therefore, the unique pairwise stable network is the complete network. □

The above two propositions are extensions of previous propositions in Section 3.3. We now provide one further proposition below. By using the minimum absolute intolerance (MAI) measure, we are able to *decompose* a heterogeneous connections model into separable homogeneous connections models, each governed as per the cost ranges provided in Proposition (1). The significance of this is that if the degree of tolerance is too low across types, the resulting pairwise stable networks have separate components evolving independently of one another, with the number of (non-trivial) components upwards bounded by the number of types, as per Proposition (2).

Proposition 8. *Let there be a type set $T = \{T_1, ..., T_m\}$. Further suppose that there is a set of agents $\phi = \{1, ..., N\}$. For all $i \in \phi$, let the degree of tolerance be the minimum absolute intolerance given by $\sigma_i = {\sigma_i}^{MAI}$. Then any pairwise stable network g will be composed of at most $|T|$ non-trivial non-empty components, with each evolving independently from one another.*

Proof. Start at the empty network. Consider any two agents $i, j \in \phi$. If these agents are of different types, they will not form a link as this would yield a strictly negative marginal utility. Any connections formed or deleted between agents of the same type will be governed as per Proposition (1). Now consider any other stage of the network formation process. Assume that there are no connections between agents of different types. If two agents of different types get to form a link they will not choose to as they will get a strictly negative marginal utility. Therefore, no agents of different types will be directly connected. Since the initial network is the empty network, then by induction no agents of different types will be linked. There will be at most $|T|$ non-trivial non-empty components in any pairwise stable network g, and the evolution of these separate components are independent of any other types. □

5.3 The Generative Algorithm

In this section, we outline the generative algorithm and pseudo-code we have designed to implement our theoretical framework. First, we discuss the main elements and data structures we need to achieve this. Then, we provide a high-level explanation of how the process works. Finally, we present the pseudo-code of the whole network formation process.

We have already alluded to some elements we will need for a generative algorithm. The full list is as below:

1. **An undirected graph** - this will represent the social network itself. Each node will have a type, a degree of tolerance, and a utility function. Edges in the graph are represented by an adjacency matrix.

2. **A directed graph** - this will represent the g-tree. Each node represents a state of the social network during its formation process. Every node is associated with the adjacency matrix that captures information on the state of the social network at that given iteration.

3. **An array of potential edges** - this is an array that stores all the potential edges in the network that have not been analysed for pairwise stability. If a potential edge is selected from the array, it is removed, and assessed according to pairwise stability conditions. There are two cases here. Either the potential edge is consistent with the conditions, and thus the network is unchanged - in which case, the next potential edge is selected from the array (until the array is empty). Or, the potential edge is not consistent with the pairwise stability conditions, and so a link is formed if the potential edge does not currently exist, or the link is removed if it previously was there. Given the network structure has changed, this means that all potential edges have to be reassessed, and so the array needs to be reinitialised to contain all the potential edges that the graph may have between all its nodes.

We now provide a high-level explanation of how the generative procedure can work, splitting the explanation into two parts: initialisation and network formation.

Let us begin by briefly discussing initialisation. The social network itself, as described before, will be initialised as an empty network. Therefore, we need to initialise the undirected graph with a set of nodes only. We generate an array of nodes, with each element of this array being an array itself - containing the type of the agent, its degree of tolerance, and the utility function it is to use. The directed graph for the g-tree is created as an empty graph with no nodes initially. Before the network formation process begins, we create the adjacency matrix of the empty network (representing the initial state of the social network), and add that node to the g-tree. This creates the initial starting point to record the history of the network formation process, and identify any cycles that may occur in any such process. Finally we initialise the array of potential edges to contain all $\frac{N(N-1)}{2}$ potential edges that could occur in a graph g with N nodes. Each edge itself is represented as a tuple - for example $(0, 1)$ states that a potential edge can occur between nodes 0 and 1.

We now provide some explanation of the network formation process itself. We begin the iteration of the network formation process with two checks. First, we check that the g-tree (i.e. the directed graph) has no cycles. Second, we check that the array of potential edges is not empty - if it is, then the social network is pairwise stable. If either of these checks fails, then the algorithm terminates.

Given these checks are passed, we remove at random a potential edge from the array to analyse. We store the two nodes on either side of the potential edge. We then check the social network's adjacency matrix for the status of this potential link. If the potential edge is not 'filled' - that is, there is no link between the two nodes - we need to check whether both agents are weakly better off with the link that without it. We first store the agents' utilities from the network at the beginning of the iteration. Then, we add the link, and recalculate the agents' utilities. If both with the link are weakly greater than without it, we recalculate the adjacency matrix, and add a new node to the g-tree with this adjacency matrix as a attribute to it (to store the state of the network). We further add an edge from the previously added node of the g-tree to the one we add now so we capture the evolution of the social network formation in the g-tree. Finally, we also need to reinitialise the potential edges array as we need to reassess all potential edges again with this new state of the social network. On the other hand, if either agent is worse off with the link than without it, we remove the link from the social network (and thus the state of the network remains the same as per the beginning of the iteration), and we proceed to the next iteration.

The process mirrors that of the above in the case where a potential edge, as per the adjacency matrix, already has a link in place between the two nodes. In this case, we calculate the utilities for both agents prior to any change in the network structure. Then we remove the link. If either agent is strictly better off without the link than with it, then the social network's state is changed. As before, we recalculate the adjacency matrix, add a new node to the g-tree (with an edge from the prior node added to the current one), and reinitialise the array of potential edges. Otherwise, we reinstate the link to the social network and proceed to the next iteration as the state of the network has not changed.

For ease of implementation, all such elements and data structures are created as attributes or members of a class object. Each class representing the pairwise stable network game (PSNG) therefore has a reference to, and instantiation of, its own social network (i.e. the undirected graph), the g-tree (i.e. the directed graph), and the array storing all potential edges that need to be considered.

Below we provide the pseudo-code of the network formation process. In the pseudo-code, the social network is referred to as *sn*, the g-tree as *g_tree*, the array of potential edges array as *potential_edges*, and the social network's adjacency matrix as *adj*. All actual code for implementation can be found in a separate ZIP file attached to this project on CATE.

It is worthwhile to briefly touch upon the means by which the g-tree is updated for a change in the social network's adjacency matrix. Throughout we maintain a reference to the last node added to the g-tree. When it is time to update the g-tree due to a change in the state of the social network, we first check whether this node already exists in the g-tree (i.e. by comparing the current adjacency matrix with those associated with the nodes already present in the g-tree). If the node does not exist, we add the node to the g-tree, and an edge between the previously added node to the one which we have just added (and in doing so we update the reference to the latest added node). Otherwise, if the node already exists in the g-tree (i.e. the revisited node), we add an edge between the reference to the last added node in the g-tree and the revisited node, thus creating a cycle.

Moreover, we briefly touch upon why we pursued an iterative approach to the social network formation problem. Another alternative would be to attempt a dynamic programming solution. However, dynamic programming relies upon overlapping sub-problems. In our case, say that during the formation process we revisit a potential edge. Despite it looking as if we are repeating the same analysis, this is not necessarily the case, as there may have been a change in some other element in the adjacency matrix. The utility function for an agent is primarily a function of the entire network structure. Moreover, if the formation process generated an adjacency matrix that we had previously visited (at some prior node in the g-tree), then the problem will not be analysed again, since there would be a cycle in the g-tree, and the algorithm would terminate. Therefore, a dynamic programming solution is not appropriate in this case.

Thus, we have to opt for a more computationally intense method. We are interested primarily in the statistical properties of networks as they form according to pairwise stability

Algorithm 1 Dynamic Network Formation

```
procedure DNF
    while g_tree has no cycle and potential_edges is not empty do
        Remove randomly selected edge from potential_edges and store in edge_to_assess
        first_node ← edge_to_assess[0]
        second_node ← edge_to_assess[1]
        is_edge_filled ← adj[first_node][second_node]
        if is_edge_filled == 0 then
            if check_to_add_edge(first_node, second_node) == 0 then
                Remove edge_to_assess from sn
            else
                Update adj
                Add adj to g_tree as a new node, with edge from last added node to adj
                Reinitialise potential_edges
        else
            if check_to_remove_edge(first_node, second_node) == 0 then
                Add edge_to_assess from sn
            else
                Update adj
                Add adj to g_tree as a new node, with edge from last added node to adj
                Reinitialise potential_edges
    return
```

Algorithm 2 Check to Add an Edge

```
procedure CHECK_TO_ADD_EDGE(node1, node2)
    node1_prior ← calculate_total_utility(node1)
    node2_prior ← calculate_total_utility(node2)
    Add edge between node1 and node2 to sn
    node1_after ← calculate_total_utility(node1)
    node2_after ← calculate_total_utility(node2)
    if node1_after ≥ node1_prior and node2_after ≥ node2_prior then
        return True
    return False
```

Algorithm 3 Check to Remove an Edge

```
procedure CHECK_TO_REMOVE_EDGE(node1, node2)
    node1_prior ← calculate_total_utility(node1)
    node2_prior ← calculate_total_utility(node2)
    Remove edge between node1 and node2 to sn
    node1_after ← calculate_total_utility(node1)
    node2_after ← calculate_total_utility(node2)
    if node1_after > node1_prior or node2_after > node2_prior then
        return True
    return False
```

rules. For that reason, we restrict ourselves to a small number of nodes in order to run as many simulations as possible and uncover insights. The reason for this is that if we can identify patterns that occur in small networks, then we can better understand the nature of sub-graphs in a larger network game. Many networks (i.e. social connections, economics etc.) are local, and more important develop locally. Hence, it makes sense to analyse small scale networks as a starting point in the analysis.

5.3.1 Implementation Methodology

All code, including the generative algorithm, was implemented in *Python* using a simple object oriented design explained in the previous section. We use Python as it abstracts away from a lot of the finer details regarding data structures, and thus our code is more legible. Additionally, it is considerably less verbose. Moreover, there are a suite of useful libraries that help accelerate development.

In particular, we make extensive use of $NetworkX$. $NetworkX$ contains many extremely useful functions and classes for the construction and manipulation of both undirected and directed graphs, as well as a series of standard algorithms for fast implementation (i.e. such as finding simple cycles in a directed graph). $NetworkX$ stores graphs as a three-layered dictionary. This allows for fast data lookups with a suitable amount of storage for sparse networks (such as our own during the early network formation process). Equally as important, we can store attributes associated with both nodes and edges. This therefore allows us to store the type, degree of tolerance and utility function type associated with each node in the social (undirected) graph. For the (directed) g-tree, we can store the adjacency matrix associated with each node as well, allowing us to identify when a social network has reached a state that has been previously visited.

We further make use of $NumPy$ and $Pandas$ for data manipulation and, post simulations, to run analysis on the collected network data. Finally, we further utilise $Matplotlib$ for data visualisation.

Three scripts have been generated for the purpose of the project. A *network.py* file maintains the class object with associated utility functions to run a dynamic social network formation process. The *generator.py* file allows the creation of a simulation of social networks for given parameter values, and stores the results in a *Pickle* object format. Finally the *analysis.py* file allows for the statistical analysis of these results.

6 Simulation Results

We then proceed to simulate the social networks generated from our algorithm. First, we define our own segregation measure, and the statistics we are interested in measuring, in Section 6.1. Second, we generate simulations for the two-party model, varying the cost of linking, and explain the statistical findings from our results in Section 6.2. Finally, in Section 6.3 we repeat the simulation analysis for the three-party model case.

6.1 Statistical Focus

To generate insights into the types of social networks formed, across a number of simulations, a series of statistical measures need to be identified. Below we describe the metrics used for the purposes of the project.

6.1.1 Defining a Segregation Measure

A key measure we need to define is that of segregation. There is no consistent measure of what segregation in a social network is, especially in the case of more than one type among nodes in the network. For that reason, we define and motivate our own measure of segregation for the purposes of the project.

Throughout the paper, we are interested in two types of segregation: *inter-group segregation* and *intra-group segregation*. Inter-group segregation refers to how well an agent of one type is connected to agents of the other type. By contrast, intra-group segregation refers to how well an agent of one type is connected to agents of the same type in the network. In ranges where the cost of linking is sufficiently low, both inter-group segregation and intra-group segregation would be expected to be low - agents are willing to connect both to their own type but also to other types as well, assuming that connecting to other types provides a net positive utility. Where the cost of linking is sufficiently higher, we would expect there to be higher inter-group segregation since agents of one type would gain a net negative utility (due to the cost of linking) from a connection to an agent of another type. At prohibitively high costs of linking, we would expect there to complete inter- and intra-group segregation since no agent has an incentive to form a link with any other agent - the network is full of singleton nodes.

In motivating a measure for segregation, there are a few ideal features that such a metric would have. First, the measure should be bounded for any social network that could be formed. Of all the possible social networks that can be formed, the two extremes in terms of network architecture that can be formed are the empty network (i.e. a network full of singleton nodes) and the complete network (i.e. every nodes is directly connected to all other nodes). Therefore an ideal measure of segregation is one that is bounded for the empty network, and has a distinct separate bound for the complete network *for any type of agent*. The reason that the latter property is ideal is since then the bounds for the segregation measure, for a complete network, are independent of the number of nodes and the number of types that any given social network might feature. Second, the segregation measure should take into account indirect connections since agents' payoffs are a function of their indirect connections as well as their direct ones.

Based on the above, we propose the following metric to measure segregation between types T_i and T_j:

$$S_{T_i,T_j} = \begin{Bmatrix} \frac{1}{|T_i|}\sum_{i\in T_i}\sum_{j\in T_j\setminus\{i\}} \frac{1}{d(i,j,g)}\cdot\frac{1}{|T_i|-1} \quad if \;\; T_i = T_j \\ \frac{1}{|T_i|}\sum_{i\in T_i}\sum_{j\in T_j} \frac{1}{d(i,j,g)}\cdot\frac{1}{|T_j|} \quad if \;\; T_i \neq T_j \end{Bmatrix} \tag{17}$$

In the case where $T_i = T_j$, then S_{T_i,T_j} measures intra-group segregation. For each node of this type, we calculate the sum of all the inverse geodesic distances (given by $d(i, j, g)$) between this node and all other nodes of the same type. This summation is inversely weighted by the number of all nodes of this type (bar the one under consideration), hence the weighting by $T_i - 1$. Finally, the average of this measure, across all nodes of this type, yields a measure for intra-group segregation. In the case, where $T_i \neq T_j$, then S_{T_i,T_j} measure inter-group segregation, with the same logic as before but for nodes of the other type T_j.

The segregation measure satisfies our required features. First, it takes into account indirect connections in the social network model since it uses geodesic distances to weight how closely nodes are connected to one another, and is therefore a rich and informative metric that goes just beyond considering direct connections. Moreover, it implicitly accounts for the case where two nodes are not connected at all in a network g - in this case, $d(i, j, g) \to \infty$, and the contribution of these two nodes to the segregation measure is zero, as to be expected.

Second, the segregation measure has both a lower and upper bound, independent of the social network architecture as well as the types and number of nodes. Consider the empty network. Then $d(i, j, g) \to \infty \ \forall j \in T_j$. It is easy to therefore see that any measure of intra-group or inter-group segregation will be zero. Similarly, consider the complete network. In this case $d(i, j, g) = 1$ between all nodes in the network. It is then trivial to show that the segregation score, for both intra-group or inter-group segregation, will equal 1. Thus S_{T_i,T_j} has both an upper and lower bound, and is applicable for all social network architectures as a comparable measure.

Regarding the interpretation of S_{T_i,T_j}, the higher S_{T_i,T_j} is, the more *integration* there is between nodes of types T_i and T_j in the network. Equally, the lower the measure, the higher the level of segregation for nodes of types T_i and T_j. If $S_{T_i,T_i} > S_{T_i,T_j}$ for $T_i \neq T_j$, then we can say that there is a higher level of inter-group segregation than intra-group segregation - that is, agents of the same type prefer to be connected to each other than agents of other types.

6.1.2 Other Measures

During the simulations of social networks, we further collect more statistical measures on the structure of the social networks formed. In particular, we collect information on the following:

Stability. During a social network formation process, we collect information regarding the termination state of the network. In particular, we capture whether the social network formation process converges to a pairwise stable network, or, revisits a previous state of the network (thus forming a cycle in the g-tree and hence terminating the algorithm). This measure can provide us with an insight into the likelihood that a social network will be stable for certain parameter values. It is not a perfect measure, since a cycle in a g-tree is not certain proof that a social network will not converge to a pairwise stable state (i.e. it provides evidence only that the network has a strictly positive probability of not converging to a stable state). However, it is indicative of suggesting which parameter values might provide more or fewer stable networks.

Number of cliques. In graph theory, a clique C, of an undirected graph $G = (V, E)$ - where V and E are vertices and edges respectively - is a subset of vertices of the graph G (i.e. $C \subseteq V$), such that any two distinct vertices belonging to C are directly connected. Thus the number of cliques is an indication of how well connected all nodes are in a given graph; the higher the number of cliques, the more nodes are directly connected. Thus, we use the number of cliques as an indication of the direct connectedness of a given network. In particular, we calculate the mean and variance of the number of cliques over the simulations of networks for given parameter values to better assess this.

Degree centrality. The *degree* of a node is the number of other nodes it is connected to. Thus, the highest degree for a node in a network with N nodes is $N-1$. If the degree of node i under network g is $d_i(g)$, then the degree centrality for i is the normalised metric given by $\frac{d_i(g)}{N-1}$. The higher the degree centrality for a node i, the more direct connections it has with other nodes in the network. An average measure of degree centrality of all nodes in the network therefore provides a measure of how directly connected a network is. We furthermore calculate the mean and variance of these average degree centralities over simulations of networks for given parameter values to better assess this.

6.2 Simulation Results: Two Party Model

We proceed to generate simulations of social network formation for the two-party model under a variety of parameter values to assess differing natures of the networks generated. In deciding the scope of our analysis we make the critical decision to hold the decay rate δ fixed for a given payoff function, and allow the cost of linking to vary. The reason for this is due to the fact that the network structure under pairwise stable networks is dependent upon the relative magnitudes of the cost of linking and the decay rate. Therefore, fixing one, and varying the other achieves the same analytical effect as vice versa. Moreover, we decide on a consistent payoff function across all simulations, as per Equation (16) which allows for symmetry around an agent's ideal type. In Figure (2) we summarise the model parameters:

Model Parameter	**Value(s)**
Number of iterations	10,000
Decay rate	0.9
Cost shift (for payoff function)	0.2
Number of nodes per type	7
Cost of linking	0.1 - 0.9 (at 0.1 intervals)
Type 1 preferred type	1
Type 2 preferred type	0

Figure 2: Two-Party Model Simulation Model Parameters

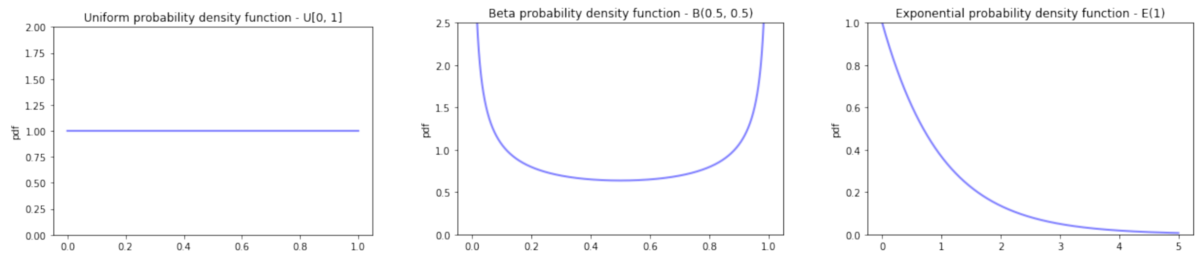
We adjust the cost of linking from 0.1 to 0.9 at 0.1 intervals to assess how network architectures change as generating links becomes prohibitively more expensive. Given the functional form given by Equation (16) and Proposition 6, we can calculate that if the cost of linking is above 0.72, then the empty network is the guaranteed pairwise stable network (since $0.9 \times (exp(0) - 0.2) = 0.72$). This provides us with a way to see whether our implementation of the model is correct.

In the setting of parameter values above, we allow for the heterogeneity between agents' types, with Type 1 agents having a preferred position of 1 (i.e. $T_1 = 1$), and 0 (i.e. $T_2 = 0$) for Type 2 agents respectively. However, we need to introduce heterogeneity within the types of agents to generate a fully flexible heterogeneous connections model with tolerance. To assess how the choice of the tolerance parameter impacts the pairwise stability of the network, we opt to repeat our simulations over a set of *probability distributions* (Figure (4)) over which tolerance parameter values are drawn for each node in the network. By doing this we are able to assess how similar social network architectures are over a wide variety of tolerance parameters, without making our results overly reliant on an ad hoc choice of a particular parameter value range. Figure (3) highlights the choice of distributions with a corresponding rationale.

We proceed to run the simulations for the parameter values and distributions above, and after collecting simulation data, process it to generate statistical insights regarding social network

Distribution	Rationale
Uniform ($U \sim [0, 1)$)	Even distribution of values
Beta ($B(0.5, 0.5)$)	Agents are more likely to be highly tolerant or highly intolerant, with bounds on tolerance
Exponential ($Exp(1)$)	Left tail towards intolerance; no upper bound on tolerance

Figure 3: Tolerance Probability Distributions



(a) (b) (c)

Figure 4: Various probability density functions used to generate tolerance parameter values for simulations: (a) Uniform distribution over the unit interval; (b) Beta distribution with both shape parameters at 0.5; (c) Exponential distribution with rate parameter set at 1

formation. The results can be seen in Figures (6) - (8). The simulated results generate a series of interesting insights, with a series of similarities and differences across different distributions for tolerances.

Stability. The first insight is that, for non-empty networks, the probability of a social network converging directly to a pairwise stable state without incurring a cycle follows a parabola with regard to the cost of linking. We can interpret this by counting the number of networks that directly converge to a pairwise stable state and dividing it by the total number of simulations. We define this sample statistic measure as the *probability of direct convergence*. In other words, as the cost of linking increases, so does the probability of direct convergence, up until a critical cost of linking. Past that point, the probability of direct convergence decreases until the cost of linking is sufficiently high that the empty network is the only pairwise stable network (i.e. the network does not deviate from its initial state). At these cost ranges, the probability of direct convergence is 1. This result is consistent across all tolerance distributions (Uniform, Beta and Exponential).

We can develop an intuition for these results. We introduce two notions - *connection abundance* and *connection scarcity*. When connections are abundant, this means that agents have been able to generate many links between one another (i.e. due to low cost of linking). Therefore, each agent, at a given network state, has a higher probability of holding more connections with other agents in the network than at a higher cost of linking. Thus, each agent therefore has more connections that they can delete to improve their own private utility since each agent, on average, has more connections, and thus the network is more tightly connected (i.e. lower average path lengths). If a network is more tightly connected, then agents have a higher probability of generating a substantial utility gain from deleting a link. We would therefore expect networks that exhibit connection *abundance* to have a lower probability of direct convergence. In effect, during the network formation process, there are many connections, formed in the past, that are deemed redundant at a later stage of network formation, and thus can be deleted. This can lead to a higher probability of cycles occurring. As the cost of linking increases, connection abundance decreases.

Connection scarcity is the opposite characteristic - at sufficiently high costs of linking, agents, on average, hold fewer connections with one another. Deleting a link, where warranted, can yield a large utility gain for an agent if they are able to still access other agents from their remaining links (and indirect connections). The network structure itself is sparse, and so when a link forms in the network, other agents can benefit greatly from indirect connections. They each now have a higher incentive to delete their links to economise on the cost of linking. Hence, with increased connection scarcity we would expect to see a lower probability of direct convergence. The links made in earlier stages of the network formation process can be removed due to indirect connections that an agent benefits from. As the cost of linking decreases, connection scarcity decreases.

To summarise: connection abundance is driven by the volume of total connections in the network structure. Agents can optimise by pruning and reducing their connections, gaining a small utility gain from each link deletion, and enjoying the multiple indirect connections they have. By contrast, connection scarcity is driven by the marginal gain of deleting a link due to a high cost of linking. Indirect connections are highly valuable, and an agent can delete a few links to the benefit of large utility gains, if they have access to enough indirect connections.

In light of this, the results we see are extremely interesting. At low costs of linking, *connection abundance* is the dominant effect (connection scarcity is low), and so the probability of direct convergence is low. As the cost of linking increases, we observe that the connection abundance effect decreases faster than the rate at which the connection scarcity effect increases. Hence, we observe an increase in the probability of direct convergence across all tolerance probability distributions. At higher costs of linking, the *connection scarcity* effect begins to get stronger, to the point that it decreases the probability of direct convergence once again. Hence, for non-empty networks, this is a solid explanation for why we observe this parabola effect. In terms of optimising for stable networks, the insight is therefore this: if there is to be a cost of linking (i.e. non-zero), a suitably high cost of linking may be preferable to a lower value.

With regard to this, there are some differences according to the different distributions that govern the tolerance parameter values. The Uniform distribution over the unit interval yields a benchmark distribution - when the connection abundance and connection scarcity are respectively dominant effects (i.e. at 0.1 and 0.7 costs of linking), the probability of direct convergence is low at roughly 83%-84%, peaking at approximately 94% at 0.3. The Beta distribution, by contrast has a much lower probability of direct convergence: at 80% and 84%-75% for 0.1 and 0.7 costs of linking respectively, with a peak at approximately 93% at 0.3. The difference is not overly surprising. The Beta distribution with both shape parameters at 0.5 generates a tolerance distribution where it is highly likely that an agent will be very tolerant (with a bound at 1) or very intolerant. Very intolerant individuals will be more likely to delete a link if an indirect connection they have via that link yields sufficiently high disutility, and therefore there will be a higher likelihood of a cycle occurring.

The Exponential distribution (with rate parameter of 1) yields some interesting results. The distribution is skewed towards 0 (and thus intolerance) yet has no upper bound on tolerance (unlike the Beta distribution). At costs of linking of 0.1 and 0.7, the probability of direct convergence is approximately 86% and 94% respectively, considerably higher than for the Uniform and Beta distributions. The peak probability of direct convergence is approximately 98% at roughly 0.4 cost of linking. An intuition for this is as follows. First the mean tolerance value under this Exponential distribution is 1, which is considerably higher than the Uniform and Beta distributions considered here (both means of $\frac{1}{2}$) - on average agents are more tolerant. Second, there is no upper bound on tolerance - some agents may be extremely tolerant of the other type, and this may offset the *connection scarcity* effect at higher costs of linking.

What the simulations and analysis show is that the probability of direct convergence follows similar patterns irrespective of the distributions considered here. Distributions over tolerance parameters are important insofar as they determine the magnitude of the probability of direct convergence for a given cost of linking.

Number of cliques. The number of cliques measures how directly connected the network is. Unsurprisingly, we find that for all distributions a negative correlation between the average number of cliques and the cost of linking - as the cost of linking gets more expensive, the network is less directly connected. What is particularly interesting is how quickly the number of cliques declines as the cost of linking increases - the number of cliques declines rapidly at low cost values before stabilising between 0.4 and 0.7 (recall that the empty network is the only pairwise stable network for link cost values over 0.7, hence why there are 14 cliques for the 14 distinct singleton nodes in the graph). This suggests that social networks are susceptible to link cost changes when the cost of linking is initially low; and that a small change in the cost of linking can rapidly lead to a less directly connected graph. To see why, we can introduce the notion of connection abundance. At low costs of linking, during the network formation process, many agents are forming many links with one another. As the network becomes more directly connected, the marginal gain of a further connection for a given cost of linking falls - for *all* agents in the network. If the cost of linking increases marginally, many links that were previously marginally beneficial can yield disutility. Hence, many agents would prefer to cut links.

The more surprising finding is that the variance in the number of cliques drops markedly as the cost of linking increases. At low costs of linking in particular, if there is a marginal increase in the link cost, then the variance in the number of cliques drops by far more than at higher costs of linking. This suggests that at low costs of linking (exempting complete networks and cost ranges where unique pairwise stable networks can be derived), there are more permissible pairwise stable networks that could have a variety of architectures. This suggests that if we observe a wide range of network structures across many separate samples for a given scenario, then the cost of linking is likely to be low. As the cost of linking increases, we can better identify the statistical features of any pairwise stable network. Based on these results we cannot say with certainty that there are fewer pairwise stable networks as the cost of linking increases - but we can say that such pairwise stable networks share increasingly more similar statistical features as the cost of linking increases. In other words, even though we might not be able to predict the exact structure of a pairwise stable network, we can say, with a high degree of certainty, what features it can exhibit, in situations where the cost of linking is known to be relatively high.

Degree centrality. Degree centrality is another measure for how directly connected the network is. The average degree centrality across agents in the network is similar to the insights gained from analysing the number of cliques. As the cost of linking increases, average degree centrality falls rapidly before stabilising at approximately 0.14. This is consistent across distributions over tolerances. Moreover, the variance of degree centrality falls dramatically at low costs of linking when the cost is marginally increased, in line with the insights gained by analysing the number of cliques. This suggests that at high costs of linking, permissible pairwise stable networks share common features regarding the number of cliques and average degree centrality, with a high degree of confidence.

Segregation score. The average segregation score showcases how well connected agents are, both to members of their type but also to members of other types. It is a better measure of how well connected a network is as it takes into account indirect connections. Since we only consider two types in this model set-up, the segregation score is symmetric between two different types, so we consider a generic type for our analysis (there is little variation between types for intra-group segregation scores). For all distributions it is not surprising that there is less intra-group segregation than inter-group segregation for all costs of linking, as agents prefer to connect with their own type (recall that the lower the segregation score, the higher the segregation between/within the groups - it is an *inverse* measure).

At a low link cost (0.1 - 0.3), as the cost of linking increases, both the intra-group and inter-group segregation measures decline, suggesting that segregation overall increases in the network. The decline is more pronounced for intra-group segregation. A feasible reason for this is that at low costs of linking, agents hold more direct connections with agents of the same type than to others of different types, since there is a positive marginal gain from doing so with each direct connection (and since they gain more utility from same-type connections). As the cost of linking increases, agents economise on links with the same type by relying more on indirect connections. For their connections with agents of different types, there is less room to reduce links since they previously were relying on indirect connections to get access to other types of agents. This is consistent with the other measures (i.e. direct connection measures as per the number of cliques and degree centrality) discussed here.

At higher costs of linking (0.4 - 0.7), we observe interesting features in the data. For the Uniform and Beta distributions, the segregation scores stabilise at under 0.6 for intra-group segregation. Meanwhile, inter-group segregation measures continue to decline. This means that on average, agents of the same type prefer to keep their connections with members of the same type, and choose to economise on their links with agents of the other type to increase their utility as the link cost increases. For the Exponential distribution we observe more interesting behaviour - the intra-group segregation score actually increases between 0.4 and 0.7. Moreover, the inter-group segregation measure declines more rapidly over this range. In effect, the agents are *substituting* their connections with agents of the other type and replacing them with more connections to agents of their own type. This behaviour is driven by the high intolerance of certain agents. As the number of links in the network decreases, intolerant agents might find themselves indirectly connected to agents which they gain disutility from, and not enough agents who generate enough utility to maintain certain connections (as they have been cutting links themselves). Thus, it is rational for such an agent to cut his/her links, and choose instead to connect directly to agents of the same type. Thus intolerance can drive homophily and higher segregation between types.

6.3 Simulation Results: Three Party Model

We further proceed with the same analysis for a three-party model. In this scenario, we have three different types of agents: one with a type of 0, one with a type of 1, and finally one with a type of 0.5. Figure (5) provides a summary of the model parameters used in this set of simulations. We repeat the simulations across the tolerance distributions specified as per Figure 3.

The results of the simulations can be seen in Figures (9) - (11). We briefly choose to explain the findings here, but many are similar to the Two Party Model described earlier.

Stability. As with two types, we observe the same parabola shape for three types with regard to the probability of direct convergence as the cost of linking increases. The observable difference is that the probability of direct convergence peaks at a higher value (roughly between 97.5% and 98.5%) at medium link costs (roughly 0.4 - 0.5). This suggests that the impact of the connection abundance and connection scarcity effects dissipate faster as the cost of linking converges to medium levels; and that the presence of a third type (that has a median type) is beneficial in generating network convergence without reaching cycles.

Number of cliques and degree centrality. The behaviour of the average number of

Model Parameter	Value(s)
Number of iterations	10,000
Decay rate	0.9
Cost shift (for payoff function)	0.2
Number of nodes per type	5
Cost of linking	0.1 - 0.9 (at 0.1 intervals)
Type 1 preferred type	1
Type 2 preferred type	0
Type 3 preferred type	0.5

Figure 5: Three-Party Model Simulation Model Parameters

cliques and average degree centrality is similar to that of the two party state model. Similarly, the variance in both measures decreases as the cost of linking increases. Hence, pairwise stable network architectures share more common statistical features as the cost of linking increases.

Segregation score. For the three party state model we show all the 9 possible segregation scores across all permutations of types. The results are similar across all distributions. As before, at low costs of linking, we observe a reduction in all segregation scores, implying that inter-group and intra-group segregation is increasing (recalling that our score metric is an inverse measure). However, beyond a cost of linking of 0.4 - 0.5, the intra-group segregation score actually increases for *all* distributions to the detriment of inter-group segregation scores. This suggests that the presence of a mediating type might actually lead agents of a particular type to substitute their inter-type connections in favour of intra-type connections at high costs of linking. In other words, a variety of types might not be sufficient to introduce integration into a social network if the cost of linking is high.

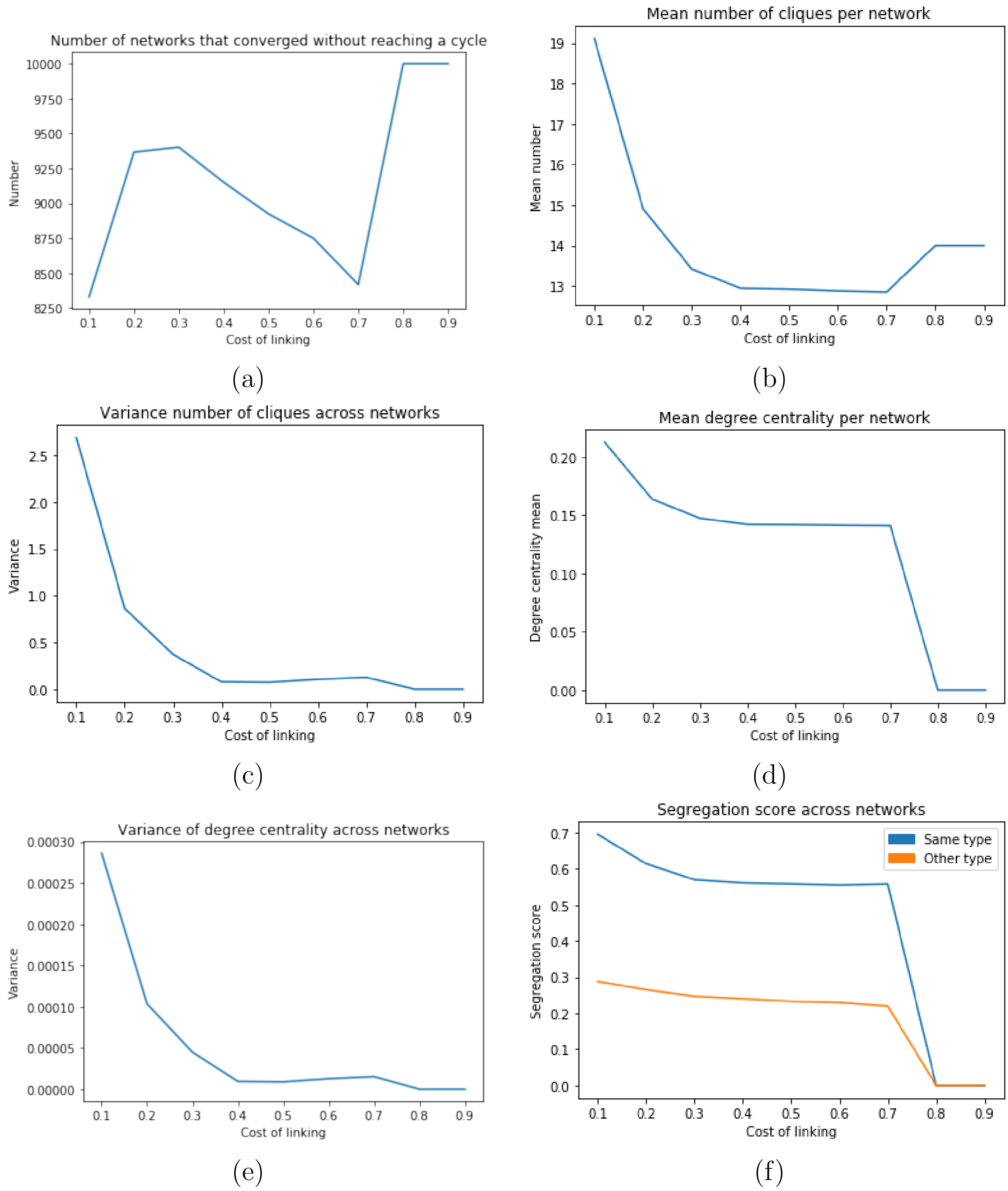


Figure 6: Results for a Two-Party Model with tolerance parameters allocated as per the Uniform distribution over the unit interval. Graphs are: (a) Number of networks that converged to a pairwise stable state without reaching a cycle; (b) Average number of cliques per network; (c) Variance in the number of cliques across networks; (d) Average degree centrality per network; (e) Variance of degree centrality across networks; (f) Average segregation score across networks, delineated by type

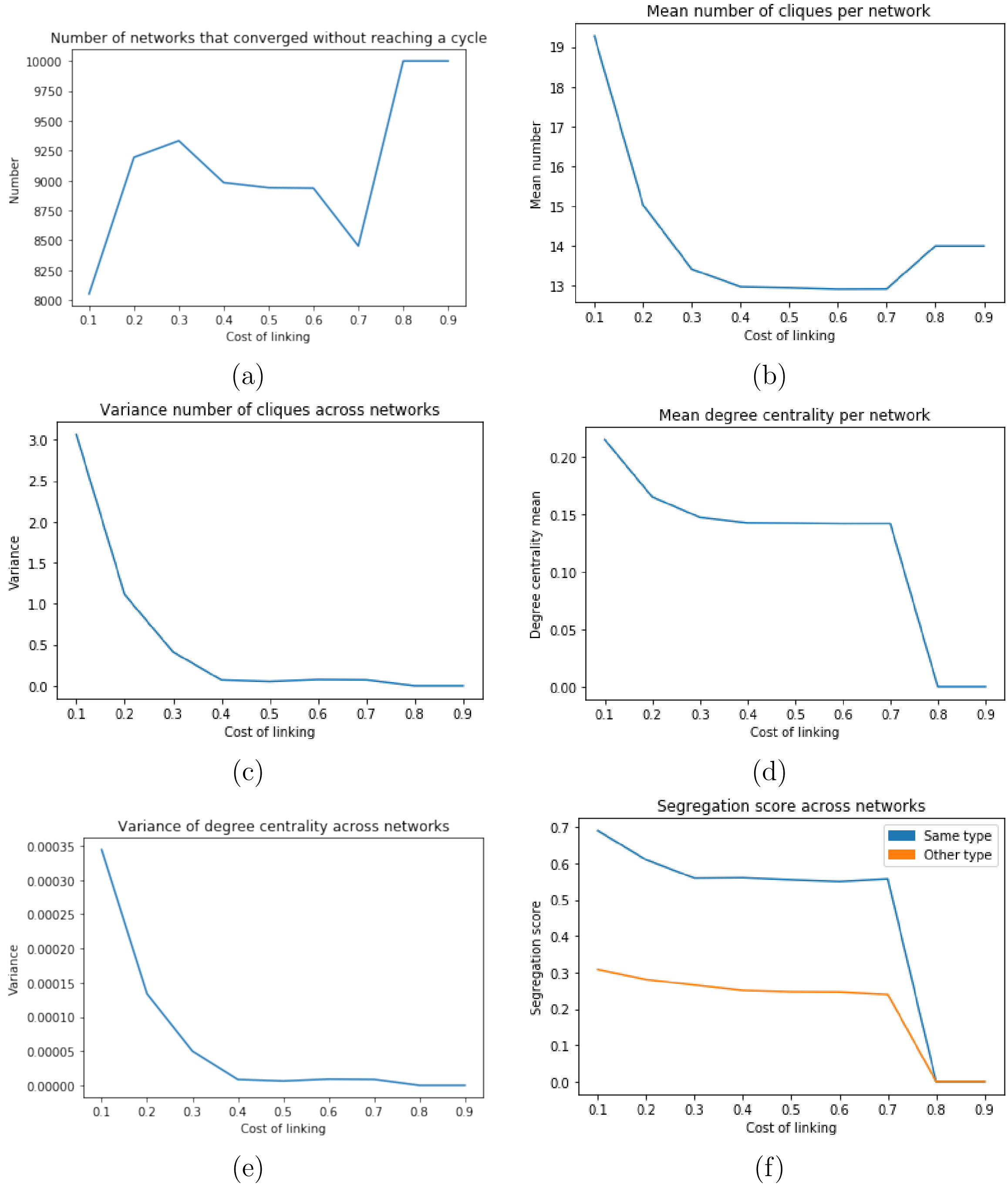


Figure 7: Results for a Two-Party Model with tolerance parameters allocated as per the Beta distribution with both shape parameters set at 0.5. Graphs are: (a) Number of networks that converged to a pairwise stable state without reaching a cycle; (b) Average number of cliques per network; (c) Variance in the number of cliques across networks; (d) Average degree centrality per network; (e) Variance of degree centrality across networks; (f) Average segregation score across networks, delineated by type

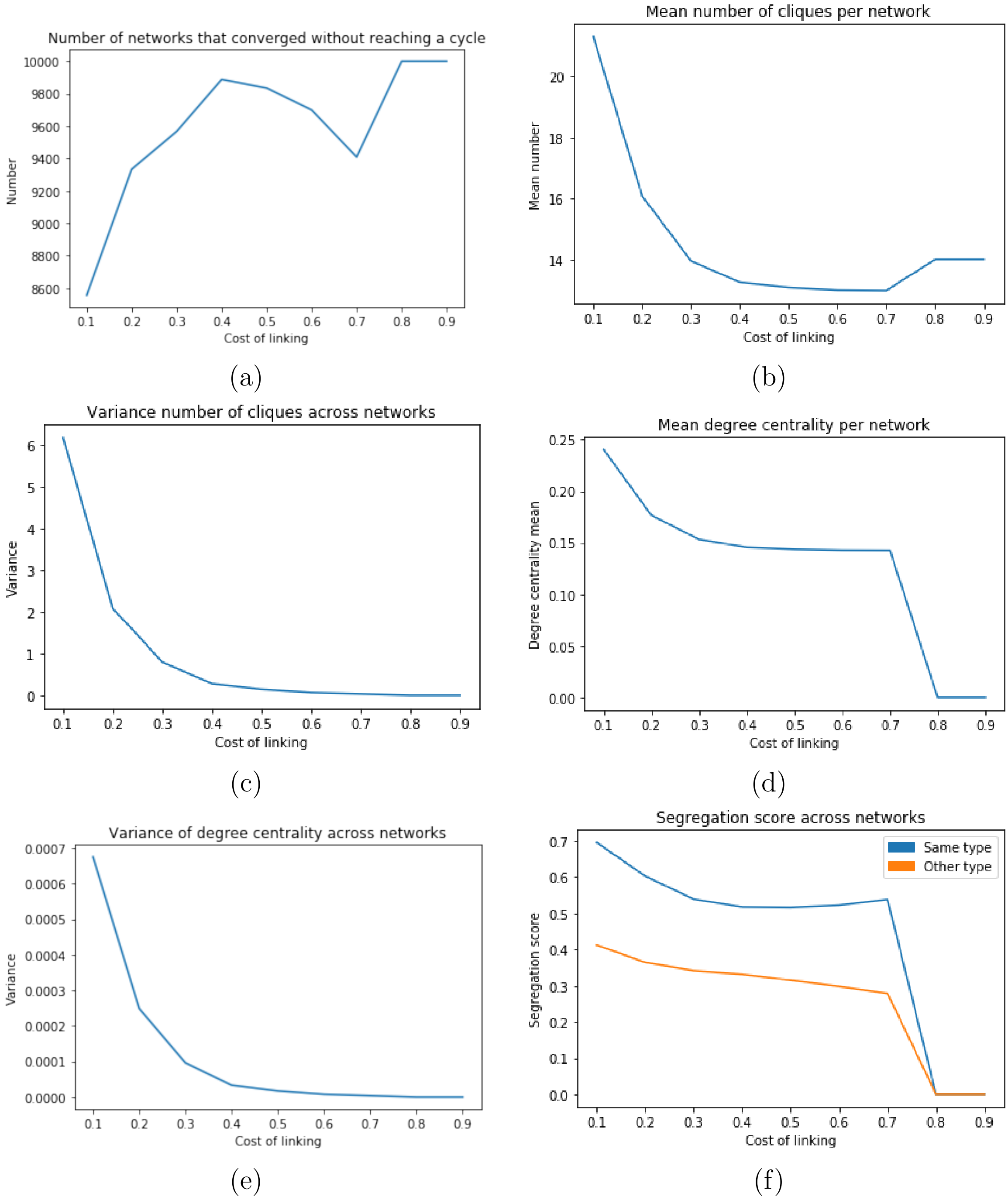


Figure 8: Results for a Two-Party Model with tolerance parameters allocated as per the Exponential distribution with the rate parameter set at 1. Graphs are: (a) Number of networks that converged to a pairwise stable state without reaching a cycle; (b) Average number of cliques per network; (c) Variance in the number of cliques across networks; (d) Average degree centrality per network; (e) Variance of degree centrality across networks; (f) Average segregation score across networks, delineated by type

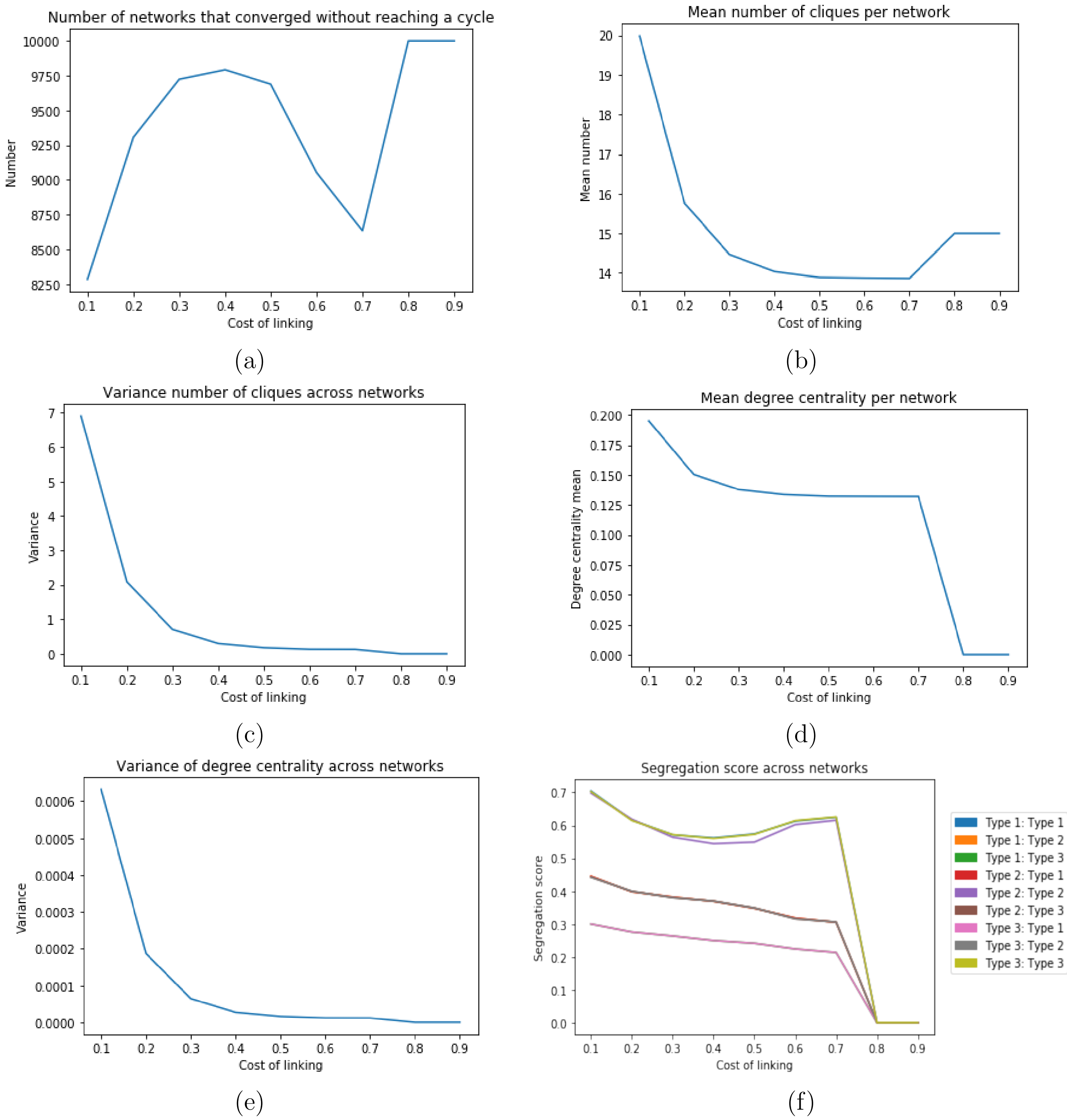


Figure 9: Results for a Three-Party Model with tolerance parameters allocated as per the Uniform distribution over the unit interval. Graphs are: (a) Number of networks that converged to a pairwise stable state without reaching a cycle; (b) Average number of cliques per network; (c) Variance in the number of cliques across networks; (d) Average degree centrality per network; (e) Variance of degree centrality across networks; (f) Average segregation score across networks, delineated by type

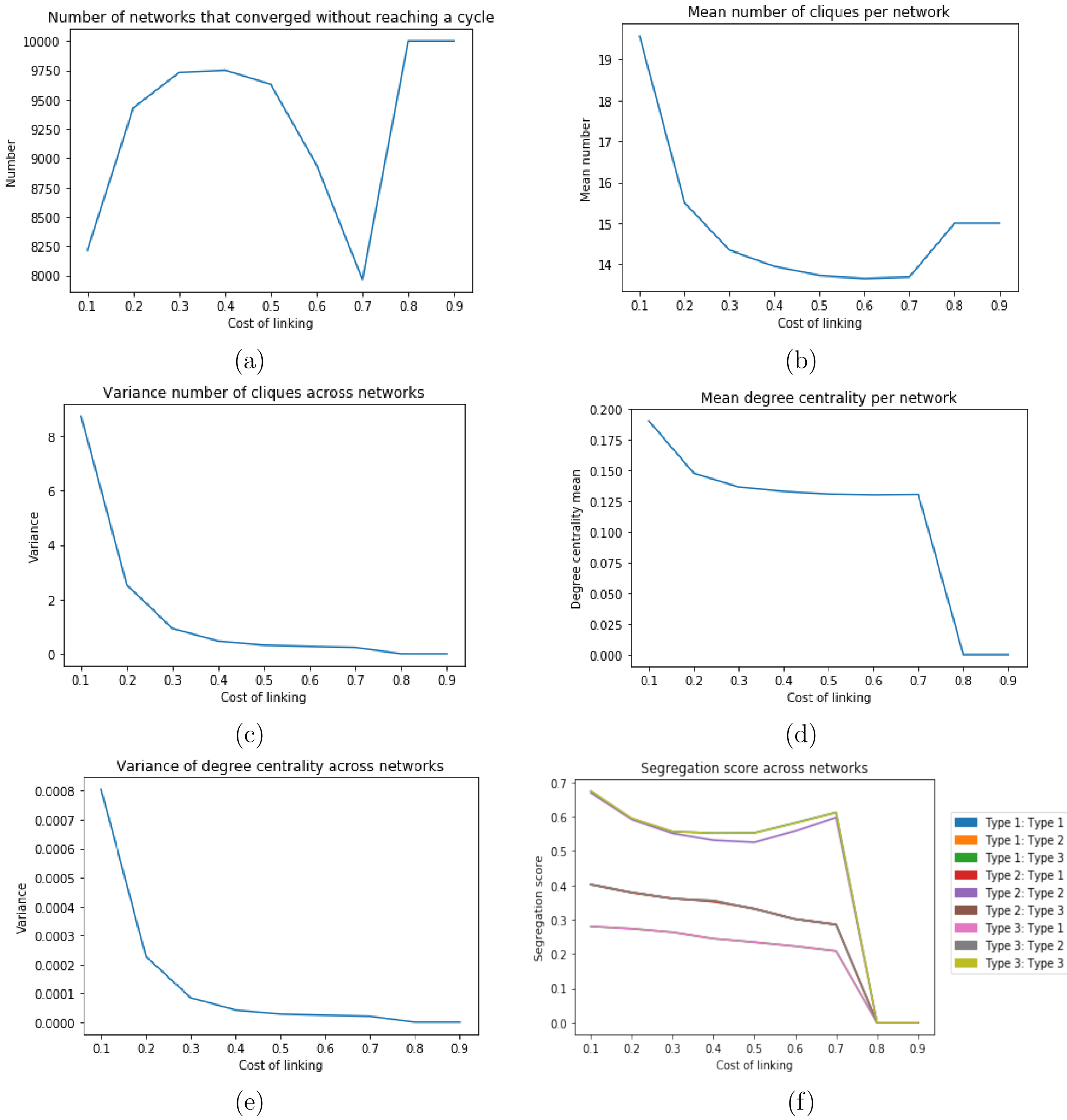


Figure 10: Results for a Three-Party Model with tolerance parameters allocated as per the Beta distribution with both shape parameters set at 0.5. Graphs are: (a) Number of networks that converged to a pairwise stable state without reaching a cycle; (b) Average number of cliques per network; (c) Variance in the number of cliques across networks; (d) Average degree centrality per network; (e) Variance of degree centrality across networks; (f) Average segregation score across networks, delineated by type

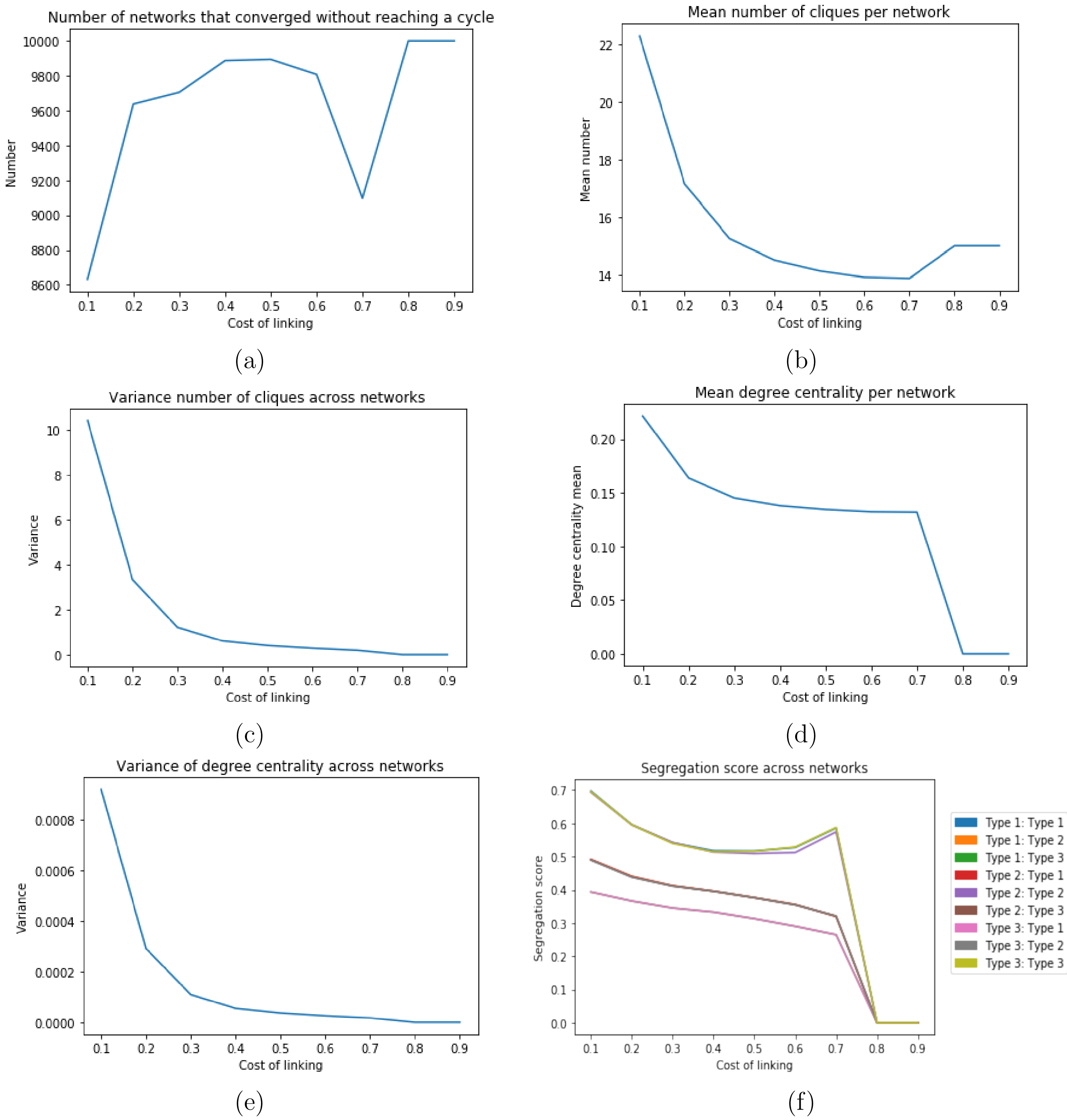


Figure 11: Results for a Three-Party Model with tolerance parameters allocated as per the Exponential distribution with the rate parameter set at 1. Graphs are: (a) Number of networks that converged to a pairwise stable state without reaching a cycle; (b) Average number of cliques per network; (c) Variance in the number of cliques across networks; (d) Average degree centrality per network; (e) Variance of degree centrality across networks; (f) Average segregation score across networks, delineated by type

7 Legal, Social and Ethical Considerations

In this section we briefly discuss any legal and ethical considerations we have taken into account during the process of this research. A completed ethics checklist can be found in the Appendix. The project itself is theoretical in nature. Our approach is purposely agnostic to the dimensions by which segregation can occur, and in doing so, takes no political, religious, socio-economic or otherwise controversial view on any topic. The focus on our approach is how we can use simple economic equilibrium concepts, in a networks setting, to generate segregation, by however dimension. Moreover, we do not hold any data on any other being. Our computational results are simulated from the theoretical framework we have designed. Finally, our work is to be viewed as an attempt to understand how social networks can be formed and how segregation can be generated within them. We do not prescribe policies, in any capacity to be perceived as unethical, in relation to our work. It is to be viewed as an academic study of segregation in social networks only.

8 Conclusion

We have developed a rich and powerful game theoretic model that is capable of generating group segregation, utilising a micro-founded agent framework introduced via the Economics discipline. It is achieved by utilising two kinds of heterogeneity - both across types of agents, but also within the same type of agent through what we define as 'tolerance' of others. Our theoretical model is able to generate both intra-group and inter-group segregation, and accounts for the impact of both direct and indirect connections. In this regard, it is a highly flexible and sophisticated model. Moreover, we extend the model to a dynamic setting, designing our own generative algorithm. Multiple simulations varying the cost of linking uncovers interesting insights. We find that the probability of direct convergence (i.e. a network converging to a pairwise stable state without reaching a cycle) follows a parabola pattern as the cost of linking increases. This suggests that a low cost of linking is not always indicative of a stable network. Moreover, we find a fascinating result - that as the cost of linking increases, pairwise stable networks increasingly share the same statistical characteristics with increasing certainty. This suggests that while we may not be able to predict ex-ante the end state of a social network, depending on the cost of linking, we may be more able to predict its underlying features and characteristics.

There are multiple avenues that the research can extend the findings in this paper. First, the focus on simulations has been on small networks so far. Repeated simulations across large networks with many agents associated with multiple types would be a worthy avenue to explore, to see if the findings found here change with the global size of the network. Such a task would be computationally expensive, so innovations in efficient means to dynamically generate social networks would likely be required as well as more powerful computational resources. Second, theoretically, the model presented here is one of perfect information - the structure of the network as a whole affects an individual's utility. In reality, agents have imperfect information of parts of the network which they are connected to. It would be worthwhile to repeat both the theoretical and computational analysis conducted here under a model of imperfect information to see what changes occur. Finally, this model presents a powerful way to explore how diffusion of ideas and habits propagate through social networks. The model can be extended to a two-stage game: in the first stage agents connect to one another as in this paper; in the second stage, their types can be adjusted depending on who they are connected to, and thus influenced by. In this regard, the framework introduced here has multiple ways of being extended, both theoretically and computationally.

References

[1] Venkatesh Bala and Sanjeev Goyal. "A Noncooperative Model of Network Formation". In: *Econometrica* 68.5 (2000), pp. 1181–1229.

[2] Albert-László Barabási and Réka Albert. "Emergence of scaling in random networks". In: *Science* 286 (1999), pp. 509–512.

[3] Itzhak Benenson and Erez Hatna. "Combining Segregation and Integration: Schelling Model Dynamics for Heterogeneous Population". In: *Journal of Artificial Societies and Social Simulation* 18.4 (2015), p. 15.

[4] Itzhak Benenson and Erez Hatna. "Minority–Majority Relations in the Schelling Model of Residential Dynamics". In: *Geographical Analysis* 43.3 (2011), pp. 287–305.

[5] Itzhak Benenson and Erez Hatna. "The Schelling Model of Ethnic Residential Dynamics: Beyond the integrated segregated dichotomy of patterns". In: *Journal of Artificial Societies and Social Simulation* 15.1 (2012), p. 6.

[6] L. Elisa Celis and Aida Sadat Mousavifar. "A Model for Social Network Formation: Efficiency, Stability and Dynamics". In: *Unpublished* (2016).

[7] Sergio Currarini, Matthew O. Jackson, and Paolo Pin. "An Economic Model of Friendship: Homophily and Segregation". In: *Econometrica* 77.4 (2009), pp. 1003–1045.

[8] Donna Eder and Maureen T. Hallinan. "Sex Differences in Children's Friendships". In: *American Sociological Review* 43.2 (1978), pp. 237–250.

[9] Paul Erdös, Alfréd, and Rényi. "On random graphs". In: *Publicationes Mathematicae, Debrecen* 6 (1959), pp. 290–297.

[10] Paul Erdös, Alfréd, and Rényi. "On the Evolution of Random Graphs". In: *Publication of the Mathematical Institute of the Hungarian Academy of Sciences* 5 (1960), pp. 17–61.

[11] Paul Erdös, Alfréd, and Rényi. "On the strength of connectedness of a random graph". In: *Acta Math. Acad. Sci. Hungar.* 12 (1961), pp. 261–267.

[12] Ove Frank and David Strauss. "Markov Graphs". In: *Journal of the American Statistical Association* 81 (1986), pp. 832–842.

[13] Andrea Galeotti, Sanjeev Goyal, and Jurjen Kamphorst. "Network formation with heterogeneous players". In: *Games and Economic Behaviour* 54.2 (2006), pp. 353–372.

[14] Edoardo Gallo. "Small world networks with segregation patterns and brokers". In: *Unpublished* (2009).

[15] M. Girvan and M. E. Newman. "Community structure in social and biological networks". In: *PNAS* 99.12 (2002), pp. 7821–7826.

[16] Sanjeev Goyal, Marco J. van der Leij, and José Luis Moraga-González. "Economics: An Emerging Small World". In: *Journal of Political Economy* 114.2 (2006), pp. 403–412.

[17] Daniel Hojman and Adam Szeidl. "Core and Periphery in Endogenous Networks". In: *Mimeo, Harvard University* (2003).

[18] Matthew O. Jackson. "The Economics of Social Networks". In: *Unpublished* (2005).

[19] Matthew O. Jackson and Brian W. Rogers. "Meeting Strangers and Friend of Friends: How Random are Social Networks". In: *The American Economic Review* 97.3 (2007), pp. 890–915.

[20] Matthew O. Jackson and Brian W. Rogers. "The economics of small worlds". In: *Journal of the European Economic Association* 3.2-3 (2005), pp. 617–627.

[21] Matthew O. Jackson and Alison Watts. "The Evolution of Economic and Social Networks". In: *Journal of Economic Theory* 106 (2002), pp. 265–295.

[22] Matthew O. Jackson and Asher Wolinsky. "A Strategic Model of Social and Economic Networks". In: *Journal of Economic Theory* 71.0108 (1996), pp. 44–74.

[23] Cathleen Johnson and Robert P. Gilles. "Spatial Social Networks". In: *Review of Economic Design* 5.3 (2000), pp. 273–299.

[24] Matthijs Kalmijn. "Intermarriage and Homogamy: Causes, Patterns, Trends". In: *Annual Review of Sociology* 24 (1998), pp. 395–421.

[25] Paul Lazarsfield and Robert Merton. "Friendship as a social process: a substantive and methodological analysis". In: *Freedom and Control in Modern Society* (1954), pp. 18–66.

[26] Hugh Louch. "Personal network integration: transitivity and homophily in strong-tie relations". In: *Social Networks* 22 (2000), pp. 45–64.

[27] Michael McBride. "Imperfect monitoring in communication networks". In: *Journal of Economic Theory* 126 (2006), pp. 97–119.

[28] Lynn Smith-Lovin Miller McPherson and James M Cook. "Birds of a Feather: Homophily in Social Networks". In: *Annual Review of Sociology* 27 (2001), pp. 415–444.

[29] Derek De Solla Price. "A General Theory of Bibliometric and Other Cumulative Advantage Processes". In: *Journal of the American Society for Information Science* 27 (1976), pp. 292–306.

[30] Thomas C. Schelling. "Dynamic Models of Segregation". In: *Journal of Mathematical Sociology* 1 (1971), pp. 143–186.

[31] Thomas C. Schelling. "Models of Segregation". In: *The American Economic Review* 59.2 (1969), pp. 488–493.

[32] Stanley Wasserman and Katherine Faust. "Logit Models and Logistic Regressions for Social Networks: I. An Introduction to Markov Graphs and P*". In: *Psychometrika* 61 (1996), pp. 401–425.

[33] Alison Watts. "A Dynamic Model of Network Formation". In: *Games and Economic Behavior* 34 (2001), pp. 331–341.

9 Appendix

9.1 Ethics Checklist

Consideration	**Yes/No**
Does your project involve Human Embryonic Stem Cells?	No
Does your project involve the use of human embryos?	No
Does your project involve the use of human foetal tissues / cells?	No
Does your project involve human participants?	No
Does your project involve human cells or tissues? (Other than from "Human Embryos/Foetuses" i.e. Section 1)?	No
Does your project involve personal data collection and/or processing?	No
Does it involve the collection and/or processing of sensitive personal data (e.g. health, sexual lifestyle, ethnicity, political opinion, religious or philosophical conviction)?	No
Does it involve processing of genetic information?	No
Does it involve tracking or observation of participants? It should be noted that this issue is not limited to surveillance or localization data. It also applies to Wan data such as IP address, MACs, cookies etc.	No
Does your project involve further processing of previously collected personal data (secondary use)? For example Does your project involve merging existing data sets?	No
Does your project involve animals?	No
Does your project involve developing countries?	No
If your project involves low and/or lower-middle income countries, are any benefit-sharing actions planned?	No
Could the situation in the country put the individuals taking part in the project at risk?	No
Does your project involve the use of elements that may cause harm to the environment, animals or plants?	No
Does your project deal with endangered fauna and/or flora /protected areas?	No
Does your project involve the use of elements that may cause harm to humans, including project staff?	No
Does your project involve other harmful materials or equipment, e.g. high-powered laser systems?	No
Does your project have the potential for military applications?	No
Does your project have an exclusive civilian application focus?	No

Figure 12: Ethics checklist

Consideration	**Yes/No**
Will your project use or produce goods or information that will require export licenses in accordance with legislation on dual use items?	No
Does your project affect current standards in military ethics – e.g., global ban on weapons of mass destruction, issues of proportionality, discrimination of combatants and accountability in drone and autonomous robotics developments, incendiary or laser weapons?	No
Does your project have the potential for malevolent/criminal/terrorist abuse?	No
Does your project involve information on/or the use of biological-, chemical-, nuclear/radiological-security sensitive materials and explosives, and means of their delivery?	No
Does your project involve the development of technologies or the creation of information that could have severe negative impacts on human rights standards (e.g. privacy, stigmatization, discrimination), if misapplied?	No
Does your project have the potential for terrorist or criminal abuse e.g. infrastructural vulnerability studies, cybersecurity related project?	No
Will your project use or produce software for which there are copyright licensing implications?	No
Will your project use or produce goods or information for which there are data protection, or other legal implications?	No
Are there any other ethics issues that should be taken into consideration?	No

Figure 13: Ethics checklist (cont.)