The impact of topology on diffusion processes in complex networks

Marshall, Christine

2019-10-18

NUI Galway

http://hdl.handle.net/10379/16142

Some rights reserved. For more information, please see the item record link above.
The Impact of Topology on Diffusion Processes in Hyperbolic Networks

Christine Marshall

Supervisors:
Colm O’Riordan and James Cruickshank

A thesis submitted for the degree of Doctor of Philosophy in the School of Computer Science, College of Science and Engineering, National University of Ireland, Galway

September 2019
Acknowledgments

I would like to express my sincere appreciation to my supervisors, Colm O’Riordan and James Cruickshank for their expertise and guidance during this research and I wish to extend my thanks to my Graduate Research Committee: Seamus Hill, Josephine Griffith and Rachel Quinlan for their invaluable advice and support. I also wish to thank my fellow postgraduate students in the Computational Intelligence Research Group and in the School of Mathematics, Statistics and Applied Mathematics for their camaraderie and encouragement. Thanks to Niall Madden for encouraging my participation in broader aspects of academic life in NUIG; this has been enjoyable, extremely rewarding and much appreciated. A special word of thanks to Tina Earls for helping me navigate my way through the system and to Mary Hardiman for her encouragement, and always being available with friendly advice and moral support.
In this thesis we explore how the structure of a network impacts upon spreading behaviours on that network. We have created a dataset of 2,400 hyperbolic random geometric graphs, of increasing edge density from 0 to 1, and analysed local and global graph properties in the set, with a view to correlating graph properties with emergent behaviours in the set of graphs. These networks share many features of real world complex networks, such as power law degree distribution, high clustering and short path length.

A key focus is on observing the impact that the net result of all the local node interactions will have on the emergent behaviour in a population. When an individual node in a population is viewed in isolation, it is extremely difficult to forecast the likely global effect this one node might have on a spreading process. In spatial networks, each node is an integral part of a neighbourhood, defined by its links with its neighbours. This community structure can have a profound effect on the outcome of a spreading process, such as the spread of a disease, or the adoption of an activity within a population.

Modelling such networks also provides opportunities to control the outcomes, for example looking at measures to control the spread of a contagion by identifying influential nodes. In a computer network hub nodes might be selected to receive enhanced virus protection, or key players in a population might be immunised in an epidemic scenario. Governments might seek to identify influential nodes to enhance the spread of public information and warnings in emergency situations.

Our experiments use agent based simulation of diffusion processes on the set of graphs. We observe the evolution of defection in the simple game of the Prisoner’s Dilemma, and the spread of an activity by bootstrap percolation. In both processes, we observed a clearly defined state transition threshold as the number of edges increases in the set of graphs. Above the threshold, the activity completely spread, below the threshold, the network was robust against the spread. Our analysis of graph properties allows us to correlate emergent behaviour with topological features in the graphs. This has allowed insights into features of the graphs which inhibit or facilitate the spread of an activity.
As an extension to this work we developed a modified form of the standard bootstrap percolation process, which allows for recovery. This introduces a stochastic element to an otherwise deterministic process. This had a delaying impact on the spread of an activity, which was enhanced when targeted at nodes of high degree, compared with at random.

In our final set of simulations, our goal is to enhance this inhibitory effect on percolation, returning to the standard form of bootstrap percolation, by specific targeting of nodes to be immune to the percolation process. Instead of allowing a percentage of active nodes to recover, we investigate outcomes when certain nodes are chosen to be unaffected by the percolation process, essentially granting them immunity from percolation. Our selected graph properties were high degree, betweenness and closeness centrality, low local clustering, and random for comparison.

This work has demonstrated that identifying influential nodes and targeting them for immunity has an inhibitory effect on the bootstrap percolation process on hyperbolic networks, when compared with random immunity. This indicates that in these graphs certain nodes are highly influential in the network and warrant being protected. Our results suggest that under similar conditions it may be possible to immunise a relatively small number of influential nodes and effectively grant herd immunity to the whole network.
# Contents

1 Introduction 19

1.1 Methods and motivations ....................................... 20
  1.1.1 Agent Based Simulation on Spatial Networks ............... 21
  1.1.2 Random Graph Models ....................................... 21
  1.1.3 Dynamic Processes ......................................... 23

1.2 Open Research Questions ........................................ 25
  1.2.1 Random graph models ....................................... 25
  1.2.2 Diffusion Processes on Random Geometric Graphs ........... 26

1.3 Research Questions ............................................. 27

1.4 Hypotheses ....................................................... 28

1.5 Contribution ...................................................... 29

1.6 Thesis Overview .................................................. 30
  1.6.1 Chapter 2 Background ........................................ 31
  1.6.2 Chapter 3 Hyperbolic Random Geometric Graphs .......... 31
  1.6.3 Chapter 4 The Prisoner's Dilemma .......................... 31
  1.6.4 Chapter 5 Bootstrap Percolation ............................ 31
  1.6.5 Chapter 6 Bootstrap Percolation with Recovery ............ 32
  1.6.6 Chapter 7 Bootstrap Percolation with Inhibitory Nodes .... 32
  1.6.7 Chapter 8 Conclusion ....................................... 33

2 Background and Main Concepts 35

2.1 Network Theoretic Concepts ..................................... 35
  2.1.1 Network ....................................................... 36
2.1.2 Graph .............................................. 36
2.1.3 Paths .............................................. 36
2.1.4 Component ........................................... 37
2.1.5 Connected Graph ................................. 37
2.1.6 Node Degree .................................... 37
2.1.7 Density and Average Degree .................... 37
2.1.8 Skewness ........................................... 38
2.1.9 Degree Distribution ............................... 38
2.1.10 Clustering Coefficients ......................... 42
2.1.11 Centrality ....................................... 43

2.2 Theoretical Models of Network Structure .......... 47
2.2.1 Random Graph Models ............................. 48
2.2.2 Erdős Rényi model of Random Graph .......... 48
2.2.3 Small World Graphs ............................... 50
2.2.4 Scale Free Models ................................ 50
2.2.5 Random Geometric Graphs ....................... 51
2.2.6 Hyperbolic Geometry .............................. 52
2.2.7 Hyperbolic Random Geometric Graphs ........ 54

2.3 Properties of Real World Networks .................. 56

2.4 Diffusion Processes on Networks .................... 56
2.4.1 Game Theory ..................................... 57
2.4.2 Bootstrap Percolation ............................. 61
2.4.3 Epidemic Modelling ............................... 65
2.4.4 Influential Nodes in Networks .................. 67

3 Hyperbolic Random Geometric Graphs: Creation and Analysis 71
3.1 Introduction ............................................ 71
3.2 Graph Creation ....................................... 72
3.2.1 Point distribution ................................. 72
3.2.2 Edge creation ..................................... 72
3.2.3 Implementation ........................................... 73
3.3 Graph Analysis .............................................. 74
  3.3.1 Mapping $R$ to edge density ......................... 74
  3.3.2 Global graph properties ............................. 75
3.4 Discussion .................................................. 78

4 Agent based modelling of The Prisoner’s Dilemma ............ 81
  4.1 Introduction ............................................... 81
  4.2 Research Questions and Hypotheses ..................... 82
    4.2.1 Research Questions .................................. 82
    4.2.2 Hypotheses .......................................... 83
  4.3 Experimental Design ..................................... 83
    4.3.1 Simulation of the Prisoner’s Dilemma ............. 83
  4.4 Results .................................................. 84
    4.4.1 Game-Theoretic Transition Threshold ............... 84
    4.4.2 Graph Properties at Threshold ..................... 85
  4.5 Discussion .............................................. 89

5 Bootstrap Percolation on Hyperbolic Random Geometric Graphs 93
  5.1 Introduction ............................................... 93
  5.2 Research Questions and Hypotheses ..................... 94
    5.2.1 Research Questions .................................. 95
    5.2.2 Hypotheses .......................................... 96
  5.3 Experimental Design ..................................... 96
  5.4 Results .................................................. 98
    5.4.1 Graph Properties at Threshold ..................... 99
  5.5 Discussion .............................................. 104

6 Bootstrap Percolation with Recovery ......................... 107
  6.1 Introduction ............................................... 107
  6.2 Research Questions and Hypotheses ..................... 110
6.2.1 Research Questions .............................................. 110
6.2.2 Hypotheses .......................................................... 111
6.3 Process Definition for Bootstrap Percolation with Recovery .... 111
6.4 Experimental Simulation of Bootstrap Percolation with Recovery .. 111
6.4.1 Random recovery ..................................................... 113
6.4.2 Targeted recovery based on node degree ranking ............... 113
6.5 Results ................................................................. 114
6.5.1 Bootstrap Percolation with Recovery .............................. 114
6.6 Discussion ............................................................... 119

7 Bootstrap Percolation with Immunisation ............................ 123
7.1 Introduction ............................................................. 123
7.1.1 Immunisation and Influential Nodes .............................. 126
7.2 Research Questions and Hypotheses ................................. 126
7.3 Framework for Identifying Influential Nodes to Inhibit Bootstrap Percolation on Hyperbolic Networks ...................... 127
7.4 Experimental Set Up ................................................... 128
7.4.1 Simulation of Bootstrap Percolation ............................... 130
7.4.2 Simulation of Bootstrap Percolation with Immunisation ....... 130
7.5 Results ................................................................. 132
7.5.1 Upper boundary of percolation threshold for activation threshold of 10 ..................................................... 132
7.5.2 Upper boundary of percolation threshold for activation threshold 2 .............................................................. 137
7.5.3 Graph properties at the upper boundary of the percolation threshold, for activation thresholds of 2 and 10 .............. 141
7.5.4 Discussion ............................................................. 142
7.6 Conclusion ............................................................... 143

8 Conclusion ............................................................... 145
8.1 The Prisoner’s Dilemma ............................................... 146
8.2 Bootstrap Percolation ............................................. 147
8.3 Bootstrap Percolation with Recovery ......................... 147
8.4 Bootstrap Percolation with Immunity ....................... 147
8.5 Contribution ....................................................... 148
8.6 Application to Pandemic Scenarios ......................... 150
8.7 Future Work ......................................................... 153
List of Figures

2-1 Negative and positive skew diagrams by Rodolfo Hermans is licensed under CC BY-SA 3.0 ................................................................. 38
2-2 Binomial Distribution with probability $p = 0.5, 0.1$ and $0.7$ [69] . . . 40
2-3 Poisson Distribution [1] ................................................................. 40
2-4 Normal Distribution [105] .............................................................. 41
2-5 PowerLawDistribution [19] ............................................................. 42
2-6 Graph displaying normalised degree centrality values. .................... 44
2-7 Graph displaying non-normalised betweenness centrality values. ......... 45
2-8 Graph displaying normalised closeness centrality values. ............... 46
2-9 Erdős Rényi model, $G(n, p)$, with $n = 10, p = 0.6$ .................... 49
2-10 Spaces of positive, negative and flat curvature [114] .................... 52
2-11 Types of parallelism in the Poincaré disc. Lines $m$ and $\lambda$ share an ideal point and are parallel. Lines $m$ and $n$ are hyperparallels to line $\lambda$ [44]. 53
2-12 Claudio Rocchini Order-3 heptakis heptagonal tiling 2007, distributed under a CC BY 2.5 licence [109] ................................. 54

3-1 Distance parameter $R$ mapped to density ................................... 75
3-2 Clustering coefficients mapped to distance parameter $R$ ............ 77
3-3 Graph centralisation measures mapped to distance parameter $R$ .... 77
3-4 Average degree mapped to distance parameter $R$ ...................... 78
3-5 Path length mapped to distance parameter $R$ ......................... 79
4-1 Heatmap showing Prisoner's Dilemma outcomes on hyperbolic graphs created with varying average degree. Nodes sorted by outcome in descending order of cooperators ........................................... 85
4-2 Size of giant component at varying distance parameter $R$ ............. 86
4-3 Edge density at varying distance parameter $R$ ............................... 87
4-4 Clustering coefficients, values in the range 0 - 1. ........................... 87
4-5 Diameter and average shortest path at varying average degree ....... 88
4-6 Degree, betweenness and closeness centralisation ........................... 89

5-1 Heat map for standard bootstrap percolation showing number of final active nodes at equilibrium. $A_o = 20$, with $AT$ from 2 to 10 ......... 98
5-2 Size of giant component at varying distance parameter $R$ ............. 100
5-3 Edge density at varying distance parameter $R$ ............................... 101
5-4 Clustering coefficients, values in the range 0 - 1. ........................... 102
5-5 Diameter and average shortest path ................................................. 103
5-6 Degree, betweenness and closeness centralisation ........................... 103

6-1 Bootstrap percolation with random recovery ................................. 115
6-2 Bootstrap percolation with targeted recovery ................................. 116
6-3 Percolation threshold values for random and targeted recovery at activation thresholds (AT) 2 to 7, with varying recovery rates from 10 to 90% ................................................................. 117
6-4 Percolation threshold values for random and targeted recovery at activation thresholds (AT) 8 to 10, with varying recovery rates from 10 to 90% ................................................................. 118

7-1 Conceptual Framework for Identifying Influential Nodes to Inhibit Bootstrap Percolation on Hyperbolic Networks ................................. 128
7-2 Heat map for standard bootstrap percolation showing number of final active nodes at equilibrium. $A_o = 20$, with activation thresholds from 2 to 10. Figure 5-1 reprinted here for ease of viewing. ................................. 129
7-3 Graph 5.713. Comparison of outcomes for Immunity, with 1000 simulations at each activation threshold ............... 133
7-4 Graph 5.713. Comparison of outcomes for Immunity, with 1000 simulations at each activation threshold ............... 134
7-5 Graph 5.713. Comparison of outcomes for Immunity, with 1000 simulations at each activation threshold ............... 135
7-6 Rate of decline in percolating simulations as number of immunised nodes increases from 0 to 25 (immunity applied to top ranked nodes of highest degree) ......................... 136
7-7 Graph 3.013. Comparison of outcomes for standard bootstrap percolation, random immunity and immunity targeted at nodes of high degree, with 1000 simulations at each activation threshold ......................... 138
7-8 Graph 3.013. Comparison of percolation outcomes for immunity targeted at nodes of high degree, high betweenness and high closeness, with 1000 simulations at each activation threshold ......................... 139
7-9 Graph 3.013. Comparison of percolation outcomes for immunity targeted at nodes of high and low clustering, with 1000 simulations at each activation threshold ......................... 140
7-10 Hyperbolic Graph Centralisation compared with percolation thresholds. .................................................. 142
List of Tables

2.1 Generalised Payoff Matrix for Prisoner’s Dilemma Game, where $s < p < r < t$ .................................................. 58

4.1 Payoff Matrix for Prisoner’s Dilemma Game ........................................ 84

7.1 Global graph properties at the upper boundary of the percolation threshold ......................................................... 141
Chapter 1

Introduction

In recent times there has been much interest in activities on real world networks, for example concerns about the spread of fake news in online social media and the potential effect on voting behaviour in populations [3]. Other examples include the spread of viruses in computer networks, such as the 2017 WannaCry cyber attack which had a major impact on the UK’s National Health Service [89].

Social networks and communication networks are examples of complex systems, consisting of many components and the interactions between them. Modelling networks and activities on those networks allows us to gain insights into those systems in a simplified form. Researchers are interested in modelling the underlying networks to understand how these activities spread, and how to facilitate or prevent the spread. Marketing managers are interested in finding ways to spread their brand message in a social network by identifying social media influencers [49]. In biological systems researchers might model the spread of an activity to observe, and potentially mitigate, critical points at which a population might be under threat. In evolutionary game theory, which models strategic decision making between individuals or groups, the interested parties might make decisions based on common interest or perhaps self interest will win out when there is a temptation to defect from the common good. What all these have in common is that they can be modelled as spreading processes on networks. The links between nodes are the communication links and the activity spreads through interactions between connected players.
A key focus is on observing the impact that the net result of all the local interactions will have on the emergent behaviour in a population. When an individual node in a population is viewed in isolation, it is extremely difficult to forecast the likely global effect this one node might have on a spreading process. In spatial networks, each node is an integral part of a neighbourhood, defined by its links with its neighbours. This community structure can have a profound effect on the outcome of a spreading process, such as the spread of a disease, or the adoption of an activity within a population.

Modelling such networks also provides opportunities to control the outcomes, for example looking at measures to control the spread of a contagion by identifying influential nodes. In a computer network hub nodes might be selected to receive enhanced virus protection, or key players in a population such as teachers or medical staff might be immunised in an epidemic scenario. Governments might seek to identify influential nodes to enhance the spread of public information and warnings in emergency situations.

1.1 Methods and motivations

The main approaches to analysing complex networks involve:

- statistical methods, using statistical inference on sample data to gain insights into typical behaviour
- mathematical analysis, which uses techniques, such as differential equations, to solve the system
- empirical data analysis involving observations of real world complex networks
- numerical simulations, which implement mathematical models that are too complex for mathematical analysis
- agent based simulations, which use computational intelligence to model the interactions of autonomous agents.
1.1.1 Agent Based Simulation on Spatial Networks

Our approach uses agent based simulation on spatial networks, which models a population as a set of autonomous agents, observing how interactions between these agents will impact on emergent behaviour in the population.

Other approaches, such as equation based models or statistical models, frequently use a mean field approach, which gathers probabilities based, for example, on grouping together like nodes; the interactions for each node are therefore approximated by the average effect in the whole system. An advantage of this method is that it can be reduced to one equation for the whole system, which is analytically tractable. However, this generalisation loses the detail for the impact of individual nodes and their community structure on the outcome of the process.

An advantage of agent based simulation is that you can directly observe the role of an individual node on the emergent behaviour, and can see this node in relation to its local neighbourhood. This is useful as we can identify individual nodes which are influential in the process, which creates a potential opportunity for targeting these nodes in some way to modify process outcomes. This might include a control strategy to inhibit the spread of an activity by the effective quarantine or immunisation of a node that is deemed influential in this behaviour.

The main disadvantage of this method is that it is computationally expensive to run simulations. Agents change state at each time step according to a pre-determined state transition rule. This involves identifying the state of each node, and that of its neighbours, many times in a simulation. However, this disadvantage is offset by techniques such as optimisation, increased memory resources, parallel processing and efficient algorithms.

1.1.2 Random Graph Models

The choice of a random graph model arose as it is easier to create and control the parameters in synthetic models than in real world networks. Using random graph models also ensures a level of stochasticity in the resultant graphs, since any graph
created is a sample from all possible graphs within this distribution. Using the graph creation parameters, we can easily create many graphs with similar graph properties for our simulations, which helps validate the results. This may not be possible with real world networks. We can also easily modify our graphs by altering the graph creation parameters. For example, in our case we are interested in the effect of increasing or decreasing the number of edges in the graphs; in a real world network it is difficult to find a consistent approach to this, whereas by using our graph creation algorithm we can create an ensemble of graphs with increasing or decreasing edge density. Our graph analysis shows that at each graph creation parameter, the resultant graphs have very similar graph properties, the variation arises from the random positioning of each node, with respect to a uniform distribution, and the resultant connections to nodes lying within a specified distance of each other. This similarity in graph properties allows for direct comparison over all such graphs created at this parameter.

Having a ready-made data set of 2,400 graphs, already analysed for graph properties, allows us to perform modifications to our simulations, and affords direct comparison with our standard simulations run on these same graphs. This demonstrates the utility of having an existing dataset of graphs, each sampling from a probability distribution, with increasing edge density from 0 to 1. In terms of fundamental research, this has advantages over using one real world network as you can produce many different graphs at the same edge density, all sampling from the same probability distribution, and with similar graph properties. One key feature of fundamental research is that insights gained may be useful in a broad range of applications. For example, observations from simulations on these random graphs may be useful for a variety of different real world networks.

Our choice of hyperbolic random geometric graphs as a model for complex networks arose because they are a recently developed graph model which typically display power law degree distribution, short path lengths and clustering, which are features shared with many real world complex networks [42, 36]. Modelling processes on these networks might therefore allow insights into real world networks. For example,
Krioukov et al. mapped the Internet Graph to an underlying hyperbolic geometry [55] which suggests that these graphs might be appropriate for studying processes on similar networks.

There is little research on game theory on these graphs, most of the research on the game of Prisoner’s Dilemma has been on lattice grids. Other research on spreading behaviours, such as bootstrap percolation, on hyperbolic networks tends to concentrate on statistical mechanics and probabilistic analysis. This is a definite lack, as these graphs share features with many real world complex networks.

1.1.3 Dynamic Processes

One of the key features from network science is that research in one field may offer insights into another field. For example, the insights gained in the spread of cooperation in a simple strategy game may offer insights into evolutionary behaviours in ecosystems. Equally such insights can be applied to diverse fields such as the use of gossip algorithms in data routing in wireless networks. Additionally, modelling different spreading phenomena on the same network may offer insights into local node interactions and how they play out under different sets of rules.

The processes we have chosen to investigate on the set of hyperbolic graphs are the evolution of defection in the game of the Prisoner’s Dilemma, and the spread of an activity by Bootstrap Percolation. The spatial and evolutionary form of the Prisoner’s Dilemma models strategic decision making between intelligent rational actors in a network, where potential interactions receive scores based on a payoff matrix; the process evolves over time by agents adopting the strategies of their best scoring neighbours. Bootstrap percolation is a different kind of process, which can model social influence, whereby an activity which is popular with the neighbours of a node is likely to be adopted by that node, the activity spreads once the number of users reaches a tipping point.

Our motivation for choosing to model the Prisoner’s Dilemma on hyperbolic random graphs arose naturally out of our previous research simulating evolutionary game theory on three different random graph models [70]. That research concen-
trated on comparing outcomes on three graph models of equal edge density, where all had unique edge distribution throughout the graph. This research seeks to explore emergent behaviour on a larger data set of hyperbolic graphs, of increasing edge density from 0 to 1.

Our motivation for choosing to explore simulations of Bootstrap percolation on the same set of graphs arose from some similarities between the two processes, with some key differences.

The standard form of the Prisoner's Dilemma is a binary state process, where agents can either cooperate or defect. In the evolutionary form, we observe the emergent strategies over time. Bootstrap percolation is also a binary state process, where agents can be inactive or active. However, a key feature in Prisoner’s Dilemma is that agents can change state from inactive to active to inactive on numerous occasions. Whereas Bootstrap Percolation allows only uni directional state change from inactive to active, no reverse state change is possible. The state transition mechanism for both processes is also different. In Prisoner’s Dilemma, strategies evolve over time dependent on the adoption of the best scoring strategy in a neighbourhood. In bootstrap percolation, an activity spreads when the number of active agents reaches a tipping point.

Our motivation for choosing to modify the Bootstrap Percolation process arose from observing significant features of the Prisoner’s Dilemma game. We speculated that introducing features from one process to the other might enrich our knowledge of spreading behaviours on networks. This modified approach to bootstrap percolation introduces a recovery element, which allows for reverse state change from active to inactive. Our goal was to observe if this would impact upon the percolation process. Following on from this modification to the process, we further speculated on the likely impact of selectively removing nodes from the percolation process by effectively immunising them.
1.2 Open Research Questions

1.2.1 Random graph models

The Erdős Rényi random graph model was developed in 1959 [34]. In this model, edges between pairs of nodes are created with probability $p$. As the number of nodes increases, the initial binomial degree distribution approaches a normal distribution, allowing for an increasing regularity of node properties within the graph.

Subsequently, models were developed which allowed for distinct variations in graph properties. In 1998, Duncan Watts and Steven Strogatz took a simple ring lattice and rewired each edge with a defined probability, essentially creating short cuts across the graph. This allowed most nodes in the graph to be reached in a small number of steps, the so called “small world” effect [134]. In 1999, Lazlo Barabási and Reka Albert noticed that real world networks have an underlying “scale free” structure, displaying power law degree distribution. They developed an algorithm to create a Scale Free model, where each new node has edges created using preferential attachment [14]. This modelled the “rich get richer” phenomenon and created graphs with power law degree distribution. Much of the interest since then has been on how the variations in graph topology, arising from different patterns of edge distribution, have affected pathways between nodes within a network.

In the 1970s, random geometric graphs, or distance graphs, were developed where the distance between nodes determines whether a connection exists or not. These have been used to model situations such as the spread of a forest fire from tree to adjacent tree or to determine connectivity in sensor networks. These graphs have been rigorously analysed for graph properties such as degree distribution, component size and connectivity [99, 101, 100]. These graphs have conventionally been created in the Euclidean unit square and unit disc. However, in 2010 Krioukov et al. investigated a hyperbolic random geometric graph, where points are distributed in a hyperbolic space and edges created based on the hyperbolic distance between them [55]. The resultant graph topology is distinctly different to the Euclidean model, displaying power law degree distribution, high clustering and short average path length [42, 23].
The reasons for this are explored in Chapter 2.2.6 and 2.2.7. These are features shared with many real world networks, which suggests that research on these graphs might offer insights into diffusion processes on complex networks.

### 1.2.2 Diffusion Processes on Random Geometric Graphs

The standard form of spatial Prisoner’s Dilemma is a binary state process where agents can either cooperate or defect, and change of state occurs according to the expected payoff from different interactions, or the relative payoff of interactions with other connected nodes. In this game, the best mutual strategy is for both players to cooperate however, without knowledge of their opponent’s strategy, there is a temptation to defect. In fact, defection is the dominant strategy in this game and work has been undertaken to examine scenarios where cooperation might be encouraged to persist, for example in graphs such as small world and scale free networks [93, 113, 132, 102, 103, 110, 135]. Most of the research in evolutionary game theory on random geometric graphs has concentrated on the Euclidean model, particularly in n-dimensional lattice grids [88, 86, 87] and in rewired variants of these grids [92, 91]. To our knowledge there has been no work exploring game theory on hyperbolic random geometric graphs, and this is a definite lack as these graphs share features with real world networks, such as high clustering, short path lengths and power law degree distribution. In particular, our own approach to this investigation is to observe the effect of increasing or decreasing edge density on these graphs.

Bootstrap percolation is also a binary state process, where inactive nodes become active once the number of active neighbours reaches a tipping point. However, in contrast to the Prisoner’s Dilemma, the direction of change is from inactive to active only.

Much of the focus in research on bootstrap percolation has been on identifying the size of the initial seed set or the choice of nodes within this set to ensure the spread of an activity, see for example [133, 9, 15, 48, 29]. Node degree is often seen as the most influential property, witness the perceived importance of hub nodes in a network [5], also sometimes known as super spreaders. Most of the research
in bootstrap percolation has been performed in the field of statistical mechanics, using probabilistic methods. Candellero et al. undertook a mathematical analysis of bootstrap percolation in hyperbolic random geometric graphs and found that the activity almost surely either spread completely or failed to spread [24]. One of the distinctive features of their analysis is that it adopts a mean field approach, grouping sets of nodes with similar properties together to determine the likely outcome of a spreading process. There has been no agent based simulation of bootstrap percolation on hyperbolic random geometric graphs. The advantage of this method is that you can identify the effects that an individual node has on the emergent behaviour in the population. This leads naturally to the identification of influential nodes in a network and their potential usefulness in control strategies to alter the outcome of a process, by either facilitating or inhibiting the spread of an activity. It is an open research area to identify nodes which will inhibit bootstrap percolation, given a seed set selected at random. There has been work on immunising nodes, using the SIS model of epidemic spread which has identified node degree as the most important feature to ensure a successful immunisation strategy [98]. Our own approach is aimed at investigating different measures of influence to select for immunity, using agent based simulation of bootstrap percolation.

1.3 Research Questions

- In the spatial form of the Prisoner’s Dilemma on hyperbolic random geometric graphs, if we take a network of unconnected nodes and increase the edge density from 0 to 1, can we determine a transition threshold between the robustness of cooperation and the dominance of defection?

- In Bootstrap Percolation on the same set of graphs, is it possible to identify a percolation threshold as edge density increases?

- Are there any significant graph properties at these thresholds?

- If we modify the rules of bootstrap percolation to allow reverse state change, will
this have any impact on percolation outcomes and on the percolation threshold? If we target recovery at nodes of high degree, will this have a greater inhibitory effect than random selection of active nodes for recovery?

- Can significant graph properties at the threshold be utilised to introduce control strategies aimed at preventing percolation, for example by introducing immune nodes into the graph? If we select our immune set based on graph properties which display skewed distributions in the hyperbolic graphs, will this have a greater inhibitory effect on percolation, when compared with random selection?

1.4 Hypotheses

- In the game of the Prisoner’s Dilemma, it is possible to identify a clear transition threshold for edge density, which marks the boundary between the robustness of cooperation and the dominance of defection.

- In Bootstrap Percolation, it is possible to identify a clear transition threshold for edge density, which marks the boundary for percolation.

- The graphs at these thresholds display a clear pattern of global graph properties which might be used to identify nodes which are influential in the spreading process.

- Allowing reverse state change in bootstrap percolation will inhibit percolation and raise the percolation threshold to graphs of higher edge density. Targeting active nodes of high degree for recovery will have greater inhibitory effect on the percolation process than random selection.

- Targeting nodes with high centrality measures to become immune to the percolation process will have greater effect than random selection of immune nodes. Targeting nodes that are top ranked for properties that have highly skewed distributions within each graph will maximise the inhibitory effect.
1.5 Contribution

- **Graph Data Set**

  We have created a synthetic data set of 2,400 hyperbolic random geometric graphs with increasing edge density from 0 to 1, and have undertaken an analysis of graph properties in each graph, described in Chapter 3. This data set facilitates our observations of the effect of graph topology on diffusion processes, as we can directly compare different activities on the same set of graphs. Additionally, we have observed graph properties for each of the graphs and can associate these properties with different outcomes. This analysis has been used to correlate graph properties with emergent behaviour simulated on the set of graphs, published in [71, 73, 72].

- **Simulation of Diffusion Processes on Hyperbolic Networks**

  We have used agent based simulation of the Prisoner’s Dilemma on the set of hyperbolic graphs and compared outcomes with graph properties, described in Chapter 4. This work has been published in [71]. We have performed standard bootstrap percolation on the same set of graphs, described in Chapter 5. Our analysis of global graph properties at the transition threshold in both sets of simulations allows us some insight into features which facilitate or obstruct either spreading process. Our simulations of bootstrap percolation complement the research in the area, which has mainly concentrated on mathematical analysis and probabilistic methods.

- **Introduced modified form of Bootstrap Percolation**

  We have developed a recovery mechanism as a modification of the standard bootstrap percolation model, described in Chapter 6 and published in [73]. This introduces a stochastic element into an otherwise deterministic process. When modelling social contagion, this might reflect a change of mind scenario. Our results demonstrate that this modification has a delaying effect on the adoption of an activity. This effect is enhanced when activated hub nodes are specifically
targeted for recovery.

- **Introduced Immunity to the Percolation Process**

In our last simulations we chose to focus on inhibiting percolation, by introducing the idea of immunity, described in Chapter 7 and published in [72]. This was modelled by a priori selection of nodes with various properties to be immune to the percolation process. These measures amount to control strategies, which seek to alter the outcomes in the process. In real life, this might relate to identifying influential nodes in a network to stop the spread of a contagious disease, or to prevent social contagion such as rumour propagation, or the spread of fake news.

- **Identifying Influential Nodes**

Using results from the simulations of bootstrap percolation with recovery and from bootstrap percolation with immunity, we have gained insights into the identification of nodes which have significant impact on diffusion processes on these networks, described in Chapter 7 and published in [72]. Our previous analysis of global and local node properties in the set of graphs allows us to correlate simulation outcomes with particular patterns of properties. Node degree is commonly regarded as the standard measure of influence in a network, our results show that this varies according to the structure of the network, and to some extent we can a priori identify node properties which are influential in different graphs, based on the measure of skewed distribution of the property in that specific graph.

### 1.6 Thesis Overview

The structure of this thesis is as follows:
1.6.1 Chapter 2 Background

In this chapter we describe investigations of spreading behaviours on complex networks, this includes descriptions of random graph models, an overview of game theory and bootstrap percolation, and research examining these processes on random graph models.

1.6.2 Chapter 3 Hyperbolic Random Geometric Graphs

In this section we describe hyperbolic random geometric graphs, and our graph creation process. We have created a set of hyperbolic graphs with increasing edge density from 0 to 1. We have also analysed graph properties for a large subset of these graphs and we present this analysis in terms of varying edge densities.

1.6.3 Chapter 4 The Prisoner’s Dilemma

In our first experiments, simulating The Prisoner’s Dilemma on the set of hyperbolic graphs, we were interested in observing any changes in emergent behaviour as we increased the edge density in the graphs. We tested this by taking each node in a graph as the sole initial defector seed in a population of cooperators, simulating the game over many generations and observing the final ratio of cooperators to defectors in the population. This was repeated for all graphs in the set. Our results showed that as the number of edges in the hyperbolic graphs increased, there was a clear threshold for edge density above which the strategy of defection completely spread, and below which cooperation was robust.

1.6.4 Chapter 5 Bootstrap Percolation

In this section we simulated the process of bootstrap percolation on our set of hyperbolic graphs and compared outcomes on graphs of increasing edge density, correlated with various global graph properties. As the number of edges increased in the set of graphs, we observed a distinct percolation threshold above which the activity percolates on all the graphs, and below which, the activity fails to percolate. At this
threshold, we have identified a particular pattern of centrality and clustering measures, which has proved useful in our later simulations where we attempt to alter outcomes by selecting various node properties for special treatment.

1.6.5 Chapter 6 Bootstrap Percolation with Recovery

In this chapter we introduce a modification to the standard bootstrap percolation process, which allows reverse state change. In standard bootstrap percolation, the only possible state change is from inactive to active; in this work we investigate the impact on outcomes when a percentage of active nodes is allowed to recover at each time step. The choice of active nodes selected to recover is performed in two ways, at random and by selecting active nodes of highest degree. We have selected node degree as this is the most common method of defining influence in a network. Nodes of high degree have access, at a short number of steps, to a large section of the graphs due to their high number of contacts, for this reason hub nodes are often described as super-spreaders.

Our initial motivation for the introduction of a recovery option arose when comparing the outcomes from the Prisoner’s Dilemma and bootstrap percolation. It seems reasonable to assume this might model a change of mind scenario in the adoption of a new activity. With regard to social influence, for example, people are more likely to adopt a new gadget if a number of their contacts already use that technology; introducing recovery nodes models a change of mind scenario, where people are persuaded to adopt by peer pressure, but soon realise that they are not that interested in the device. We find that selecting nodes of high degree to recover has a delaying impact on the percolation threshold, when compared to random selection of active nodes to recover.

1.6.6 Chapter 7 Bootstrap Percolation with Inhibitory Nodes

In this section we switch our focus from modifying the rules of bootstrap percolation to attempting to delay percolation by the introduction of inhibitory nodes in each
graph. These nodes are effectively immune to the percolation process and remain so throughout the simulations. The inhibitory set within each graph is selected prior to simulation, and the sets of experiments are repeated using targeted selection, based on various graph properties, and at random for control. Our motivation, given a known network, is to determine if it is possible to select a handful of influential nodes to resist the spread of an unwanted activity in that network, to effectively inhibit percolation of the activity.

On selected percolating graphs at the upper edge of the percolation threshold, we observed that targeted selection of the inhibitory nodes had a greater inhibitory effect on percolation than random selection. We also noted that there appeared to be some correlation between increased impact and the targeting of top ranked nodes for properties with highly skewed distributions on this set of graphs. For example, several of the selected graphs displayed highly skewed betweenness, degree and/or closeness centralisation. In these graphs, targeting the top ranked nodes for high degree or high closeness or high betweenness had a greater inhibitory impact than targeting properties with low skewness. This is interesting as it suggests the potential for intelligent selection of inhibitory nodes, based on prior knowledge of a network’s structure.

1.6.7 Chapter 8 Conclusion

In this chapter we outline our conclusions, summarise the findings in this research, our contribution and suggestions for future work.
Chapter 2

Background and Main Concepts

In this chapter we describe the background to our research, describing the fundamen-
tals of network theory, network models and spatial diffusion on networks.

Section 2.1 outlines some of the related general concepts of network theory that
we use in our research.

Section 2.2 describes the development of random graph models and the intro-
duction of random geometric graphs, in the Euclidean and hyperbolic planes. As a
background to the geometric graphs, we briefly describe non Euclidean geometries, fo-
cusing on hyperbolic geometry, and the development of hyperbolic random geometric
graphs.

Section 2.4 describes the modelling of spreading behaviours on a network. This
includes evolutionary game theory, with a specific focus on the spatial form of the
Prisoner’s Dilemma game. We then describe the Bootstrap Percolation process which
is a simple model for the spread of contagion in a population, with an outline of related
aspects of the compartmental models of epidemic spread. Finally, we introduce the
concept of immunity and influential nodes.

2.1 Network Theoretic Concepts

In this section we define terms from network theory which are related to our research.
2.1.1 Network

A network is a collection of people or objects connected by communication links. The underlying structure of a network is a graph of nodes and edges. Networks may be physically linked, such as a network of interconnected computers and printers. In abstract networks such as social networks, where links between people are defined by their communities of interest such as family, work or leisure activities, these ties may or may not involve physical contact, or may involve communication links such as telephone or internet, e.g. in networks of mutual funds.

2.1.2 Graph

A graph $G = (V, E)$ consists of a set of vertices $V$ (or nodes) and a set of edges $E$, connecting vertices. In an undirected graph, a connection between node $A$ and node $B$ is the same as a connection between node $B$ and node $A$ and is known as an edge. In a directed graph, each connection has a direction and is known as an arc. In mathematics, the labels vertices, edges and arcs are frequently used in graph theory; in computer science, the terms nodes and edges are often encountered. In this research these terms are used interchangeably, as appropriate.

2.1.3 Paths

A path is a sequence of nodes and edges in which no node is repeated. The length of a path is its number of edges.

Shortest Path and Diameter

The shortest path is the path between two vertices that follows the least number of edges. The average shortest path is the average over the entire network. The diameter is the the length of the longest shortest path in the network. The shortest path is a common measure in network analysis, as it is found to be a defining feature of certain graphs, such as small world graphs which have short path lengths, as described in section 2.2.3. In real world networks, path length can be a measure of the efficiency
of communication in a network, with the diameter as a measure of the maximum time of communication to all nodes in a network. The average path length is an indicator of the effective communication rate to the majority of nodes in a network.

2.1.4 Component

A component is a structure wherein each pair of vertices is joined by a path. A giant component consists of one component which covers some pre-specified large proportion of the vertices.

2.1.5 Connected Graph

A connected graph means that every node in the graph can be reached by a path from any other node.

2.1.6 Node Degree

Node degree \((k)\) refers to the number of edges incident to a node. Node degree also indicates the size of the neighbourhood of a node, as each edge leads to a connected neighbour.

2.1.7 Density and Average Degree

The density of a network is defined as the ratio of existing edges to the total number of possible connections in the graph:

\[
Density = \frac{\text{Number of actual edges}}{\text{Number of potential edges}} = \frac{2E}{n(n-1)}
\]

The average degree measures the sum of all the degrees in the graph divided by the number of nodes:

\[
Average \ degree \ < k > = \frac{2 \text{ Number of edges}}{\text{Number of nodes in the graph}} = \frac{2E}{N}
\]
For a fixed number of nodes, average degree and density are proportional. Average degree may be a better measure of network structural cohesion than density as it does not depend on network size. Average degree may therefore be used to compare networks of different sizes.

2.1.8 Skewness

Skewness is a measure of the asymmetry of a distribution. If a distribution is symmetrical, there is zero skew, as in the normal distribution. Positive skewness describes a distribution where the right tail is longer than the left tail. Negative skewness describes distributions where the left tail is longer than the right. Calculations of skewness basically involve measuring the sum of the differences between a variate and the mean value, divided by the standard deviation. There are many alternative definitions of skewness, see for example [80].

![Negative and Positive Skew Diagrams](image)

Figure 2-1: Negative and positive skew diagrams by Rodolfo Hermans is licensed under CC BY-SA 3.0

2.1.9 Degree Distribution

The degree distribution is defined by $P(k) =$ the probability that a randomly selected node has degree $k$. Degree distribution relates to the frequency with which partic-
ular degrees arise in the graph. A skewed distribution has node degrees distributed asymmetrically, typically displaying a mound with a long tail, whereas plots with low skew are symmetrically distributed around the mean.

**Binomial Distribution**

This is seen in graphs where $n$ nodes are connected with a probability $p$. An example is the Erdős Rényi Random graph, $G(n, p)$, where edges are generated with probability $p$. As all probabilities sum to one, the probability that edges between pairs of nodes do not exist is $(1 - p)$; these graphs are described in section 2.2.2. In this graph model, the binomial distribution maps the probability that a node has degree $k$ in $N$ Bernoulli trials. Bernoulli trials are experiments in which events have a probability of success $p$. In a coin tossing experiment, with a fair coin, the probability of getting heads ($p$) is 0.5. The probability of getting tails $(1 - p)$ is 0.5. In the Erdős Rényi model, the maximum degree $k$ of any node is $n - 1$, each node has $n - 1$ attempts to gain an edge. The general binomial formula for the probability that a node has degree $k$:

$$B(n - 1; k; p) = \frac{n!}{k!(n - k)!} p^k (1 - p)^{n-k}$$

In general, the shape of a binomial distribution is affected by the values of $n$ and $p$. For $p < 0.5$ the distribution displays positive skew, for $p = 0.5$, the plot will be symmetric and for $p > 0.5$ the plot will display a negative skew, as illustrated in Figure 2-2 by Johnathon Marchini [69].

**Poisson Distribution**

A Poisson distribution describes the symmetric plot where most nodes have degree very close to the average degree value, as illustrated by Lada Adamic [1] in Figure 2-3. The tails die away rapidly, indicating that values which deviate significantly from the mean are rare. A Poisson distribution has low skewness. When $n$ is large (generally greater than 20) and $p$ is small (generally less than 0.5) Poisson probabilities can be
used to approximate binomial probabilities.

$$p_k = \frac{\lambda^k e^{-\lambda}}{k!}$$

This approximation is especially good for $n \geq 100$ and $\lambda = np \leq 10$.

Normal Distribution

A normal distribution displays a characteristic bell-shaped curved which is symmetrical about the mean. The normal distribution is a family of curves, each with its own
mean $\mu$ and standard deviation $\sigma$. The shape of the curve is such that 68% of the
values lie within one standard deviation of the mean, between $\mu - \sigma$ and $\mu + \sigma$, 95%
lie within 2 standard deviations of the mean and 99% lie within 3 standard deviations
of the mean, so that almost all values are relatively close to the mean. The normal
distribution often results from measurements such as the height of people, the marks
of students in a test or errors in measurements, where most data lies closely ranged
about the mean. The normal distribution is a limiting case of the binomial distribution
as $n$ becomes large, with $\mu = np$ and the variance $\sigma^2 = np(1-p)$. Figure 2-4
displays a normal distribution, as illustrated by Nicos Psomas [105].

![Normal Distribution](image)

**Figure 2-4: Normal Distribution [105]**

**Power Law Distribution**

A power law distribution, as displayed in figure 2-5, is a highly skewed distribution
where most nodes have low degree and some nodes have high degree. This is typi-
cally seen in small world graphs (and scale free graphs) which grow by preferential
attachment, i.e. the “rich get richer”. Small world graphs and scale free graphs are
described in sections 2.2.3 and 2.2.4, respectively. On a log-log plot, where both
axes use logarithmic scales, a power law distribution approximates a straight line.
Figure 2-5 displays a power law distribution, as illustrated by Stowe Boyd [19]. This
distribution has positive skewness, as the right tail is longer than the left. On the
left, most nodes in the graph have low connections, whereas moving towards the right only a few nodes have high degree.

![Power Law Distribution](image)

Figure 2-5: Power Law Distribution [19]

### 2.1.10 Clustering Coefficients

Clustering coefficients are a measure of the extent to which network nodes cluster together. The local clustering coefficient of a node represents the amount of interconnections in the neighbourhood of the node. In a friendship network this evaluates the extent to which friends of a node are friends of each other [7].

The local clustering coefficient of a node is defined by:

$$c_i = \frac{e_i}{\frac{k_i(k_i-1)}{2}}$$

where \((e_i)\) is the number of connections in the neighbourhood of a node and \(\frac{k_i(k_i-1)}{2}\) is the maximum number of connections in the neighbourhood of a node of degree \(k\).

Transitivity describes the links between triples of nodes. A fully transitive relationship is represented by a closed triple, where the “friends of my friends are my friends” [124]. The concept of network transitivity was introduced by Rapoport [107] where “the likely contacts of two individuals who themselves have been in contact are expected to be strongly overlapping.”
Transitivity = \frac{3 \times \text{the number of triangles}}{\text{Number of open or closed triples}}

Global Clustering

The Network Average Clustering Coefficient was introduced by Watts and Strogatz in 1998 in their work on Small World Networks, which they characterised as having a distinctive combination of high clustering with a characteristic short path length [?]. The network average clustering coefficient \( \bar{C} \) (also known as the Watts Strogatz clustering coefficient) is the average of the local clustering coefficients \( c_i \) over all nodes \( n \):

\[ \bar{C} = \frac{1}{n} \sum_{i=1}^{n} c_i \]

2.1.11 Centrality

Centrality measures indicate how central, or important, a node is in a network. If such a measure is applied to the whole graph it is known as centralisation. The centrality measures used in this research are degree, betweenness and closeness centrality, described below, along with other common centrality measures.

Degree Centrality

Degree centrality places value on a node dependent on the number of its connections to other nodes, it is therefore a measure of node degree. The degree centrality \( C_D \) of a vertex \( v \), for a given graph \( G := (V, E) \) with \( |V| \) vertices and \( |E| \) edges, is defined as

\[ C_D(v) = \deg(v) \]

As the number of edges incident to a node increases, this increases the amount of connections that a node has with other vertices. Nodes with high degree centrality may refer to hubs. Highly central nodes may be effectively targeted for network disruption or to facilitate information flow. In a model of disease spread, these nodes
may allow rapid spread of contagion; identifying and isolating central nodes may stem the course of the infection. In a directed graph, degree centrality may be based on in-degree or out-degree. Normalised degree centrality is calculated by dividing the degree of a node by \((n - 1)\), the maximum possible degree. Figure 2-6 shows a small scale graph with the central node having high degree centrality, as it has the maximum number of possible connections to other nodes in the graph. In contrast, the other nodes have low degree centrality, as they have one connection, out of a possible 5 connections.

![Figure 2-6: Graph displaying normalised degree centrality values.](image)

**Betweenness Centrality**

Betweenness centrality is calculated by determining the number of times the node appears on shortest paths between all possible pairs of nodes. A node with a high betweenness centrality score can represent a “brokerage” role in a network, as information between subnetworks has to pass through this node. Such nodes can also represent a good choice to disrupt a network, or as a focus for security measures. Subnetworks can also be effectively quarantined by targeting such nodes. In business, nodes with high betweenness can be targeted to facilitate advertising and brand adoption. Normalised betweenness centrality is calculated by dividing the betweenness of
a node by \((n - 1)(n - 2)/2\), which is the total number of pairs of nodes excluding the node itself. Figure 2-7 shows a small scale graph displaying the non-normalised betweenness centrality values of each node. The central node has the highest betweenness, as the paths between all other nodes in the graph must pass through this node. The outlying nodes have zero betweenness as they are not on a path between any other pairs of nodes in the graph.

![Graph displaying non-normalised betweenness centrality values.](image)

**Closeness Centrality**

Closeness centrality is a measure of the distance between a node and all other nodes in the graph. Nodes with high closeness centrality may be closely linked to important nodes, whilst not necessarily having high betweenness or degree centrality. These nodes are close to the action and may have influence out of proportion to their direct connections, either by receiving information or being able to widely disseminate information.

\[
C(x) = \frac{1}{\sum_y d(y, x)}
\]
Closeness centrality is calculated by finding the average length of the shortest path between a node and all other nodes in the graph. Normalised closeness centrality is calculated by dividing the closeness centrality of a node by \( n - 1 \), the maximum number of other nodes in the graph. Figure 2-8 shows a small scale graph indicating the closeness centrality values for each node. Node B has the highest closeness centrality value, as this node, compared to any other node in the graph, has the least average distance between itself and all other nodes in the graph. Node A has low degree and betweenness centrality, similar to that of node C, yet it is close to the hub node B and therefore has higher closeness centrality than node C.

![Figure 2-8: Graph displaying normalised closeness centrality values.](image)

**Other Common Centrality Measures**

Eigenvector centrality is a recursive measure of influence in a network. Connections to high scoring nodes in a network contribute more to a node’s score, whereas connections to low ranking nodes will contribute less to the score.

For a given graph \( G := (V, E) \) with \( |V| \) vertices, let \( A = (a_{v,t}) \) be the adjacency matrix, i.e. \( a_{v,t} = 1 \) if vertex \( v \) is linked to vertex \( t \), and \( a_{v,t} = 0 \) otherwise. The relative centrality, \( x \), score of vertex \( v \) can be defined as:
\[ x_v = \frac{1}{\lambda} \sum_{t \in M(v)} x_t = \frac{1}{\lambda} \sum_{t \in G} a_{v,t} x_t \]

where \( M(v) \) is a set of the neighbors of \( v \) and \( \lambda \) is a constant. With a small rearrangement this can be rewritten in vector notation as the eigenvector equation \( Ax = \lambda x \).

Page Rank is a variant of eigenvector centrality, used by the Google Search Engine [94]. This assigns values to a web page, based on the number of links to a page, and the number of links to those link pages.

Game centrality uses concepts from game theory to assign levels of importance to a node. For example, high game centrality might be applied to a node which has a major contribution in facilitating the spread of defection in a network [122].

Graph Centralisation Measures

In Section 2.1.11 we defined common centrality measures, where node centrality is a measure of how important a node is in a graph. The definition of centrality can be extended to the whole graph, focussing on the overall organisation of the network. It provides a single value to measure how centrality is distributed in a graph. A high centralisation score represents a skewed centrality distribution, whereas a low centralisation score represents a graph where most nodes are in a similar centrality range. Skewness is a measure of the degree of asymmetry of a distribution, as described in Section 2.1.8. Centralisation coefficients lie in the range \([0,1]\), with values close to 1 representing highly skewed centrality, and values closer to 0 representing low skew. Figures 2-6, 2-7 and 2-8 all display highly skewed degree centralisation, reflecting the high level of disparity in node centrality measures in each graph.

2.2 Theoretical Models of Network Structure

The recent growth of Network Science largely involves studies of networks where the focus is on the relationship between structure and behaviour. These approaches use
theoretical models, based for the most part on random graph models, to study the form of networks and more recently to gain insight into the structure and behaviour of real world networks.

2.2.1 Random Graph Models

Mathematical models of networks are developed to represent real life systems and facilitate the analysis of the structure and processes within these networks. Models have traditionally been developed using random graphs, and particularly the Erdős Rényi model of random graph [34], where each node has a specified probability of being connected to other nodes in the graph. Over the last 20 years, these graph models have been refined to more closely mirror situations seen in real life. Examples include the small world network, where most nodes in the graph are reachable in a small number of steps, which models the “six degrees of separation” scenario [74] and the Scale Free model, which simulates situations where “the rich get richer” [14], in which popular nodes become increasingly more successful at making new connections with new nodes.

A key feature of random graph models is that they describe a collection of all possible graphs created using this model, effectively describing a probability distribution, where all graphs conform to the abstract principles of the model, yet retain a stochastic element.

2.2.2 Erdős Rényi model of Random Graph

In the 1950s, Paul Erdős and Alfréd Rényi developed the concept of random graph models by creating an undirected graph where nodes are connected at random [34]. The Erdős Rényi model of random graph, denoted \( G(n, p) \), describes a graph with a fixed number of nodes \( n \) and edges created between each pair of nodes with a predefined probability \( p \). Erdős and Rényi also developed an alternative model, denoted \( G(n, M) \) which refers to the set of all random graphs of \( n \) nodes which have exactly \( M \) edges.
In this research we are specifically interested in the $G(n, p)$ model. For each pair of nodes, an edge is created with probability $p$, this gives rise to a binomial distribution as described in section 2.1.9. The maximum degree of a node is $(n-1)$. For small $(p)$, the degree distribution of the graphs can be approximated by a Poisson distribution, see section 2.1.9. For large $n$, the degree of Erdős Rényi random graphs approximates a normal distribution.

![Figure 2-9: Erdős Rényi model, $G(n, p)$, with $n = 10, p = 0.6$](image)

As the number of nodes increases, the degree of each node within the graph displays low deviation from the average degree; this indicates an absence of hubs and enables all nodes within the graph to be reached in a short number of steps. The low deviation from average degree for each node, leads to a certain regularity of appearance and allows us to use mathematical calculations to approximate properties such as the average shortest path $[120]$. The length of the average shortest path $(l_{av})$ can be approximated as follows, noting that a node with average degree $z$, is connected to $z$ nodes, all of average degree $z$, so that the number of nodes at distance $l$ is $z^l$:

$$l_{av} = \frac{\log n}{\log z}$$

Erdős Rényi random graphs have been used as the basis of other models such as the
small world model, by systematically rewiring the connections in a predefined manner. They are therefore useful in acting as a benchmark against which to measure other types of models and the resultant differences in graph properties.

### 2.2.3 Small World Graphs

In 1998, Duncan Watts and Steven Strogatz explored models of networks that were not completely regular nor fully random by rewiring edges in a simple ring lattice of \( n \) nodes of degree \( k \). Each edge was rewired with probability \( p \), \((0 < p < 1)\). The resultant graphs were characterised by high clustering and short path length, caused by random short-cuts between pairs of distant nodes. These short-cuts also enabled the clustered neighbours of the node with a rewired edge to be more closely connected to the distant node, yet removing an edge had minimal effect on local clustering. They termed these models “small-world” networks, relating them to the small-world phenomenon observed by Stanley Milgram in experiments which explored empirical chains between randomly selected members of a population, known as the “six degrees of separation” [15].

### 2.2.4 Scale Free Models

In 1999 Barabási and Albert noted that most random graph models and the small world models failed to take into account growth in a network and network dynamics [14]. By exploring large databases of existing networks such as the World Wide Web, they reported that the networks had a distinctive organisation, independent of the particular system, which they classed as scale free; displaying a power law degree distribution as a result of growth and preferential attachment. They noted that older nodes, with higher degree tended to gain more new connections at the expense of newer nodes; they termed this the “rich-get-richer” phenomenon.
2.2.5 Random Geometric Graphs

Random geometric graphs are a type of distance graph model where points are randomly distributed in a space of interest, according to some specified random variable, and connected if they lie within a specified distance parameter of each other. The earliest random geometric graphs were developed in the Euclidean unit disc [27] and unit square $[0, 1]^2$ [100]. They consist of a set of nodes $n$ which are uniformly and independently distributed throughout the geometric space of interest, with edges created wherever the Euclidean distance between two nodes is less than a specified distance parameter $r$. Mathew Penrose set out a detailed study of the graph-theoretic properties of random geometric graphs and noted that for large $n$ the expected degree of each vertex was $n(\pi/2)^2 r$. This contrasts with the Erdős Rényi model which has expected degree of $np$. In general, geometric graphs also display a larger diameter and higher clustering coefficient than the Erdős Rényi model [100].

These graphs are used to model scenarios where the distance between nodes is of interest, such as in the spread of a forest fire, where the fire will spread if trees are within striking distance. Other potential applications include modelling ad hoc wireless networks where devices become connected if they move within range of each other, or sensor networks where monitored devices must lie within transmission distance of a hub node [16]. They can also be used to model the relationship between connectivity, the capacity of a network, potential outages, and other limiting factors in wireless networks [43]. Other applications include determining barrier coverage in wireless sensor networks [56], the use of gossip algorithms to study information exchange [16, 18] and sampling-based path planning algorithms in robotics research, where the ensuing data structures share similar features to geometric graph models [50]. They have been used in modelling neural networks [22] and mapping protein-protein interactions [104]. In percolation theory [24], these graphs are used for modelling processes such as diffusion in a network, fluid percolation in porous materials, fracture patterns in earthquakes, and conductivity [111].

A model for Hyperbolic random geometric graphs was developed in 2010 by Kri-
oukov et al. [55]. This graph model utilises the characteristic features of hyperbolic geometry, described in the next section, 2.2.6, to create graphs with different features.

2.2.6 Hyperbolic Geometry

In geometry, the Euclidean plane is flat, the sphere has positive curvature and the hyperbolic plane is the space of negative curvature. Every point on a hyperbolic surface is a saddle point, resembling a pass between two mountains, but a collection of these local points does not represent the appearance of the larger hyperbolic surface. For this reason, the hyperbolic plane is commonly transformed to a two-dimensional representation using a Möbius transformation. The most frequently used models are the unit disc model, also known as the Poincaré disc model, and the upper half plane model. In the Poincaré disc model, the hyperbolic plane is represented as a disc of radius 1, where the boundary of the circle represents the plane at infinity and is therefore not included in the plane. In the native representation, which we use in our research, $\mathbb{H}^2$ is represented as $\mathbb{R}^2$. In the native model, a point $p$ is represented in polar coordinates, edges maintain their true hyperbolic values and are represented as straight lines.

In hyperbolic geometry, the interior angles of a triangle sum to less than $180^\circ$ and a geodesic or hyperbolic straight line is a circle that intersects the unit circle at right angles. Diameters of the disc are the only Euclidean straight line geodesics. Points on the unit circle are known as ideal points, or points at infinity. Using the Euclidean notion that parallel lines never meet, hyperbolic parallel lines occur if they share an ideal point and do not intersect within the unit disc. Although the lines share an ideal point, these lines do not intersect as the ideal point lies on the boundary, which
Figure 2-11: Types of parallelism in the Poincaré disc. Lines $m$ and $\lambda$ share an ideal point and are parallel. Lines $m$ and $n$ are hyperparallels to line $\lambda$ [44].

is not part of the hyperbolic plane. Hyperparallel lines are hyperbolic lines which do not share an ideal point and do not intersect inside the unit disc; there are infinitely many such lines. Two lines which are hyperparallel to a hyperbolic line, may intersect each other.

The underlying geometry of the Euclidean and Hyperbolic planes create differences in the structure of random geometric graphs created in those spaces. Points that are uniformly distributed within a hyperbolic disc have a markedly different appearance to points with uniform distribution in a Euclidean disc. A hyperbolic disc of radius $r$ effectively has more space within it than a Euclidean disc of radius $r$. The Euclidean disc has circumference $2\pi r$ and area $\pi r^2$. The hyperbolic disc has circumference $2\pi \sinh(r)$, the area is $2\pi (\cosh(r)-1)$. As the radius of a Euclidean disc increases, the circumference increases linearly; in a hyperbolic disc, the circumference increases exponentially. When points are uniformly distributed within the hyperbolic disc, points in the centre appear closer to the viewer, with outer points exponentially distant towards infinity at the boundary.

The hyperbolic tree data structure exploits these features, since child nodes have almost as much space as parent nodes. In consequence, the hyperbolic tree is a way of visualising large amounts of hierarchical data within a finite space and was developed by Lamping et al. in 1995 [58], inspired by the geometric construction of art by M. C. Escher. Escher utilised the concept of the two-dimensional representation of
hyperbolic space bounded by infinity [30] which tends to give a fish eye view to nodes within the disc, with greater emphasis placed on central nodes, while outer nodes are exponentially distant. See for example MC Escher’s woodcut *Circle Limit IV* in the online gallery at https://www.mcescher.com/gallery/recognition-success/circle-limit-iv/. This effect is captured in an image by Claudio Rocchini [109], see Figure 2-12.

![Figure 2-12: Claudio Rocchini Order-3 heptakis heptagonal tiling 2007, distributed under a CC BY 2.5 licence [109]](image)

As a result of the distribution of distances within the disc, the hyperbolic random geometric graphs discussed in Section 2.2.7 are characterised by central hub nodes with leaf nodes towards the boundary.

2.2.7 Hyperbolic Random Geometric Graphs

Hyperbolic random geometric graphs were investigated by Krioukov et al. in 2010 [55]. In this approach, *n* nodes are distributed in a disc of radius *R* in \( \mathbb{H}^2 \), the hyperbolic plane. Nodes are connected if the hyperbolic distance between them is less than or equal to *R*. Using the fact that as the radius *R* of a hyperbolic disc increases, the area expands exponentially, Krioukov et al. described an approximately logarithmic relationship between *R* and the number of nodes *n* in the graph model, such that *R* \( \sim \ln n \). Intuitively, if nodes within the disc are displayed as a hyperbolic tree, starting with a set of nodes in the centre, laying out children at equidistant hyperbolic distance, the child nodes appear at a similar distance to each other and the parent node toward the centre appears closer to the viewer. For a simple binary tree, the
maximum number of nodes at a level \( n \) is \( 2n \). As the radius \( R \) of the hyperbolic disc increases, this creates a hyperbolic network hierarchy of increasing depth. Therefore the relationship of \( R \) scales with \( N \) such that \( R = 2 \log N \).

Krioukov et al demonstrate that for uniform node distribution at curvature \( K = -1 \), the resultant graphs display power law degree distribution, with adjustable exponent. They note that this heterogeneity in degree distribution arose naturally from the hyperbolic geometry intrinsic to the network, and was dependent on node density and average degree. They also introduced more relaxed constraints to give a greater variety in degree distribution, close to uniform distribution.

Gugelmann et al. [42] explored this graph model in more detail and made fine adjustments where the resultant graphs have a power law distribution, whose exponent can be modified. This produces slight variations in the density function for the radial coordinate \( r \), which favours points at different distances from the origin.

They proved mathematically that the degree distribution of the hyperbolic random geometric graph is scale-free up to the maximum degree, and further that the graphs displayed high clustering, by proving a constant low bound on the clustering coefficient. The metrics used for graph creation, and detailed mathematical analyses are described further in [54, 95, 42].

Typically hyperbolic models are highly clustered [23], with power law degree distribution and short path lengths, attributes commonly seen in real world complex networks. Recent work suggests that the Internet graph conforms to an underlying hyperbolic geometry [55] and a tool has been developed to map real world complex networks, such as the Autonomous Systems internet, to a hyperbolic space [96].

Recently researchers have developed faster algorithms for creating large hyperbolic networks. Von Looz et al. have developed a speedy generator to create representative subsets of hyperbolic graphs, allowing for networks with billions of edges, while retaining key features of hyperbolic graphs [130], while Bringmann et al. have developed a generator to create a generalised hyperbolic model in linear time, by avoiding the use of hyperbolic cosines [20].
2.3 Properties of Real World Networks

Real world networks display many properties which set them apart from some of the earliest random graph models, such as the Erdős Rényi model of random graph. For example, Faloutsos et. al. observed that the Internet exhibits a scale-free degree distribution [35]. Mitzenmacher [75] noted that the World Wide Web has power law degree distribution, and this stimulated the design of random graph models that yield web like graphs. Another distinctive feature of large real-world graphs is the high level of clustering, reported in social networks as well as biological and technical networks [82, 117, ?].

2.4 Diffusion Processes on Networks

In this section we explore the background to diffusion processes, or spreading phenomena, on networks. Modelling the relationship between network structure and network behaviours often affords insights into activities on real world networks.

In their work on biological network analysis, Ma et al. [68] noted the relationship between collective behaviours and global network properties, individual node behaviours and properties and neighbourhood, or subnetwork, behaviours [68]. However, networks in different disciplines of science, nature and technology share similar structural features and are amenable to similar analytic techniques [13]. A key feature of the emergent field of network science is that insights gained from the study of one particular type of network may have applications in a totally unrelated field of study. This facilitates cross-pollination of ideas and the rise of interdisciplinary studies. Since the advent of high powered computers, the subsequent development of computational intelligence allows for increasingly complex analysis of networks and empirical studies. Models of networks have been used to model such processes as the spread of diseases in a population, information diffusion in an on-line network, the adoption of brand technology, and changes in voter trends. What all these have in common is that the networks are computationally tractable, and using agent based
simulation allows us to track the spread of activities throughout a network, without altering the actual underlying network. For example, modelling the spread of a disease, or observing the effectiveness of immunisation strategies might share the same fundamental methodologies as research into security threats in an on-line network. This means that the study of networks can be applied to many disciplines, including science and humanities, and the fundamental research in this area may therefore offer insights into many different fields of study, including interdisciplinary studies.

As background to our simulation of spreading behaviours on spatial networks, in Section 2.4.1 we describe game theory, with particular emphasis on the spatial form of the Prisoner’s Dilemma. Bootstrap percolation is described in Section 2.4.2. We describe compartmental models of epidemic spread in Section 2.4.3 and introduce the notion of Immunisation and Influential nodes in Section 2.4.4.

### 2.4.1 Game Theory

Game theory is the study of strategic decision making in interactions between independent intelligent individuals [77]. Classical game theory introduced the concept of the individual rational player, involving careful reasoning about the utility of any potential choice they and their opponents face. In 1944, Von Neumann and Morganstern explored the notion of determining the optimal strategies in competitive situations where individuals, or groups of individuals, can choose a strategy of cooperation or defection [131]. This work provided a fundamental basis for studies of decision-making under uncertainty.

The simple game of Prisoner’s Dilemma, described in Section 2.4.1, is an early example of a Social Dilemma game involving the binary choices of cooperation or defection. In a Social Dilemma game, players can choose to act cooperatively for the benefit of society, or can act selfishly, to their own advantage, despite any detriment to the society at large. In the real world, social dilemmas can occur during times of resource shortages, where selfish individuals may hoard resources, or cooperative individuals take their fair share.
Prisoner’s Dilemma

The simple game of the Prisoner’s Dilemma captures the binary strategy of cooperation and defection, using a pay-off matrix to determine scores. There are four possible outcomes in the simplest form of the game, dependent on the strategies of the players; the generalised payoff matrix is shown in Table 2.1. If the row player co-

<table>
<thead>
<tr>
<th></th>
<th>Cooperate</th>
<th>Defect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cooperate</td>
<td>( r, r )</td>
<td>( s, t )</td>
</tr>
<tr>
<td>Defect</td>
<td>( t, s )</td>
<td>( p, p )</td>
</tr>
</tbody>
</table>

Table 2.1: Generalised Payoff Matrix for Prisoner’s Dilemma Game, where \( s < p < r < t \)

operates and the column player defects, the row player has a sucker payoff \( s \) whereas the column player receives a temptation payoff \( t \), indicated by the top right hand cell. The dilemma arises as both players are better off cooperating and receiving a higher combined reward score \( r \), yet there is a temptation to defect and receive the maximum score \( t \), despite the risk of a lower combined punishment score \( p \) if both players defect.

John F. Nash Jr. observed that there were equilibrium points in \( n \)-player games, reached when the strategy of each player yields the highest possible expectation on countering the strategies of all the other players [79]. In standard games, it is assumed that an individual will make the most rational choice for themselves based on knowing the possible strategies of the other player. This is therefore a deterministic process and in the standard form of the Prisoner’s Dilemma this leads to a Nash equilibrium of defection by both players.

Mixed strategies assign a probability to the different strategy choices, so that for each game a player will choose a certain strategy with a particular probability [131, 78]. The adoption of mixed strategies has the advantage of introducing a stochastic element to the game, which may model more complex situations. However, mixed strategies based on random probabilities may bear little relation to the way that people make choices in real life [6].
Iterated Prisoner’s dilemma

The iterated form of the game [7, 8], allows repeated game play, where players recall the history of their games with other players. This means that an individual player can assess the reputation of another player, and also leaves open the opportunity for revenge for past moves. This leads to different strategies, such as the Tit for Tat strategy. The iterated game allows for exploration of different behaviours, such as cooperation, deception, threatening and exploitative activities [112, 62]. However, this may also encourage players to cooperate over time.

Evolutionary Game Theory

Evolutionary game theory is a development where the strategies of individuals may evolve over time [123]. This approach uses Darwinian concepts for the evolution of species, such as the concept of natural selection where small individual genetic variations allow for the “survival of the fittest” species in competition for resources. Games can be simulated using computational models of evolutionary processes such as survival of the fittest. The mechanism for this learning involves assigning a strategy to each individual; each individual’s score is based on their interactions with other players and on the relevant values in the pay-off matrix. At the end of each round of games, a proportion of the individuals adopts the strategy of higher scoring players and therefore learns to adopt a successful strategy over time.

Reciprocity [84] describes the idea where cooperative interactions are rewarded (positive reciprocity) and negative interactions are punished (negative reciprocity). Evolving strategies may be based on direct and indirect reciprocity. In 2 player games reciprocity, such as punishing defectors, may be used to encourage cooperation. However, in n-player games, this might not be beneficial, as the punishment by a player will be extended to all players with negative interactions, promoting an escalating level of defection [38].

Yao et al. [136] studied n-player variants of the evolutionary form of iterated Prisoner’s Dilemma, as a more generalised model with greater applicability to real
world situations than the conventional 2 player game. They observed the effect of group size on the emergence of cooperation and noted that cooperation becomes less likely with increasing group size.

Evolutionary Games on Graphs

Evolutionary game theory was extended to spatial dimensions in the early 1990s, particularly on 2 dimensional spatial arrays, or lattice grids. Spatial game theory involves interactions between connected neighbours and is ideally suited to modelling as a network [83]. Random graph models have been used extensively in game-theoretic applications, using strategic games such as the Prisoner’s Dilemma to model the spread of cooperation and defection in networks [21, 119, 118]. Key questions involve whether the spatial structure affects the adoption of particular strategies, such as recent interest in the robustness of cooperation [85, 59, 63, 106]. Computational models are created in which the agents are “attached” to nodes in a graph. Each agent is then assigned a strategy and their behaviour is observed, with results showing that spatial effects have a major impact in determining the outcomes of the games.

For example, Nowak et al. [88, 86, 87], researching spatial Prisoner’s Dilemma on 2 or 3 dimensional spatial arrays, noted that the spatial structure allowed for local relative pay-offs which supported the existence of cooperation and defection. This was in contrast to homogeneous populations with random contacts, where one or other strategy might have predominated.

More recent work investigated games on random graph models which share features with real world networks, such as power law degree distribution seen in scale free and small world graphs, described in Sections 2.2. For example in [92] O’Riordan et al. created lattice grids with strong community structure, i.e. independent clusters with loose connections to other clusters. They noted that this structure allowed cooperation to emerge. In an extension to this research, they created small world graphs by rewiring these lattice grids [91]. They reported that if the community structure was preserved, this allowed cooperation to persist. However, defection emerged if the community structure was destroyed.
Initially such analyses were performed mathematically, however the advent of computational intelligence has facilitated the study of more complex simulations over a wide variety of scenarios. In computer science, game theory is used extensively in the study of multi-agent systems, which are computerised systems of autonomous independent agents.

In research on small world and scale free networks, many researchers report that the power law degree distribution and hub nodes help maintain cooperation [113, 132, 102, 103, 110, 135]. For example, Pacheco et al. noted that in scale free and small world graphs, the power law degree distribution allows cooperation to dominate, and demonstrated that their results applied to graphs of different sizes, from 100 nodes up to very large graphs [93]. Li et al. [64] showed that in graphs of low density, high transitivity supported the robustness of cooperation.

More recently, $n$-player games have been explored on dynamic network structures which co-evolve with strategy over time [108, 61, 140]. Szolnoki et al. examined the impact of age distributions on cooperation in a population on lattice grids [126], which introduces a heterogeneous strategy element. Ohtsuki et al. explored the evolution of cooperation and the relationship between “social viscosity”, reputation and strategic complexity [90]. Other research, particularly on standard and toroidal lattice grids, includes variants where agents are allowed to move within the graph, based on neighbourhood stimuli [2, 128, 37, 38].

### 2.4.2 Bootstrap Percolation

Bootstrap percolation describes the process where an activity spreads to individuals in a population when the number of their active contacts is at least a specified activation threshold. Bootstrap percolation was originally developed to model the process of ferro magnetism, in which materials become progressively magnetised [26]. It is a form of threshold model where a population is seeded with a number of active cells and inactive cells become active over time if the sum of their active neighbours exceeds the threshold.

This process can be applied as a model for social reinforcement, under the assump-
tion that individuals are considered more likely to adopt a new activity that is popular with their contacts, and has been used to model a diverse variety of topics, such as the diffusion of information [66, 76, 60], viral marketing [32], spreading behaviours [25], as seen in the forming of opinions, voter trends, the adoption of products, technology and social networking innovation [39, 47, 121], and in cascading power system failures. Interestingly, as the concept was developed in the field of solid state physics, much of the research in bootstrap percolation has adopted a mathematical and probabilistic approach, particularly in the field of statistical mechanics. This applies even for applications of this process to the study of information propagation and other social network applications. This also includes research on information diffusion in online social media which inferred the underlying network by probabilistic methods [41].

The bootstrap percolation model is essentially a two state model, with agents either active or inactive. In some of the literature these states are termed infected or uninfected, but basically means that an activity is present or not present. Standard bootstrap percolation is a discrete-time process, where the activation mechanism occurs synchronously for all nodes in the graph in rounds, at each time step. Starting with an inactive population of agents, a set of nodes is chosen as the active seed set, this choice may be at random or deterministically. An integer value activation threshold is chosen and the activation mechanism occurs at each time step, whereby a node becomes activated if the number of its active neighbours is at least that value. The rounds of activation are then repeated until equilibrium, where no further state change is possible, or at some predetermined parameter, such as the number of rounds. In the spatial form of the process, agents are attached to nodes in a network and activation depends on the number of active nodes directly connected to a node.

Bootstrap percolation has been studied on a variety of random graphs and complex networks [9, 48, 29]. A key consideration in the field is whether or not an activity will spread completely across a network. In 2002, Duncan J Watts [133] explored the potential for cascade failures in a special case of the uniform random graph, where
pairs of vertices are connected with probability

\[ p = \frac{\text{average degree}}{\text{number of nodes}} = \frac{z}{n} \]

with \( z \ll n \), to explain the origin of large cascades triggered by a single node, or a small active seed set, in social and economic networks. They noted that sparse networks were robust against small triggers. In more dense networks, a greater subset of nodes, those with average connectivity, were more likely to trigger cascades. However, as the degree distribution became more heterogeneous this had a protective effect against cascade failure. However, in terms of real world cascades, they were unable to determine which of their theoretical distributions accounted for real world global cascades. They suggested that their framework might stimulate theoretical and empirical studies of more realistic network models. Early work studied the process on 2 and 3 dimensional lattices [115, 46, 10, 45, 11], observing the existence of sharp percolation thresholds on these networks.

It has also been studied on trees [17] and on the regular random graph [12]. In 2010, Baxter et al. studied bootstrap percolation on uncorrelated complex networks and noted that in scale-free networks, percolation may result from a very small trigger [15]. Sellitto et al. studied spin models of glassy dynamics on a Bethe lattice and noted that at low temperatures the phase transition had a similar mechanism to the bootstrap percolation process [116]. In 2014, Amini et al. studied bootstrap percolation in power law random graphs, observing the relationship between in-homogeneity of degree distribution, size of active seed set and complete percolation. observing the critical function where the activity fails to percolate, and the critical point above which the activity spread to a positive portion of the graph. They noted that this was dependent on the presence of hub nodes and contrasted the emergent behaviour with outcomes in the case of the Erdős Rényi random graph, \( G(n, p) \) [5].

Much of the research involved has focused on the relationship between the initial active seed set, and the emergent state of the population. This typically involves looking at the choice of the initial active seed set to optimise the spread of the ac-
tivity [51], or the size of the initial seed set to ensure percolation [40]. Kempe et al. [52] analysed algorithms to optimise the selection of influential nodes on a variety of networks, noting that this was NP-hard. Detering et al. performed a mathematical analysis of bootstrap percolation on directed inhomogeneous graphs with nodes assigned weights based on their sensitivity to the activity in their neighbourhood. They noted a lower bound for the final active set [31].

Candellero et al. [24] researched bootstrap percolation on hyperbolic random geometric graphs by analysing node position within the hyperbolic disc, using inductive reasoning based on node position within the hyperbolic disc to determine the size of the initial seed set for which the activity would percolate completely, fail to percolate, or spread to a positive amount with high probability.

All of these works have involved mathematical analysis based on features such as the probability of an edge existing. In our own work we simulate the spread of an activity on graphs sampled from the random geometric graph model, so that we can observe the actual spread from node to node throughout each individual graph, rather than using probabilistic analysis.

The bootstrap percolation process has been used to model information diffusion [66, 60, 76] and viral marketing [32], behavioural diffusion [25], such as opinion formation and voter trends, the adoption of brands, technology and innovation in social networks [39, 47, 121], and also cascading failures in power systems.

In the standard form of bootstrap percolation activated nodes must remain active, no reverse state change is allowed. In 2014, Coker and Gunderson investigated the idea of bootstrap percolation with recovery on lattice grids based on an update rule that infected nodes with few infected neighbours will become uninfected [28]. They used a probabilistic approach to determine thresholds for the probability of percolation based on the size of the initial active seed set. By examining various configurations of 2-tiles (pairs of sites that share an edge or a corner in the lattice grid) they determine the critical probabilities for percolation.

In 2014, Einarsson et al. [33] analysed an extension of the bootstrap percolation process, in which inhibitory and excitatory nodes were introduced to Erdős Rényi
random graphs. They assume each vertex is inhibitory with probability $\gamma$, and excitatory with probability $1 - \gamma$, independently. Activation also differed from the standard bootstrap percolation process, with an inactive node turning active in some round $i$ if after round $i - 1$ the number of active excitatory neighbours exceeds the number of its active inhibitory neighbours by at least $k$, the activation threshold. They noted that in a round based approach, where activation occurred in rounds, the introduction of inhibitory nodes did not result in stable behaviours. Either inhibition had no impact on the percolation process or tiny changes in the size of the initial seed set could lead to dramatic changes in the size of the final active set. Moving from the round based approach to a continuous time model, where the transmission time for each node was applied at random, the process reached equilibrium and percolation occurred faster than in the traditional round based approach.

### 2.4.3 Epidemic Modelling

In epidemiology, related models are the *SI* and *SIS* models of disease propagation [127] which have similar state changes to bootstrap percolation and to bootstrap percolation with recovery, respectively. These models are known as compartmental models, where individuals are classified into different population groups or compartments. Compartmental models are governed by a system of differential equations that track the population as a function of time, stratifying it into different groups based on risk or infection status. In the SI model, the population is compartmentalised into either *Susceptible* (i.e. uninfected) or *Infected* sinks. Each susceptible individual has a probability of becoming infected, based on the transmission rate, which is a function of the typical number of contacts per individual, and of the infectivity of the disease. Once infected, individuals move to the infected compartment and remain infected. The SIS model is an extension of this model where infected individuals will recover (and become susceptible again) based on the rate of recovery..

The key point of interest is the epidemic threshold, and the effects of varying rates of infection and recovery on the spread of the disease. In the SI model, the infection will ultimately spread to the entire population. In the SIS model, for high recovery
rates the illness will die out in the population and for low recovery rates the illness will become endemic [13].

In compartmental models, the population in each sink is homogeneous, therefore each individual within a compartment has the same likelihood of changing state. One of the advantages of this feature is that it can model the behaviour in a large population relatively simply over time, the same rule will apply to each agent in a compartment, the only things changing are the flow rates between compartments. In advanced models, new compartments can be added to create greater layers of complexity, such as the $SEIR$ model. The model tracks the number of individuals in each of the following categories:

- **Susceptible**
  - Individual is able to become infected

- **Exposed**
  - Individual has been infected with a pathogen, but due to the pathogens incubation period, is not yet infectious

- **Infectious**
  - Individual is infected with a pathogen and is capable of transmitting the pathogen to others

- **Recovered**
  - Individual is either no longer infectious or has been otherwise removed from the population

where $N = S + E + I + R$ is the total population.

This model has a non-linear dimension based on feedback loops. For example, as the size of the infectious compartment increases, this means that susceptible people are more likely to be in contact with an infectious person therefore this has a reinforcing effect on beta, the rate of moving from susceptible to exposed. If vaccination was introduced, then the idea is that this would be a balancing feedback structure.
These models have been used effectively in observing the course of pandemics, such as the observing the mean speed of an Influenza pandemic in a geographic location, or the median delay between the notification of the first case and the mortality peak [129]. They have also been used to gauge the impact of intervention measures such as prison infectious control measures [81], or when comparing the effectiveness of the social isolation measures and vaccination against the Ebola virus in three different states in West Africa [4]. These studies demonstrate the effectiveness of observing an overview of spreading behaviours in a population.

Bootstrap percolation is distinctly different from the epidemic models, as the population of agents is heterogeneous, each with local knowledge of node properties in their neighbourhood. Once the initial active set has been chosen, the spread of the activity is determined by the number of neighbourhood links to active nodes, which is a function of the underlying network topology. This allows the observation of specific nodes in a network, within their own local neighbourhood, and the effect of local structure on the emergent global behaviour.

To address issues of network structure in the epidemic models, Pastor-Satorras and Vespignani in 2001 [97] introduced new compartments based on node degree, with nodes of same degree having the same likelihood of changing state. In large scale free models, the hub nodes ensure that the infection is likely to spread [13]. However, this approach does not take into account community structure or local clustering. Agent based modelling of bootstrap percolation on a network, is a simple dynamic process ideally suited to accounting for individual node features and their subsequent impact on global outcomes. This also facilitates the identification of individual nodes which exert particular influence on the percolation process.

2.4.4 Influential Nodes in Networks

Particular nodes in networks may be described as influential, for example when identifying nodes which might facilitate the flow of a process, or which might inhibit the same process. Examples in real world social networks include social media influencers who may be targeted by marketing companies to promote brand awareness [49], or
to provide public information and warnings in disaster management scenarios [138]. Particular nodes may be influential in spreading unwanted activities and efforts may be made to identify and obstruct such threats [53, 125, 137].

The topological differences in complex networks have meant that over the years a variety of measures have been used to identify influential nodes in networks. In a recent comparative review of progress in this area, Lu et al. clarified concepts and commonly used measures of influence, noting that several studies have found different measures to be influential based on whether the goal was to facilitate or obstruct the progress of the activity [57].

Common measures of influence are centrality measures, such as degree centrality, betweenness centrality and closeness centrality, described in section 2.1.11. The identification of such nodes means that these nodes might be exploited when trying to alter the outcomes of a process on a network. For example, nodes which obstruct a desirable activity, or nodes which facilitate the spread of an unwanted process, might be targeted to mitigate their effect on the process. Similarly, nodes which facilitate the spread of a desired activity might be targeted to disrupt the activity and therefore might warrant protection, for example from a virus in a computer network. Equally, hub nodes in a social network, commonly classed as super-spreaders, might be disconnected from the network, or otherwise prevented from spreading an activity such as fake news in an on-line network.

Recent work has looked at hybrid strategies, taking into account network structure and node activity [67, 139, 65]. Ghanbari et al. have looked at correlation of centrality measures in cascade failures, noting that in their networks, removing nodes of high degree had less severe impact than removing nodes of lower degree [36].

**Immunisation and Influential Nodes**

In this context, the notion of immunity comes from epidemiology and the spread of diseases. In 2002 Pastor-Satorras and Vespignani introduced immunised nodes into a simple SIS (Susceptible - Infected - Susceptible) model of scale free networks such as the sexual partnership web and the Internet [98]. They noted that uniform random
immunity had little effect, due to the inhomogeneity of node degree, and suggested that successful immunisation strategies should target immunity based on node degree ranking. Recent work in the spread of animal diseases has focussed on the role of resilient nodes [53] in controlling disease transmission in farm networks.
Chapter 3

Hyperbolic Random Geometric Graphs: Creation and Analysis

3.1 Introduction

In this chapter we describe hyperbolic random geometric graphs and our method for creating the set of hyperbolic graphs which we use in all of our simulations of dynamic processes.

We choose hyperbolic random geometric graphs for our simulations as they are a recently developed random graph model with interesting graph properties that make them suitable for modelling complex networks. The main areas in modelling involve mathematical analysis, statistical descriptions, empirical analysis of data and simulations. Yet most of the research into diffusion processes on these graphs has been performed as mathematical and probabilistic analysis, see for example [28, 24]. To the best of our knowledge there has been no research using agent based simulation on these graphs. This seems to be a missed opportunity, since these graphs share many features of real world networks, such as high clustering, short path lengths and power law degree distribution [42, 36], as described in Section 2.2.7.

For the purposes of our simulations and for ease of comparison, we have created a synthetic data set of hyperbolic random geometric graphs, all having 1000 nodes, and with increasing edge density from 0 to 1, that is from disconnected up to fully
connected (complete) graphs. We create 20 graphs at each graph creation parameter, map edge density to this distance parameter and measure various graph properties at each parameter. Overall we have a set of 2,400 graphs, with 20 created at each of 120 distance parameters.

Results from this chapter have been used to correlate graph properties with emergent behaviour simulated on the set of graphs, and have been previously published [71, 73, 72].

3.2 Graph Creation

The process we use to create our hyperbolic random graphs is a geometric process based on points in the hyperbolic plane. This approach, outlined in Section 2.2.7, was developed in 2010 by Krioukov et al. [55]. Points are chosen in a disc of radius $R$ in the hyperbolic plane. Edges are created where points lie within $R$ of each other, with a range of values for $R$.

3.2.1 Point distribution

Nodes in the graph are represented as points with polar coordinates $(r, \theta)$. For each value of $R$, we choose a polar coordinate $(r_i, \theta_i)$ with $i = 1$ to $n$. These coordinates are generated as random variates, such that $r_i = \cosh^{-1}(y(\cosh(R) - 1) + 1)$ ($0 \leq r \leq R$), $y$ is randomly generated from $[0, 1]$ and $\theta$ is randomly generated from $[0, 2\pi]$. $y$ and $\theta$ are independently and uniformly distributed random variates. $r_i$ is a random variate between $[0, R]$ with non-uniform distribution.

3.2.2 Edge creation

Edges are created between pairs of nodes where the hyperbolic distance between them is less than the radius $R$ of the disc of interest. Mathematically, this is calculated as follows: let a pair of nodes have polar coordinates $(r, \theta)$ and $(r', \theta')$ respectively. The
hyperbolic distance \((d)\) between these nodes is:

\[
d = \cosh^{-1}\left[\cosh(r)\cosh(r') - \sinh(r)\sinh(r')\cos(\theta - \theta')\right]
\]

Pairs of nodes are connected if the hyperbolic distance \(d \leq R\).

### 3.2.3 Implementation

In this research we use graphs of 1000 nodes, i.e. \(n = 1000\). All of our graphs have 1000 nodes, as our previous research demonstrated that this is sufficiently large to analyse complex contagion processes, yet small enough to be computationally tractable [70].

To create a useful range of graphs we have chosen values for the distance parameter \(R\) in increments of 0.1, from 0 to 12, and create 20 graphs at each value for \(R\). This results in the creation of a set of 2400 hyperbolic graphs of varying edge density from 0 to 1.

The graphs were created using Java. The data structure chosen was the adjacency matrix. An adjacency matrix consists of a matrix of rows and columns where each column and row label represents a node. Each element in the matrix is either a 1 or 0, representing the presence or absence, respectively, of a connection between the respective nodes. As our graphs are undirected and do not contain self-loops, the elements on the diagonal are all zero and the matrix is symmetrical. Whilst this structure is computationally expensive for data storage, it is computationally efficient for performing the row operations which are required in the analysis of node interactions. These operations involve sampling multiple changes of state in each generation. Our graph creation algorithm is described in Algorithm 1.

Our density function for the radial coordinate \(r\) favours points towards the origin; this tends to mitigate boundary effects and give more edge distribution towards the centre. Taking values of \(R\) from 0 to 12, this generates a useful range of graphs with nice properties to form part of our data set of hyperbolic graphs. Analysis of these

\(^1\)Since \(d[v_i - v_j] = d[v_j - v_i]\), then \(A[i, j] = A[j, i]\)
Algorithm 1 Create Hyperbolic Geometric Graph $G(n, R)$ with $n$ nodes in a disc of hyperbolic radius $R$ in $\mathbb{H}^2$

\begin{algorithm}
\textbf{Input:} Number of vertices $n$, distance parameter $R$
\textbf{Output:} Adjacency matrix: $A[i, j]$
\begin{algorithmic}
\State For each vertex $v_i \in V[G]$ do
\State \hspace{1em} Generate polar coordinates:
\State \hspace{2em} $r \in [0, R]$, $\theta \in [0, 2\pi]$
\State \hspace{1em} For each pair of vertices $v_i, v_j \in V[G]$ do
\State \hspace{2em} for $i \leftarrow 1$ to $n$ do
\State \hspace{3em} for $j \leftarrow 1$ to $n - 1$ do
\State \hspace{4em} Calculate hyperbolic distance $d[v_i - v_j]$ \(^1\)
\State \hspace{4em} if $d < R$ then
\State \hspace{5em} $A[i, j] \leftarrow 1$
\State \hspace{4em} else
\State \hspace{5em} $A[i, j] \leftarrow 0$
\end{algorithmic}
\end{algorithm}

properties is discussed in Section 3.3. Since each point $(r, \theta)$ is a point in the native representation of the hyperbolic plane [55], we can think of the resultant graphs as hyperbolic random geometric graphs.

### 3.3 Graph Analysis

In this section we describe the graph properties of the set of hyperbolic graphs as we increase edge density from 0 to 1. We have created 20 graphs at each distance parameter $R$, and we calculate the mean value for various graph properties at each parameter. For each graph property, there is little variance displayed among the 20 graphs at any specified distance parameter.

#### 3.3.1 Mapping $R$ to edge density

The relationship between our graph distance parameter $R$ and edge density is not linear, but is described by the curve in Figure 3-1. This is basically an ‘s’ shaped logistic style curve. In this graph model, edges are created where the hyperbolic distance between nodes is less than $R$. Initially the distance parameter is extremely small and the initial number of connections grows very slowly as we increment values.
for $R$. The number of connections then increases with exponential growth in the mid range of values from 4 to 10 and levels off as we approach 12, our maximum possible value for $R$. In terms of edge density the exponential growth occurs from 0.036 to 0.67, which equates to average degree between 36 and 670 edges per node in our graphs of 1000 nodes.

Figure 3-1: Distance parameter $R$ mapped to density

3.3.2 Global graph properties

In this section we describe various graph properties in the set of hyperbolic graphs. The properties of interest are listed below, and described in section 2.1. The charts are marked with a line at $R = 2.5$, at which point all the nodes are part of the giant component i.e. where all nodes in the graph are reachable. We have also truncated our charts to below distance parameter 6, as all of our later work on simulating network processes shows that key areas of interest, such as transition thresholds, are contained within $0 \leq R \leq 6$. The properties of interest are as follows:

- Clustering
- Watts Strogatz Clustering Coefficient
- Transitivity

- Graph centralisation measures
  - Degree centralisation
  - Betweenness centralisation
  - Closeness centralisation

- Average Degree

- Path Lengths
  - Diameter
  - Average shortest path

Clustering

Figure 3-2 shows that to the right of the marked line, where all nodes are part of the giant component, the graphs display high clustering values. The Watts Strogatz network average clustering coefficient remains high and increases, whereas the transitivity decreases and levels off. This disparity between the clustering measures is a particular feature of hyperbolic random geometric graphs. As the number of edges increases, the graphs display power law degree distribution, this means that there are a few nodes with many connections, whereas most nodes have low connections. Since transitivity measures the number of triples, this actually decreases as we get more hub nodes.

Centralisation Measures

Figure 3-3 shows that degree centralisation is increasing exponentially, betweenness centralisation is falling and then begins to rise again, closeness centralisation is high and increasing. This latter finding is explained by the fact that as the number of
Figure 3-2: Clustering coefficients mapped to distance parameter $R$

As the number of edges increases, the number of hub nodes also increases which means that any node is likely to be close to an influential hub node.

Figure 3-3: Graph centralisation measures mapped to distance parameter $R$
Average Degree

Figure 3-4 shows the average degree. The average degree is low as the giant component emerges and increases slowly within the range 15 to 100 edges per node. Although this is low in terms of the maximum possible 999 edges per node, real world complex networks are similarly sparse.

![Average Degree of Hyperbolic Graphs](image)

Figure 3-4: Average degree mapped to distance parameter $R$

Path Lengths

Figure 3-5 shows path lengths in the set of graphs. The diameter decreases from 10 to 3 and the average shortest path decreases from 5 to 2.5, showing that path lengths in the graph are short and decreasing; as the number of hub nodes increases, it is only a few short steps between any pair of nodes in the graph.

3.4 Discussion

Our main contribution in this section is the creation of a synthetic data set of hyperbolic random geometric graphs. The choice of a random graph model arose as it
is easier to create and control the parameters in synthetic models than in real world networks. Using random graph models also ensures a level of stochasticity in the resultant graphs, since any graph created is part of an ensemble of graphs within this distribution. This particular hyperbolic geometric model has been chosen as it shares many features with real world complex networks [36], such as high clustering, short path lengths and power law degree distribution, as described in Section 3.3.2. The analysis of our set of graphs supports this. Gugelmann et al. found that, with high probability, sampling from this distribution returned graphs with the desired properties [42] which is a key feature in validating experimental results. We also confirm that sampling from the ensemble of graphs in this distribution, produces graphs at each distance parameter with similar graph properties.

In the data set of hyperbolic graphs, edge density has been mapped to various graph properties of interest, as described in section 3.3. In the following chapters, simulating the Prisoner’s Dilemma and bootstrap percolation on the set of hyperbolic graphs, these details will help support the analysis of the impact of topology on emergent behaviours.
Chapter 4

Agent based modelling of The Prisoner’s Dilemma

4.1 Introduction

In the next 5 chapters we explore spreading behaviours on the set of hyperbolic graphs of increasing edge density from 0 to 1. The overall aim of our research project is to investigate the effects of local and global topology on spreading behaviours simulated on the networks. Our first set of experiments apply agent-based simulations rooted in evolutionary game theory to the set of graphs, observing the evolution of strategies over time in a network. In this section, we investigate the simple game of the Prisoner’s Dilemma, embedded in the hyperbolic networks.

In simulations of the spatial and evolutionary form of the Prisoner’s Dilemma, agents with an assigned strategy of cooperation or defection are attached to each node and play the game with their connected neighbours over several generations, as described in Section 2.4.1. At the end of each round of games, all nodes adopt the strategy of their best scoring neighbour. In this way, the long term evolution of strategies can be determined. In the standard form of the Prisoner’s Dilemma game, the dominant strategy is defection. Game theorists are interested in circumstances under which cooperation persists. In our previous research, comparing game-theoretic outcomes on the Erdős Rényi model of random graph and random geometric graphs
created in the Euclidean and hyperbolic plane, we observed that defection spread rapidly on the Erdős Rényi model and also in the hyperbolic model [70]. This latter result was unexpected, as it had been anticipated that the tree like structure of the hyperbolic graph might allow for resistant clusters. However that work was restricted to a small sample of graphs, all with the same selected edge density, to allow direct comparison between the three graph models.

In this work, we extend our research by simulating the Prisoner’s Dilemma on a set of hyperbolic graphs of increasing edge density from 0 to 1. Our motivation, as we increase the number of edges in the graphs, is to determine if there is a transition threshold above which defection completely spreads and below which cooperation is robust. We then analyse the changing pattern of global graph properties as edge density increases and compare this with our threshold to see if we can identify any particular structural features which might inhibit or facilitate the spread of defection.

This chapter includes work which has been previously published [71].

Section 4.2 describes our research questions and hypotheses. Our experimental set up is outlined in Section 4.3, with results presented in Section 4.4 and discussed in Section 4.5.

4.2 Research Questions and Hypotheses

4.2.1 Research Questions

1. In the hyperbolic random geometric graph, as the average degree decreases, can we identify a game-theoretic transition threshold from the adoption of complete defection to the start of resistance to defection?

2. Given that the hyperbolic graphs display different patterns of global graph properties as edge density increases, is it possible to observe any distinct patterns of properties at the game theoretic transition threshold?
4.2.2 Hypotheses

1. It is possible to identify a game-theoretic transition threshold in the hyperbolic model, between complete adoption of defection and resistance to defection.

2. At the game-theoretic transition threshold in the hyperbolic model, it is possible to identify distinct patterns of global graph properties.

4.3 Experimental Design

All of our simulations are performed using the previously created data set of hyperbolic graphs, investigating the impact of graph topology on game-theoretic processes by embedding the game of Prisoner’s Dilemma on the graphs. Each graph has 1000 nodes. The set has 20 graphs created at each distance parameter $R$, from 0.1 to 12 in increments of 0.1. The global graph properties for each graph were analysed, and the mean values for these 20 graphs were presented in Chapter 3.3.2.

4.3.1 Simulation of the Prisoner’s Dilemma

In these simulations, a population of agents playing the game of the Prisoner’s Dilemma is embedded on each graph to observe any changes in the ratio of cooperation to defection that may arise. An agent is assigned to each node, with a predefined starting strategy. At each time step, each node plays against all of their connected neighbours. At the end of each round of games, scores are normalised and all nodes adopt the strategy of their highest scoring connected neighbour.

In the Prisoner’s Dilemma, players with an assigned strategy of either cooperation or defection, play a game and receive a score based on a simple payoff matrix. The payoff matrix for our experiment is shown in Table 4.1.

In this research, we are particularly interested in observing the effects that any structural differences might have on this process. For each graph model, we assign one agent as the sole defector amongst 999 cooperators and observe the impact that the initial defector node has on the spread of defection or robustness of cooperation;
simulated over 100 generations. This is repeated in turn for all nodes in the graph. The results are then compared and analysed.

### 4.4 Results

In this section, we describe the results obtained from simulating the Prisoner’s Dilemma game on hyperbolic random geometric graphs of varying edge density. Our results show an identifiable game-theoretic transition threshold; graph properties at this threshold are presented in Section 4.4.2.

#### 4.4.1 Game-Theoretic Transition Threshold

Figure 4-1 shows the game-theoretic outcomes at equilibrium on hyperbolic random geometric graphs created with varying average degree. Complete defection is shown in red and persistent cooperation in purple. The horizontal axis represents outcomes from all 1000 simulations on the same graph, each with a different defector seed. Each row contains average data at equilibrium obtained from simulations on the 20 graphs at each distance parameter, sorted from least to most defection at equilibrium. For example, at an average degree of 27 edges per node, the mean values for the 20 graphs at this parameter show cooperation was completely robust for only 2 seed nodes, 8 nodes had outcomes of less than 50% defection, whereas most seed nodes resulted in complete defection. Since the threshold occurs at a relatively sparse edge density, for clarity, the heat map has been truncated at an average degree of 40 edges per node, out of a possible 999 edges per node.

The heat map shows a clear transition from the robustness of cooperation below an average degree of 15 edges per node to the complete adoption of defection above
Figure 4-1: Heatmap showing Prisoner's Dilemma outcomes on hyperbolic graphs created with varying average degree. Nodes sorted by outcome in descending order of cooperators an average degree of 38 edges per node. This threshold lies between the distance parameters of 1.5 and 4, within the blue lines in Figures 4-2 to 4-6.

4.4.2 Graph Properties at Threshold

The properties of interest are component size, density, path length, centralisation measures and clustering, as described in section 2.1. The graphs become minimally connected at distance 2.5. As we are only interested in the spread of strategy in a connected graph, this critical point has been marked on the charts with a red line, and we are particularly interested in the zone between the red line and the rightmost blue line, which is the upper boundary of the transition threshold.

Component Size

Figure 4-2 shows the size of the largest component mapped to distance parameter $R$. The graphs become minimally connected at distance parameter 2.5, marked on
the chart by a red line, this equates to an average degree of 15 edges per node. At this point, all nodes are reachable by a path from any node. The blue lines mark the boundaries of the transition threshold from the persistence of cooperation, below a distance parameter $R = 1.5$, and the complete adoption of defection above 4. This equates to an average degree of between 15 and 35 edges per node.

![Figure 4-2: Size of giant component at varying distance parameter $R$](image)

**Density**

Figure 4-3 shows density mapped to distance parameter $R$. The graphs become minimally connected at a density of 0.013, which in our graphs of 1000 nodes equates to an average degree of 13 edges per node. The threshold for complete defection occurs at density 0.037, at an average degree of 37 edges per node.

**Network average clustering coefficient and transitivity**

Figure 4-4 shows the network average clustering coefficients and transitivity measures at varying distance parameter $R$. For all significant distance parameters, the graphs display high clustering. At the transition threshold there is a marked and increasing
Figure 4-3: Edge density at varying distance parameter $R$.

Figure 4-4: Clustering coefficients, values in the range 0 - 1.

Variation between the two measures, with transitivity decreasing significantly. Interestingly, this is in marked contrast to typical values in the Erdős Rényi model of
random graph, mentioned in section 4.1. Those graphs display low clustering and both clustering measures are practically indistinguishable.

**Diameter and average shortest path**

The shortest path between any pair of nodes in the graph is the least number of edges between those nodes. The average shortest path length is the average of all the shortest paths in the graph. The diameter is the longest of these shortest paths and represents the “width” of the graph. Figure 4-5 shows greater adoption of defection as diameter and average shortest path length decrease.

![Diameter and Average Shortest Path](image)

**Figure 4-5: Diameter and average shortest path at varying average degree**

**Degree, betweenness and closeness centralisation**

Figure 4-6 displays comparisons of the three centralisation measures, mapped to increasing distance parameter $R$. All the centralisation values are in the range [0,1].

In Figure 4-6, the hyperbolic graphs display low skew for degree centralisation in the transition zone, reflecting the low level of variance in node degree throughout
each graph. However, the variance in degree centrality is beginning to increase exponentially at the threshold. The chart shows decreasing betweenness centralisation as the tendency to defection increases. Closeness centralisation is increasingly skewed within the transition zone, indicating an increasing degree of disparity in individual node centrality scores. Note that closeness centralisation could not be calculated for values less than 2.3, as the graph was in several components.

4.5 Discussion

Our main contribution in this section is an observation of emergent strategies in the game of Prisoner’s Dilemma played on a set of hyperbolic graphs of increasing edge density from 0 to 1.

Our experiments show that as we increase the number of edges in the set of graphs from 0 to a fully connected graph, there is a distinct game-theoretic threshold in the hyperbolic random geometric graphs. Below this threshold, cooperation is robust and after it there is no resistance to the spread of defection. This threshold occurs at an
edge density between 0.015 and 0.035, which in our graphs of 1000 nodes equates to 
an average degree of between 15 and 35 edges per node, out of a possible 999 edges 
per node in a fully connected graph. Whilst the hyperbolic graphs at the threshold 
are relatively sparse, sparsity is also a feature of many real world complex networks.

Overall we find a threshold zone between 15 and 35 edges per node where there 
are less clear results. Although we find either cooperation or complete defection on 
the graphs in each simulation, the outcome varies according to which defector seed is 
chosen. For example at average degree of 18 edges per nodes, the ratio of simulations 
resulting in cooperation or defection is 60:40. At about 23 edges per node, this ratio 
is reversed.

At the threshold, closeness centralisation increases, meaning that there is increas-
ing disparity in closeness centrality values for individual nodes in the graphs. All of 
the minimally connected graphs have a high level of clustering but the variance be-
tween transitivity and network average clustering coefficient, our selected clustering 
measures, increases as defection becomes more likely, with transitivity decreasing as 
the number of inter-connections in the neighbourhood of a node decrease.

To some extent our results confirm results obtained on other complex networks. 
Previous research showed that heterogeneity promoted the robustness of cooperation 
on small world and scale free graphs [113, 93, 102, 103, 135], noting that the power law 
degree distribution and the presence of hubs supports the persistence of cooperation. 
In our graphs below edge density 13 edges per node, cooperation persists as the 
dominant strategy. However, a small increase in edge density now allows defection to 
predominate. These graphs are still characteristically hyperbolic, with heterogeneous 
degree distribution. Yet, as edge density increases, the increasing hub size and the 
increasingly skewed degree distribution allows for the emergence of defection as the 
dominant strategy.

Previous research had shown a relationship between average degree, transitivity 
and the maintenance of cooperation. Li et al. [64] showed that in graphs of low den-
sity, high transitivity supported the robustness of cooperation. Our results support 
this analysis, Figure 4-4 shows decreasing transitivity at the threshold for defection.
The results from our simulations might offer insights into the vulnerability to attack of real world complex networks, for example identifying potential targeting of nodes with high closeness centrality, which might not have as much security as nodes with high degree. This finding underpins some of our later work simulating bootstrap percolation on the same set of graphs, and in Chapter 7 we attempt a form of intelligent selection of nodes to inhibit the spread of an activity on a network, based on an analysis of global graph properties.
Chapter 5

Bootstrap Percolation on Hyperbolic Random Geometric Graphs

5.1 Introduction

In Section 2.4 we noted that research in network science might have applications in many disciplines, and might offer new perspectives on different network processes. This flexibility allows us to speculate about the behaviour of different processes on the same networks, and whether they might share similar patterns of outcomes, for example.

In the last chapter we observed the behaviour of the evolutionary form of the Prisoner’s Dilemma game on the set of hyperbolic graphs of increasing edge density from 0 to 1. As edge density increased, we noted a clear transition threshold between the robustness of cooperation and the spread of defection. A key feature was the well defined nature of the transition threshold; apart from the graphs at the threshold, the strategy either evolved or failed to evolve, all within a relatively short range of edge densities. We also observed a particular pattern of graph properties at this threshold, as described in Section 4.4.2. This led us to consider if the threshold resulted from structural features of the graphs, or if it was a distinctive outcome of modelling evolutionary game theory on the graphs. For that reason, we chose to model a process that was essentially different to the Prisoner’s Dilemma, but had important points of
similarity. The differences might therefore offer insights into any variance in outcomes on the set of graphs. To test this idea, we selected bootstrap percolation, which is a simple model for the spread of an activity in a population and is also a binary state process. In the Prisoner’s Dilemma the evolution of a strategy in a population is based on nodes adopting the most successful strategy in a neighbourhood, nodes may change state in either direction and on multiple occasions. In bootstrap percolation, described in Section 2.4.2, the spread of an activity is based on a tipping point in the neighbourhood of a node, and state transition is unidirectional, from inactive to active state.

In this chapter we simulate bootstrap percolation on the set of hyperbolic random geometric graphs. Our particular focus is in observing the effect of a small scale random seeding of an activity in a network. This could mimic a network where a virus is introduced to a small number of nodes in a population, or perhaps active nodes contain a brand message, to model brand adoption in a social network.

Section 5.2 describes our research questions and hypotheses, and our experimental set up is outlined in Section 5.3. Our results are presented in Section 5.4 and discussed in Section 5.5.

Work from this chapter has been previously published [73, 72].

5.2 Research Questions and Hypotheses

In this work we embed our activities on the set of previously created hyperbolic random geometric graphs. We are interested in observing the course of the activity as we increase the number of edges in the graphs and whether there is any kind of visible transition threshold where the emergent behaviour in the network changes from inactive to active.

Recently, Candellero et al. [24] performed a probabilistic analysis of bootstrap percolation on hyperbolic networks and found that the activity either failed to percolate or asymptotically almost surely (a.a.s.) percolated, dependent on the physical location of nodes in the active seed set. Their analysis used the general structural
properties of hyperbolic networks which means there tends to be a core of interconnected hubs with leaf nodes towards the boundaries. It is interesting to see if we can match their theoretical analysis with actual simulated results.

Our own focus is in investigating complex networks as a model for heterogeneous populations, to observe how the changing structure of the network has an impact on the spread of the activity, and to ascertain which node properties are significant in enhancing the spread.

The notion that having more contacts will increase a node’s probability of infection, is a well developed idea, with node degree often regarded as being highly influential in determining the spread of an activity. Given that our set of hyperbolic graphs have a distinct and varying pattern of global graph properties as edge density increases, we wish to observe if any characteristic patterns can be observed at any potential transition threshold. This might allow us to identify other centrality or clustering measures which may have greater influence in these networks in facilitating or obstructing the spread of the activity.

5.2.1 Research Questions

1. As we increase edge density from 0 to 1 in the set of hyperbolic random geometric graphs, is there a clearly defined percolation threshold, above which the activity percolates and below which there is failure to percolate?

2. Candellero et al.[24] theorise that for hyperbolic graphs, the activity either percolates completely or fails to percolate; can we confirm this from our simulations on the set of hyperbolic graphs?

3. Is it possible to identify a clear pattern of global graph properties at any percolation threshold?
5.2.2 Hypotheses

1. It is possible to identify a defined transition threshold for percolation as edge density increases from 0 to 1.

2. The activity either percolates completely, or fails to percolate, on each hyperbolic graph.

3. A clear and distinct pattern of global graph properties can be identified at the percolation threshold

5.3 Experimental Design

In this section we describe the main design of our experiments to investigate bootstrap percolation on hyperbolic random geometric graphs. We use agent based simulation, as before, since computational analysis facilitates the simulation of thousands of interactions between multiple nodes, over many time steps, with the aim of obtaining consistent and accurate results.

To investigate our research questions, we perform simulations on the same set of hyperbolic graphs used for our simulations of the Prisoner’s Dilemma, to observe what happens when each network is seeded with an activity. Our goal is to observe the dynamics of the spreading activity, and any changes in this behaviour across the set of networks as the number of edges in the graphs increase.

Bootstrap Percolation Process Definitions

1. Set the parameters
   - Size of the active seed set $A_o$
   - Activation threshold $AT$

2. Define the activation mechanism
   - For all inactive nodes, activate if number of active neighbours is at least $AT$. 
3. Define the process

- For each graph, attach an agent to each node in the graph
- Assign inactive state to all agents
- Select active seed set \( A_0 \)
- Apply activation mechanism at each time step until stopping criterion.

In this experiment we use the same set of 2,400 hyperbolic random geometric graphs, as for our game theory experiments. All of our graphs have 1,000 nodes; our choice of network size allows us to model many interactions, and using the same size for all simulations allows for ready comparison of results. These graphs have increasing edge density from 0 to 1, with 20 graphs at each graph creation parameter \( R \). Our initial experiments simulate bootstrap percolation on each of the 20 graphs at each distance parameter \( R \) to observe emergent behaviours. The process is simulated 1000 times on each graph, with a new randomly selected seed set in each simulation, and the average number of final active nodes at each graph creation parameter \( R \) is recorded.

For each graph in the set of graphs, a population of agents is attached to each graph, one to each node in the graph. All agents are initially inactive, and 20 agents are chosen at random to form the active seed set \( A_0 \), representing a small scale random attack in a network. At each time step, the state of each agent is noted and the activation mechanism is applied synchronously to all nodes in the graph. For each inactive node in the graph, if the number of its active neighbours is at least equal to the activation threshold, the node becomes activated, else it stays inactive. The process of activation continues at each time step until the stopping criterion, which in our case is when an equilibrium is reached, where no further state change is possible. At this stage the number of active and inactive cells in each graph is counted. These results are averaged for all 20 graphs at each distance parameter.

For each graph, the initial activation threshold \( AT \) is set at 2 and a new set of simulations are performed on the same graph, increasing the activation threshold value up to 10, in increments of 1.
5.4 Results

In our simulations, as the number of edges in the set of hyperbolic graphs increased there was a distinct transition threshold for edge density, above which the activity completely percolated on all the graphs and below which the activity failed to percolate on these graphs, see Figure 5-1.

![Figure 5-1: Heat map for standard bootstrap percolation showing number of final active nodes at equilibrium. $A_o = 20$, with $AT$ from 2 to 10](image)

In more detail, the blue colour represents failure to percolate, the red area represents complete percolation of the activity to all nodes in the graph. It can be seen that there is a defined transition threshold between these two zones. As we move right across the x-axis we have increasing activation thresholds from 2 to 10 and our chart shows corresponding increases in the percolation threshold. This is to be expected as a node requires more edges to gain access to a greater number of active nodes as the activation threshold increases.

At the leftmost edge of the percolation threshold, this occurred on graphs created at distance parameter $R = 3.0$, which were the first set of graphs displaying complete
percolation for any activation threshold, in this case activation threshold 2. Graphs at this distance parameter had an average edge density of approximately 0.02 with an average degree of 20 edges per node. At the rightmost edge of the percolation threshold, $R = 5.7$, all simulations on this set of graphs completely percolated for all activation thresholds from 2 to 10. These graphs had an average edge density of approximately 0.1, with an approximate average degree of 100 edges per node.

Within the threshold zone of transition, the variation in colours, neither red nor blue, suggest that the number of final active nodes at equilibrium was between 0 and 1000, in contrast to the clear binary outcomes above and below the threshold. On first reading, this might suggest there was neither complete percolation nor failure to percolate on these graphs, but in reality, the results represent an average of outcomes for the 20 graphs at each distance parameter. On examination, the outcomes for each of these individual graphs was a binary result, either complete percolation, or failure to percolate, the ratio changing in favour of percolation as the edge density increased.

It is also important to note that the giant component emerges at $R = 2.5$, and that the threshold is above this point, this is to be expected as it is only when a giant component is created that all nodes are accessible.

We can also see that the threshold appears to be levelling off as the activation threshold increases. This may be a result of increasing edge density reaching a point where each inactive node has sufficient access to a large enough group of active nodes, via the ever-increasing average degree and the shortening path lengths of the graphs.

### 5.4.1 Graph Properties at Threshold

The properties of interest are component size, density, path length, centralisation measures and clustering, as described in section 2.1. The graphs become minimally connected at distance parameter $R = 2.5$, this equates to an average degree of 13 edges per node. As we are only interested in the spreading behaviour in a connected network, this critical point has been marked on the charts with a red line, and we are particularly interested in the zone between the red line and each of the blue lines, which mark the percolation thresholds for activation thresholds 2 and 10 respectively.
Component Size

Figure 5-2 shows the size of the largest component mapped to distance parameter $R$. In the following charts, the zone of interest is between the red line and the rightmost blue line. The blue lines mark the boundaries of the transition threshold for activation thresholds 2 and 10. Below distance parameter $R = 3.0$, the activity failed to percolate on all graphs. At $R = 5.7$ and above, all graphs displayed complete percolation for all of our chosen activation thresholds. The red line marks the point at which all nodes are reachable by a path from any node.

![Figure 5-2: Size of giant component at varying distance parameter $R$](image)

Density

Figure 5-3 shows density mapped to distance parameter $R$. The graphs become minimally connected at a density of 0.013, which in our graphs of 1000 nodes equates to an average degree of 13 edges per node. The lowest percolation threshold, for activation threshold 2, occurs at a density of 0.019, and average degree of 19 edges per node. The upper percolation threshold occurs at density 0.097, at an average
degree of 97 edges per node.

![Figure 5-3: Edge density at varying distance parameter $R$](image)

**Network average clustering coefficient and transitivity**

Figure 5-4 shows the network average clustering coefficients and transitivity measures at varying distance parameter. For all significant distance parameters, the graphs display high clustering, with a markedly lower value for transitivity. As edge density increases towards the percolation thresholds there is an increasing variation between the two measures, with transitivity decreasing significantly.

This variation in the two measures relates to structural features of the hyperbolic graphs. As edge density increases in the set of graphs, the way these edges are distributed throughout each graph becomes highly irregular, with hub nodes towards the centre and many more nodes of lower degree towards the edges. As the number of edges increases, the network average clustering coefficient increases, yet the high number of nodes of lower degree have less interconnections within their neighbourhood, resulting in lower transitivity values.
Figure 5-4: Clustering coefficients, values in the range 0 - 1.

Diameter and average shortest path

Figure 5-5 shows decreasing path lengths at the percolation threshold as the activation thresholds increase from 2 to 10.

Degree, betweenness and closeness centralisation

In Figure 5-6, the hyperbolic graphs display low skew for degree centralisation in the transition zone, reflecting the low level of variance in node degree throughout each graph. The set of graphs features decreasing betweenness centralisation as the tendency to defection increases. Closeness centralisation is increasingly skewed within the transition zone, indicating an increasing degree of disparity in individual node centrality scores. Note that closeness centralisation could not be calculated for values less than 2.3, as the graph was in several components.
Figure 5-5: Diameter and average shortest path

Figure 5-6: Degree, betweenness and closeness centralisation
5.5 Discussion

Our results have demonstrated that there is a clear state transition threshold as edge density increases in the set of graphs; above this threshold there is complete percolation of the activity and below there is complete failure to percolate. On all of the graphs, there was either almost certain complete percolation or there was complete failure to percolate, with only the active seed set being active at equilibrium. This is quite an interesting result as intuition tells us that there should be some stages at which there would be an in between stage with, for example, 25 per cent or 75 percent of nodes becoming activated. This contrasts with results on small world graphs, where community structure helped preserve cooperation [91]. One could also envisage other graph models where isolated clusters remain resistant to the activity, such as in the corners of Euclidean random geometric graphs, where access to a cluster might be through one or two "gatekeeper" nodes. One explanation for the binary state outcomes lies in the structural properties of the hyperbolic graphs. In general they are typified by skewed degree distribution, with a tendency to central hub nodes and outer leaf nodes, and relatively short path lengths, which shorten as the edge density increases. In essence, the presence of central hub nodes means that any inactive node is usually only a short path length away from an active hub node, and that each inactive node is therefore likely to have lots of contacts in their own neighbourhood or a short step away. Our results also support the probabilistic analysis by Candellero et al. [24] which suggested that on hyperbolic graphs, the activity either failed to percolate, percolated completely or a.a.s percolated.

Analysis of graph centralisation measures as edge density increases in the set of hyperbolic graphs has allowed us to correlate the changing pattern of measures with outcomes in the percolation process. Figure 5-6 shows changing patterns of graph centralisation measures, marked with the threshold boundaries for activation thresholds of 2 and 10. At the upper boundary for activation threshold 10, complete percolation for all activation thresholds of 2 to 10 occurs when degree and closeness centralisation values begin to coincide and become increasingly skewed, with the
associated graph betweenness centralisation displaying low skew, so that there is not much variance in individual node betweenness centrality values. At the upper threshold boundary for activation threshold 2, closeness centralisation predominates and continues to rise, betweenness centralisation is relatively low and decreasing, and degree centralisation is low but rapidly increasing.

With the results from our agent based simulations of bootstrap percolation on the set of hyperbolic graphs, we have identified an approximate edge density at which bootstrap percolation reaches a transition phase, for various activation thresholds and process parameters. We have correlated this threshold with a distinct pattern of global graph properties. This means we have gained some insight into conditions which allow the activity to spread or fail to spread (for activation thresholds from 2 to 10).

As the activation threshold increased from 2 to 10, the heat map at Figure 5-1 shows the transition threshold beginning to level off. One possible explanation is that when the number of edges in the graphs increases past a certain point, this increases the likelihood that any chosen node is able to access one or more hub nodes, which in turn facilitates the spread of the activity, even for higher activation thresholds. In real world terms this represents a population with lots of contacts and short path lengths, so that, even when the activation threshold is increasingly high, individuals are likely to come in contact with a high number of activated individuals.

This analysis of global graph properties compared with percolation outcomes represents an opportunity to exploit these features to alter outcomes, such as inhibiting percolation, perhaps by modifying the rules for state change, allowing particular nodes to recover or by effectively immunising nodes against the percolation process. Exploration of this idea prompted our next sets of experiments, outlined in Chapters 6 and 7 respectively.

When comparing outcomes from bootstrap percolation with those from our earlier simulations of the Prisoner’s Dilemma in Chapter 4, it can readily be seen that both processes have clearly defined transition thresholds. However, the percolation threshold was higher than the game theoretic threshold. The difference is not unexpected,
as the processes are not directly comparable. Both mechanisms for state change are complex, relying on potentially multiple contacts with the activity in question, since both processes rely on the state of a neighbourhood. Bootstrap percolation is a form of complex contagion, where infection is based on multiple contacts with the disease and state change is dependent on the number of active neighbours at each time step. In the game of the Prisoner’s Dilemma, the change of state is dependent on payoffs from individual node interactions, based on a matrix of 4 possible scores, followed by each node selecting the state of their highest scoring neighbour. A further difference is that there is only one possible state change in bootstrap percolation, from inactive to active, whereas in game theory reverse state change is possible at different time steps.

Despite these differences, both processes on the same set of hyperbolic graphs display binary outcomes. In the Prisoner’s Dilemma, this is either the complete adoption of defection or the robustness of cooperation and in bootstrap percolation the activity either completely percolates or fails to percolate. Outcomes for both processes display a clear transition threshold between these zones, and within reasonably close edge densities.
Chapter 6

Bootstrap Percolation with Recovery

6.1 Introduction

In Chapter 5 we observed that, when embedding bootstrap percolation on a set of hyperbolic graphs of varying edge density from 0 to 1, there was a distinct transition threshold above which complete percolation occurred and below which the activity failed to percolate.

Bootstrap percolation was developed to model the physical process of ferro magnetism in materials, where (within reasonable bounds) physical substances will behave similarly under similar circumstances and therefore the percolation model assumes an activation mechanism which is applied equally to all members of the population, and updates occur dependent on the number of activated neighbours. However, when modelling the dynamic processes involved in real world human behaviours in a social context, such as brand adoption or voter trends, we cannot presume to determine the precise impact of peer influence, even if we know the structure of the underlying social network. By modifying the rules we may gain insight into situations where individual agents do not follow the rigid bounds of standard bootstrap percolation modelling.

Our previous simulations concentrated on observing the impact of network structure on spreading behaviours. In this chapter we switch our focus to the process itself, to look at how modifications to the rules of the process will impact upon the
spread on a network; this is then tied into our previous research by investigating any
correlation between structural features of the network and the effect of these rule
changes.

Our assumption is that the standard form of bootstrap percolation will model
situations where the spread of an activity is deterministic, so that under set circum-
stances, each node will obey the rules. Introducing modifications to the standard
rules allows us to introduce a stochastic element, which might more closely model
real world scenarios.

In the standard form of bootstrap percolation the only possible state change is
from inactive to active, no reverse state change is allowed. Our proposed modification
allows for reverse state change, in effect allowing active nodes to recover. This modi-
fication maintains the fundamental nature of bootstrap percolation, which essentially
allows an activity to spread once the number of users reaches a tipping point, but the
recovery element introduces flexibility in the subsequent spread of the activity.

In brand adoption, the recovery element might represent a change of mind sce-
nario, where the activity spreads in a population with a great degree of certainty once
a tipping point is reached, but individual users may or may not continue to use that
technology. We explore the impact of this modification on the general population,
using our set of hyperbolic random geometric graphs to test various hypotheses.

The hyperbolic graphs are typified by distinct topological features, described in
Section 3.3.2, we can further investigate if utilising these properties to target recovery
has any greater effect than randomly selecting nodes to recover.

Previous research involved a probabilistic analysis of a form of bootstrap percola-
tion with recovery on lattice grids [28] but, to the best of our knowledge, this modified
process has not been analysed mathematically or simulated on hyperbolic graphs.

Introducing a recovery element to the bootstrap percolation process bears some
similarity to the SIR model in epidemiology, described in Section 2.4.3. That model
assigns members of a population to compartments to model the spread of contagion.
One of the main issues with that method is that each individual within a compartment
is homogeneous, with each agent having the same chance of changing state. There
have been some attempts in epidemiology to modify the compartmental model to allow for differences in node degree, by introducing a node degree compartment, see for example [97, 13]. However, this approach cannot recognise an individual node’s position within the graph and its effect on global outcomes; similarly this does not take into account clustering. In spatial bootstrap percolation, agents attached to complex networks, such as hyperbolic graphs, are heterogeneous. The spread of contagion will depend on the state of the agent, and of the state of agents within the neighbourhood of that node. This means that we can allow more individual complexity for each agent and observe the effect that local topology has on emergent behaviour in the global population.

For our simulations, we use the same set of hyperbolic graphs, with the same parameters for bootstrap percolation as in the previous chapter, to allow direct comparison of process outcomes at the same activation threshold parameters. We again model a situation with a small scale random seeding of an activity in a network, however, in this case allowing for reverse state change. The idea is that at each time step after the activation mechanism has been applied, a certain percentage of all active nodes is allowed to “recover”, i.e. change state back to inactive state. Outcomes from these simulations are then compared with simulation outcomes from standard bootstrap percolation on the same set of graphs. In Chapter 3, we noted that the hyperbolic graphs typically display power law degree distribution, with most nodes having low degree and a few nodes having high degree. Therefore, we have chosen to target nodes of high degree to recover, to assess the impact these nodes have in the graph, compared with a set of control simulations where active nodes are selected at random.

Section 6.2 describes our research questions and hypotheses, and our conceptual framework for Bootstrap Percolation with Recovery is described in Section 6.3. The experimental set up is outlined in Section 6.4, with results presented in Section 6.5 and discussed in Section 6.6.

The results described in this chapter have been published in [73].


6.2 Research Questions and Hypotheses

In our overall research we are interested in investigating the impact of spatial structures in the network upon a spreading behaviour. Our goal is to identify which features of the hyperbolic networks promote spreading behaviours and which inhibit the spread of activities.

In this chapter we expand the scope of our research by investigating if some small modification to the rules of a spreading activity might allow us to enhance the spatial effects, by further facilitating or inhibiting the spread of the activity, and potentially offer more realistic insights into the dynamic modelling of real world scenarios. In standard bootstrap percolation, the only possible state change is from inactive to active state. We have developed a modified version of bootstrap percolation which allows for the recovery of a defined percentage of active nodes after each activation step; by recovery we mean the transition from active state back to inactive state.

The focus of our investigation is to observe any impact of these modifications on the spread of the activity and whether applying these changes to nodes of high degree will have a greater effect than random application.

6.2.1 Research Questions

1. If we modify the rules of bootstrap percolation to allow reverse state change, will this have any impact on percolation outcomes and on the percolation threshold?

2. Given that power law degree distribution is a typical feature of hyperbolic random geometric graphs and that hub nodes are commonly regarded as highly influential in spreading behaviours on networks; if we target recovery at nodes of high degree, will this have a greater inhibitory effect than random selection of active nodes for recovery?
6.2.2 Hypotheses

1. Allowing reverse state change in bootstrap percolation will inhibit percolation and raise the percolation threshold to graphs of higher edge density.

2. Targeting active nodes of high degree for recovery will have greater inhibitory effect on the percolation process than random selection.

6.3 Process Definition for Bootstrap Percolation with Recovery

Our proposed method involves setting up the experiments as for the standard bootstrap process, but introducing a recovery mechanism to follow the activation mechanism in each time step. The following chart outlines the main steps involved in defining the procedure.

6.4 Experimental Simulation of Bootstrap Percolation with Recovery

Our simulations involve embedding the modified form of bootstrap percolation on the set of hyperbolic random geometric graphs, and comparing outcomes with those from standard bootstrap percolation. Bootstrap percolation with recovery is simulated targeting a percentage of active nodes of high degree for recovery at each time step and the simulations are then repeated with random selection.

The bootstrap percolation process is modified to allow for reverse state change from active to inactive, as outlined in Section 6.3, by introducing a recovery phase in each time step immediately following activation. Our first set of experiments involves the random selection of active nodes for recovery. The experiments are then repeated targeting active nodes of high degree for recovery.

The initial set up is the same as in the standard bootstrap percolation process. A
**Procedure**  Bootstrap Percolation with Recovery

1: Define Parameters

- Size of the active seed set $A_o$
- Activation threshold $AT$, an integer value
- Recovery rate $RR$ percentage

2: Define Mechanisms

- Activation mechanism
  - For all inactive nodes, activate if number of active neighbours is at least $AT$.

- Recovery mechanism
  - For all active nodes, select $RR\%$ to deactivate, by random or deterministic selection

3: Define Process

- For each graph, attach an agent to each node in the graph
- Initialise all agents to be inactive
- Select active seed set
- Apply activation mechanism followed by recovery mechanism at each time step until stopping criterion
population of agents, engaged in the process of bootstrap percolation, is attached to each graph. An agent is assigned to each node, with all agents initially inactive and, from these, 20 nodes are randomly selected to form the initial active seed set, \( A_0 \). The activation mechanism involves an activation threshold \( AT \) whereby an inactive node becomes active if it has at least \( AT \) active neighbours. The recovery mechanism involves selecting a percentage of the active nodes for deactivation.

At each time step, the activation mechanism is applied synchronously to all nodes in the graph followed by the recovery mechanism applied to all active nodes. This cycle of activation and recovery at each time step is repeated until equilibrium; the number of active nodes is recorded at this point. Our experiments are then repeated at this Recovery Rate varying the activation threshold \( AT \) from 2 to 10. We then repeat the experiments incrementing the recovery rate by 10% for each set, from 10% to 90%.

Our initial simulations showed that equilibrium was reached rapidly for all graphs, and we therefore set an arbitrary cut off point at 100 time steps, which ensured that equilibrium is reached in each simulation. Therefore our stopping criterion was based on time steps. In the case of graphs which had complete percolation in the standard bootstrap process, equilibrium cycled between full activation and deactivation of the recovery rate fraction. In retrospect, our stopping criterion could have been based on reaching full activation, however we were unsure what the likely outcome would be, and wished to allow for an observation of long term behaviour.

### 6.4.1 Random recovery

After each activation time step, the recovery rate percentage of active nodes is randomly selected to become inactive.

### 6.4.2 Targeted recovery based on node degree ranking

After each activation time step, active nodes were ranked according to node degree, from highest to lowest, the recovery rate percentage of the top ranked active nodes is
then selected to become inactive.

6.5 Results

In this section we describe our results from simulating bootstrap percolation with random recovery and then targeted recovery, all on the same set of hyperbolic random geometric graphs, using averaged data from the 20 graphs created at each parameter $R$.

In all of our simulations of bootstrap percolation with recovery, the heat maps displayed a clear percolation threshold. For comparison purposes, we developed a simple algorithm to find a representative point within the threshold, to allow for representation of the percolation threshold as a curve. This involved finding the midpoint in the threshold zone at each activation threshold parameter.

6.5.1 Bootstrap Percolation with Recovery

On the following charts, the standard Bootstrap Percolation process is depicted as a case of 0% recovery. Each fence plot has a differing recovery rate percentage ($RR\%$) and represents the percolation threshold for that particular recovery rate. It can be clearly seen that either form of recovery significantly raises the threshold for complete percolation of the activity and that, in general, each increase in $RR\%$ increases this stepwise.

Bootstrap percolation with random recovery

Figure 6-1 shows the results for bootstrap percolation with recovery randomly targeting a percentage of active nodes following activation in each time step. Each curve represents results for increasing activation thresholds from 2, on the left to 10, on the right. As expected, each curve rises from left to right, because any increase in activation threshold means that more active contacts are required for an inactive node to become activated.
It can be seen that overall, the edge density of the threshold increases as the recovery rate increases from 10% to 90%, with the form of each curve similar to its neighbour. However, after the initial increase at 10%, increasing the recovery rate had little further effect before 50% recovery.

For an activation threshold of 2, these values for distance parameter $R$ at the percolation threshold increased from 2 to 4.7, representing edge densities ranging from 0.01 to 0.055. For an activation threshold of 10, the values for $R$ at the percolation threshold increased from 5.2 to 8, representing edge densities from 0.075 to 0.3.

![Bootstrap Percolation with RANDOM recovery](image)

**Figure 6-1:** Bootstrap percolation with random recovery

**Bootstrap percolation with targeted recovery**

Figure 6-2 shows bootstrap percolation with recovery selectively targeting the top ranked active nodes of highest degree after each activation period. This clearly shows that the corresponding edge density of thresholds has increased significantly for all parameters, when compared with randomly selected recovery. Unlike random recovery,
where the increase in recovery percentage had no effect between 10% and 50%, the percolation threshold curves for targeted recovery showed a regular stepwise increase as the recovery percentage increased, for all percentages.

The most notable observations, with each percentage increase in recovery rate, was that for an activation threshold of 2, the percolation threshold occurred at values of $R$ between 2 and 4, representing edge densities in the range 0.01 to 0.037. For activation threshold 10, the values for $R$ at the percolation threshold increased from 5.2 to 7.6, representing edge densities of 0.075 to 0.25. In both cases, the form of each curve was similar to its neighbour. This demonstrates that selectively targeting the nodes of highest degree for recovery has had a significantly greater impact on the spread of the activity compared with random selection, and particularly when compared with the standard bootstrap process.

Figure 6-2: Bootstrap percolation with targeted recovery
Comparison of Random and Targeted Recovery

The following figures illustrate the outcomes at each activation threshold for random selection, compared with targeted selection of active nodes to recover at each time step. It can be seen that with an activation threshold of 2, the percolation threshold was lower for targeted recovery, for all percentages of recovery. Similarly for all other activation thresholds. This essentially means that targeted recovery had a greater delaying impact on percolation than random recovery.

Figure 6-3: Percolation threshold values for random and targeted recovery at activation thresholds (AT) 2 to 7, with varying recovery rates from 10 to 90%
Figures 6-3 and 6-4 show comparison of threshold values for activation thresholds 2 to 7, and 8 to 10, with varying percentage recovery rates. In general targeting recovery at nodes of high degree has a greater delaying impact on percolation than random selection. At activation threshold 2, where a node needs two active neighbours to become activated, at 10% recovery rate, both targeted and random recovery had a similar delaying impact on percolation from an average degree of 5 edges per node to 8. However for all other recovery rates, targeted recovery had a significantly greater delaying impact on percolation, when compared to random. At 20% recovery the threshold value was 8 edges per node compared with 10 for targeted recovery. In graphs of 1,000 nodes this represents an increased delay in percolation from networks of 8,000 edges for random recovery, to 10,000 edges for targeted recovery. Interestingly, random recovery had no further increase in delay until 50% recovery rate, meaning that the delaying effect accelerated between 20% and 50%.

For activation threshold 10, with 10% recovery rate, both random and targeted recovery had the same significant delaying impact on percolation from 49 edges per

Figure 6-4: Percolation threshold values for random and targeted recovery at activation thresholds (AT) 8 to 10, with varying recovery rates from 10 to 90%
node to 73, above this rate, targeted had a greater delaying impact on percolation than random. As an example, at 20% recovery rate the average degree values were 73 for random and 83 for targeted recovery at active nodes of highest degree.

6.6 Discussion

We previously simulated bootstrap percolation on hyperbolic geometric graphs and noticed that there was a clear threshold for edge density above which the activity percolated to all nodes and below which the activity failed to percolate. In our earlier analysis of graph properties in the hyperbolic graphs, we noted particular significant features such as high clustering and power law degree distribution. Our intention was to see if we could exploit these features and alter outcomes in the percolation process. In this chapter we have developed a modified form of bootstrap percolation which allows for recovery from active to inactive state. This modification augments the system while maintaining the abstract nature of bootstrap percolation, i.e. modelling the spread of an activity when the number of users reaches a tipping point. Our interest is in observing any effect this modification has on the percolation threshold in the set of hyperbolic graphs, and if targeting recovery at active nodes of high degree has any greater impact than random selection of recovery nodes.

Our results show that introducing recovery has an inhibitory impact on the percolation process, by delaying the complete percolation in graphs which previously rapidly percolated. In fact, for some percolating networks at the threshold we have completely reversed the outcomes by introducing recovery. This threshold now occurred at a higher edge density, meaning that graphs which had previously percolated now failed to percolate and it required additional edges to allow complete percolation. We have further demonstrated that targeting nodes of high degree for recovery has a greater delaying effect on percolation than random recovery, this confirms that hub nodes are highly influential in the percolation process on this set of graphs.

One limitation of the study was choosing the same time frame as the standard bootstrap simulations; whilst this allowed direct comparability between outcomes
from different sets of simulations, it did not allow us to determine whether any of the non-percolating graphs may have percolated given a longer time period.

Standard bootstrap percolation is a deterministic process, where choosing the same set of active seeds will always lead to the same outcome. This is an effective method for modelling the spread of an activity in a network, such as the process of ferro-magnetism in materials where, within reasonable bounds, the activity will progress uniformly for all nodes in the material, dependent on activation of adjacent nodes. In this section we have introduced a modification of the bootstrap percolation process, which allows for recovery of active nodes to inactive state. This approach might offer insights into real world scenarios, where adoption of an activity by a real population is not as well defined, or deterministic, as the behaviour modelled in the standard bootstrap percolation process. This approach also allows for the introduction of a stochastic element, particularly in the case of random recovery.

A key feature of this work is observing the effect of targeting active nodes of high degree for recovery, which has a greater inhibitory effect on percolation than random recovery. We selected nodes of high degree as they are commonly seen as nodes of influence in networks and these simulations confirm this in our networks. This is interesting as the hyperbolic graphs are generally characterised by power law degree distribution, they are highly clustered with a few large hub nodes and most nodes in the graph have low degree. It is therefore interesting to speculate that other features of the hyperbolic graphs might be exploited to impact the percolation process.

It may be possible to fine tune our targeting, by choosing differing node properties or by targeting nodes highly ranked for a combination of properties. It would also be interesting to specifically target the most influential nodes in the network, with a priori rankings, so that whenever such a node became active it would always revert to inactivity. This would effectively immunise that node from activity and would represent an influential player constantly resisting an onslaught of activity.

In the next section we investigate the effect of targeted immunity, where selected nodes are effectively removed from the standard bootstrap percolation process. Node selection is based on a priori ranking for properties which are significantly skewed
in the hyperbolic graphs, such as clustering coefficient, betweenness centralisation and closeness centralisation. Our intuition is that finding measures which are highly skewed in the hyperbolic graphs will allow us to significantly delay percolation.
Chapter 7

Bootstrap Percolation with Immunisation

7.1 Introduction

This section extends the research from the last two last chapters. In Chapter 5, Bootstrap Percolation, we demonstrated that with a small scale random seeding of an activity in a network, on a set of hyperbolic graphs, there is a distinct transition threshold for edge density above which the activity percolates and below which the activity fails to percolate. In Chapter 6, Bootstrap Percolation with Recovery, we explored the idea of modifying the rules of bootstrap percolation to allow a percentage of active nodes to recover at each time step, which had a delaying effect on percolation. Our aim was to inhibit percolation by targeting nodes of high degree, compared with random selection of nodes to recover. Our results from simulations with both targeted and random selection of recovery nodes showed that the percolation threshold occurred at a higher edge density in the set of hyperbolic graphs when compared with outcomes from standard bootstrap percolation. We observed that targeted node recovery had a greater delaying effect on percolation than random node recovery. In the simulations described in this chapter our goal is to enhance this inhibitory effect on percolation, returning to the standard form of bootstrap percolation, by specific targeting of nodes to be immune to the percolation process. Instead of allowing a
percentage of active nodes to recover, we investigate outcomes when certain nodes are chosen to be unaffected by the percolation process, essentially granting them immunity from percolation.

In the last chapter, our rationale for selecting nodes of high degree for recovery was that node degree is often regarded as the standard measure of influence in networks. Nodes with a large number of connections are hub nodes and can be viewed as "superspreaders" in some processes due to their direct links with more sections of the graph than nodes of low degree. The hyperbolic graphs typically display power law degree distribution, which means that most nodes have low degree with a small number having high degree. Work in our last section confirmed that exploiting this feature had a significant impact on the embedded process of bootstrap percolation.

In real world complex networks, there is a lot of interest in determining influential nodes in networks, for example identifying social media influencers to broaden the reach of brand messaging. Measures of network influence are often based on centrality measures such as degree, betweenness and closeness centrality. Degree centrality is frequently chosen, as this value is determined by the number of links incident to a node, and high degree represents a node with many contacts. High betweenness centrality represents nodes which may have a "brokerage" role in a network. High closeness centrality represents nodes that are closely linked to important nodes while not necessarily having high degree or betweenness. Our approach is aiming for an a priori intelligent selection of influential nodes, by investigating any correlation with graph properties that are highly skewed in the hyperbolic graphs such as high closeness, and high clustering. Our intuition is that properties with highly skewed distributions will maximise the inhibitory effect.

When modelling bootstrap percolation, people are often interested in determining which nodes to choose, or how many, to maximise the spread of an activity, for example who to target in viral marketing to optimise the spread. Our own focus is on inhibiting this spread, and observing which nodes are most influential in obstructing the activity; there has been little empirical investigation of this nature on hyperbolic networks, which share many features with real world complex networks. This might
model the spread of fake news in a social network, where we wish to identify “responsible” nodes, which will take care not to share disreputable information. This might equally apply to a network of mutual funds, where there is panic selling based on rumour, certain influential nodes could be targeted to hold firm against the spread of fake information. This work is related to the study of immunisation strategies in epidemiology, and other inhibitory actions in networks, such as preventing the spread of malware in computer networks. While all are interested in identifying the best nodes to inhibit the spread of an activity, there are different underlying assumptions. In epidemiology and the study of computer viruses, the specific infection must be identified in order to introduce immunity by applying a vaccine or anti-viral software. In bootstrap percolation, modelling social influence in spreading activities, the message/activity may never previously have been encountered; in our work, the inhibitor node receives the message, but resists peer pressure, and takes no part in spreading the activity.

Our work complements existing theoretical work on identifying influential nodes in networks and outlines further avenues in an empirical manner. Our proposed method for identifying these nodes involves agent based modelling of bootstrap percolation on the previously created set of hyperbolic random geometric graphs. Our particular interest is on the set of graphs at the percolation threshold, to identify certain properties of these graphs that might facilitate or impede percolation.

As previously noted in Chapter 3, Hyperbolic Random Geometric Graphs, these graphs are typified by skewed distributions for certain node centrality measures, including degree, closeness and betweenness centrality. Global graph centralisation values for these centrality measures varied throughout the set of hyperbolic graphs, described in Section 3.3.2. Although hyperbolic graphs are generally typified by power law degree distribution, for very high values of distance parameter $R$, with edge density approaching 1, the skew in degree distribution began to level off and the global values for degree centralisation approached that of the Erdős Rényi model, where most nodes have similar number of edges. Therefore, we have taken a particular interest in analysing processes on hyperbolic graphs within a specific range for edge
density, between distance parameter $R = 2.5$, where all nodes are reachable and approximately $R = 6$, which contained the upper bounds for our percolation threshold for activation thresholds of 2 to 10.

In light of these changing patterns of global graph centralisation measures, we theorised that specific targeting of the top ranked nodes for these highly skewed measures might have a significant impact if we effectively removed these nodes from the process by immunising them. Our work in this section therefore focuses on inhibiting the percolation of the activity by selecting strategic nodes to immunise. Our motivation, given a small scale random attack in a network, is to select a small number of nodes to immunise against the activity, which would obstruct the percolation process and effectively grant a form of herd immunity.

The results described in this chapter have been previously published [72].

7.1.1 Immunisation and Influential Nodes

In Section 2.4.4 we described the concept of immunity in a spatial network, noting different studies which identified various centrality measures as influential in facilitating or obstructing the spread of an activity in a network.

Our own focus is based on a long term goal of being able to make an intelligent selection of influential nodes based on global network properties. With this in mind, our use of the previously created set of hyperbolic random geometric graphs which display distinctive patterns of centrality measures as edge density increases, aids in mapping the relationship between the chosen node property and the inhibitory effect.

Section 7.2 describes our research questions and hypotheses, and our experimental set up is outlined in Section 7.4. Our results are presented in Section 7.5 and discussed in Section 7.6.

7.2 Research Questions and Hypotheses

In Chapter 5 we observed a transition threshold for percolation outcomes, in the set of hyperbolic graphs, between graphs displaying failure to percolate and complete
percolation respectively, these latter graphs we define as percolating networks. Our basic interest is in observing what happens in a percolating network if we select a small number of nodes to be immune to the percolation process. We wish to observe the extent of any inhibitory effect on percolation from targeted immunity, when compared with random selection of the immune set. It is our hypothesis that selecting our immune set based on graph properties which display skewed distributions in the hyperbolic graphs, is likely to result in an inhibitory effect on percolation, when compared with random selection and also with standard bootstrap percolation.

Hypotheses

1. Targeting nodes with high centrality measures to become immune to the percolation process will have greater effect than random selection of immune nodes.

2. Targeting nodes that are top ranked for properties that have highly skewed distributions within each graph will maximise the inhibitory effect.

7.3 Framework for Identifying Influential Nodes to Inhibit Bootstrap Percolation on Hyperbolic Networks

Figure 7-1 shows our conceptual framework for identifying influential nodes. A hyperbolic graph is created and analysed for a variety of graph properties. A set of nodes in the graph is chosen to be immune to the bootstrap percolation process, neither active nor inactive, and may not change state throughout the process. This choice may be at random or based on various graph properties of interest. The process of bootstrap percolation is simulated on the graph, and then repeated on the same graph with the introduction of the immune set. The outcomes from both simulations are compared, which might offer insights into which set of nodes might have the most influence when effectively immunised against the bootstrap percolation process.
7.4 Experimental Set Up

Our experimental design is based on the framework outlined in Section 7.3, using the previously generated set of hyperbolic graphs which were analysed for various graph properties in Chapter 3. Using the same set of graphs allows direct comparison across the different sets of simulations, and keeping the number of nodes fixed allows us to reduce complexity and present clearer results.

Figure 7-2 illustrates the results from our simulations of standard Bootstrap Percolation in Chapter 5, Figure 5-1 reprinted here for ease of viewing. As the number of edges in a set of hyperbolic graphs increased there was a distinct transition threshold above which the activity completely percolated on all the graphs and below which the activity failed to percolate on these graphs.

In this section, we are particularly interested in carrying out our experiments on selected graphs at the threshold, because it is here that there is a clear transition between failure to percolate and complete percolation, with this transition occurring
Figure 7-2: Heat map for standard bootstrap percolation showing number of final active nodes at equilibrium. $A_0 = 20$, with activation thresholds from 2 to 10. Figure 5-1 reprinted here for ease of viewing.

within a relatively small increase in edge density. Therefore if we choose a percolating network at the threshold it is likely that any impact from the introduction of immune nodes will be obvious. Our simulations for comparison of bootstrap percolation with bootstrap with immunity are therefore performed on hyperbolic random geometric graphs created at each of two distance parameters $R$, specifically those at both edges of the upper boundary of the percolation threshold shown in Figure 7-2. At the rightmost edge, $R = 5.7$, all simulations on this set of 20 graphs completely percolated for all activation thresholds from 2 to 10. At the left edge, all 20 graphs created at distance parameter $R = 3.0$ were the first set which displayed complete percolation at activation threshold 2. These represent edge densities of approximately 0.1 and 0.02 respectively. Of the 20 graphs at each of these 2 parameters, we have selected graph 13, at random, for our simulations, which we denote by graphs $R = 3.0_{13}$ and $R = 5.7_{13}$, at the left and right ends of the threshold, respectively. In summary, in the previous chapters we were interested in spreading behaviour across the set
of hyperbolic graphs, we are now interested in observing what happens within a particular network when we introduce immune nodes.

### 7.4.1 Simulation of Bootstrap Percolation

For this set of experiments we have simulated standard bootstrap percolation on the selected graphs at the transition threshold, using the parameters already described in Chapter 5.3. For each graph, a population of agents is attached to each graph, one to each node. All agents are initially inactive, and 20 agents are chosen at random to form the active seed set $A_o$, representing a small scale random attack in a network. The initial activation threshold is set at 2 and, at each time step, inactive nodes with at least this number of active neighbours are activated. This mechanism is repeated for each time step until an equilibrium is reached, where no further state change is possible. The simulations are repeated on this graph increasing the activation threshold up to 10, in increments of 1. This is repeated for 1000 simulations at each activation threshold parameter, each with a different randomly selected seed set $A_o$. The final number of active nodes at equilibrium is recorded, and the outcome for each simulation is grouped in sets of one hundred, from 0 - 100 up to 901 - 1000; the number of simulations containing outcomes within each grouping is recorded.

### 7.4.2 Simulation of Bootstrap Percolation with Immunisation

This set of simulations has the same design as the standard bootstrap percolation set up, as described above in Section 7.4.1, however an immunised seed set $I_o$ is selected beforehand to be immune to the percolation process, neither active nor inactive, and remaining immune throughout the simulation.

There are many potential settings for $I_o$; we wish to identify a setting sufficiently large to capture the introduction of nodes with a high impact on delaying percolation, to the point where additional nodes had a diminishing impact. Our initial experiments involved an immunised seed set of 20 immune nodes, the same size as the active seed set $A_o$. With 20 immune nodes we noted that the rate of decline in percolating
simulations appeared linear. By increasing $I_o$ incrementally to 25, we noted that the rate of decline levelled off for values over 20 (see Figure 7-6). Therefore we chose $I_o$ of 25 nodes as this gave a more complete picture of the decline in percolating simulations as the number of immune nodes increased.

**Selection of Immune Nodes**

Selection of $I_o$ is performed in two ways, by targeted selection based on various node properties, and by random selection for comparison.

- **Random selection**
- **Targeted selection**
  - Degree centrality
  - Betweenness centrality
  - Closeness centrality
  - Watts Strogatz Clustering coefficient

Of the many options for centrality measures, we have chosen degree, betweenness and closeness as these have distinct and varying levels of skewed distribution in the set of hyperbolic graphs, see Section 3.3.2. This facilitates tracking the influence of each property to determine if there is any correlation between the level of skew of the centrality measure and the potential inhibitory effect of selecting nodes highly ranked for this property. We have ranked all nodes in our selected graphs for these values, in descending order, and we choose the top ranked values for immunity.

In Section 3.3.2, we noted that many of the hyperbolic graphs were typified by high clustering. We use the Watts Strogatz Clustering Coefficient, the network average clustering coefficient, as our clustering measure for this set of simulations, this measure is the average of the local clustering coefficients over all nodes in the graph. We select nodes based on high, and then low, local node clustering values.

All nodes in the graph are a priori ranked for the chosen property and the top 25 highest ranked nodes are selected for immunity. These nodes remain immunised.
and cannot change state. The experiments are repeated for 1000 simulations at each activation threshold from 2 up to 10.

7.5 Results

7.5.1 Upper boundary of percolation threshold for activation threshold of 10

The outcomes for all 20 graphs at this parameter were similar. To clearly illustrate these outcomes we have charted the results from one particular graph randomly chosen at this parameter, \( R = 5.7_{13} \), as described in Section 7.4. On this graph, we compare standard bootstrap percolation outcomes with outcomes from random selection of immunised nodes and then with targeted selection of immunised nodes.

Figure 7-3 represents outcomes on one graph for 1000 simulations of bootstrap percolation at each activation threshold. This one representative graph is at the upper edge of the percolation threshold for activation threshold 10. The only thing varying in each simulation is the random selection of 20 active seeds in \( A_o \). Each grouping along the x-axis represents the number of final active nodes at equilibrium. The height of each column represents the number of simulations that had outcomes within that grouping.

In the standard bootstrap percolation simulations, see figure 7-3a, all simulations completely percolated, confirming the initial results shown in the previous heat map.

Figure 7-3b represents outcomes for randomly selected immune nodes showing that random immunity had no impact on delaying percolation, with all simulations completely percolating.

Figure 7-4a shows outcomes for 1000 simulations with the top ranked 25 nodes of highest degree selected for immunisation. There was no impact on percolation for activation thresholds from 2 to 5; all simulations completely percolated, as before. However, it is clear that targeted selection of hub nodes has a significant impact on the percolation process for activation thresholds 6 to 10. In standard bootstrap
percolation, all simulations at activation threshold 10 completely percolated. With targeted immunity, approximately 90% of the simulations now failed to percolate. This suggests that these 25 nodes selected for immunity are highly influential in the percolation process.

To assess if any particular node in $I_o$ had a marked contribution to the delaying effect of immunity, we analysed percolation outcomes as the number of immunised nodes increased from 1 to 25. We found that no node had any outstanding contribution to the delaying effect on percolating simulations, see Figure 7-6. As the number of immunised nodes increased, the effect was cumulative with the rate of decline in percolating simulations following a logistic curve.
Figure 7-4: Graph 5.7_{13}. Comparison of outcomes for Immunity, with 1000 simulations at each activation threshold
Figure 7-5: Graph 5.713. Comparison of outcomes for Immunity, with 1000 simulations at each activation threshold

Figure 7-4b and Figure 7-5a also show similar patterns of delayed percolation when nodes top ranked for high closeness centrality values, and low local node clustering coefficient values were respectively targeted for immunity to the percolation process. This result is not surprising; during a priori node selection for immunity, it was noted that the set of nodes top ranked for highest degree were almost identical to the set of nodes top ranked for high closeness, and also for low clustering.

Figure 7-4c also shows a marked decline in percolation outcomes following the introduction of an immune set based on top ranked nodes for high betweenness values. However, the impact was less effective than results from the simulations using
high closeness, high degree and low clustering for activation thresholds 8 to 10. At activation threshold 8, these latter 3 sets displayed only 600 percolating simulations compared with 900 for the high betweenness simulations. At activation threshold 9, the 3 sets displayed only 300 percolating simulations compared with 650 for the high betweenness simulations. At activation threshold 10, the latter 3 sets displayed only 100 percolating simulations compared with 300 for the high betweenness simulations. These results means that targeting nodes of high betweenness for immunity had a substantially less inhibitory effect on percolation than the other three measures.

Again, this result is not surprising; while 19 of the nodes top ranked for high closeness, high degree and low clustering are included in the set of nodes top ranked for high betweenness, this latter set lacks 4 of the highest ranked nodes in the other sets.

Figure 7-6: Rate of decline in percolating simulations as number of immunised nodes increases from 0 to 25 (immunity applied to top ranked nodes of highest degree)
7.5.2 Upper boundary of percolation threshold for activation threshold 2

All graphs at this edge density parameter had similar outcomes. For clarity, results from the selected representative graph at $R = 3.0_{13}$ are displayed in Figures 7-7 and 7-9.

Additionally, results for activation thresholds of 7 to 10 are not included as all simulations at these parameters failed to percolate in standard bootstrap percolation, and immunity therefore could not have affected outcomes.

The results for simulations with Targeted Immunity on graphs at $R = 3.0$ are more varied than for graphs at $R = 5.7$. As before, random immunity had no impact when compared to outcomes from the standard bootstrap percolation process, see Figure 7-7b. However, when we selected the top ranked nodes for different graph properties for immunity, we had varying outcomes, see Figures 7-8a to 7-8c. In these plots, the greatest decline in percolation is most clearly seen by observing the data for activation threshold 4, in green. The delaying effect can be ranked as follows, with random immunity and immunisation of nodes top ranked for high clustering having no effect, when compared with standard bootstrap percolation; the greatest delaying effect was seen when nodes were selected for immunity based on top rankings for high closeness.

The rate of decline in percolating simulations is linear, with no immunised node having stronger effect than any other immunised node.

- Ranked effect of node properties on delaying percolation, from highest to lowest:
  - High Closeness
  - Low Clustering (Local node clustering coefficient)
  - High Betweenness
  - High Degree
  - Random Immunity, High Clustering (Local node clustering coefficient)
  - Standard Bootstrap
Figure 7-7: Graph 3.013. Comparison of outcomes for standard bootstrap percolation, random immunity and immunity targeted at nodes of high degree, with 1000 simulations at each activation threshold.
Figure 7-8: Comparison of percolation outcomes for immunity targeted at nodes of high degree, high betweenness and high closeness, with 1000 simulations at each activation threshold.
Figure 7-9: Graph 3.013. Comparison of percolation outcomes for immunity targeted at nodes of high and low clustering, with 1000 simulations at each activation threshold.
This is an interesting result, as hub nodes are commonly seen as the standard influential nodes in a network. This prompted an analysis of global graph properties at the upper boundary of the percolation threshold.

### 7.5.3 Graph properties at the upper boundary of the percolation threshold, for activation thresholds of 2 and 10

Analysis of the representative graphs at the upper boundary of the percolation threshold are presented in Table 7.1, the global graph properties for each were similar to the other graphs at their respective edge densities. Figure 7-10 shows our selected graph centralisation measures as the number of edges increases in the set of graphs, with selected threshold graph parameters marked as blue lines.

#### Table 7.1: Global graph properties at the upper boundary of the percolation threshold

<table>
<thead>
<tr>
<th>Graph Properties</th>
<th>Graph 3.0\textsubscript{13}</th>
<th>Graph 5.7\textsubscript{13}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Density</td>
<td>0.019416</td>
<td>0.098962</td>
</tr>
<tr>
<td>Average Degree</td>
<td>19.416</td>
<td>98.962</td>
</tr>
<tr>
<td>Diameter</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>Watts Strogatz Clustering</td>
<td>0.7306781</td>
<td>0.78016659</td>
</tr>
<tr>
<td>Transitivity</td>
<td>0.55326337</td>
<td>0.4753645</td>
</tr>
<tr>
<td>Degree Centralisation</td>
<td>0.03970303</td>
<td>0.63995659</td>
</tr>
<tr>
<td>Closeness Centralisation</td>
<td>0.33503071</td>
<td>0.60320727</td>
</tr>
<tr>
<td>Betweenness Centralisation</td>
<td>0.21560217</td>
<td>0.12449812</td>
</tr>
</tbody>
</table>

It is interesting to note that the network average clustering coefficient is very highly skewed in both graphs, which might suggest that nodes with high local node clustering would be influential in the percolation process. However, our results show that it is nodes of low local clustering values that have more impact than high clustering. This effect could, in part, be due to the nature of the hyperbolic graphs, which typically display central hub nodes (high degree) with leaf nodes towards the boundary. These hub nodes generally have low local node clustering values. The disparity of the number of edges in the hubs, compared with other nodes, means that transitivity, i.e.
closing a triple, is less likely to occur. In fact the hyperbolic graphs display a marked difference in the values for the network average clustering coefficient and transitivity, with the latter significantly lower than the former.

![Hyperbolic Graph Centralisation Measures](image)

**Figure 7-10:** Hyperbolic Graph Centralisation compared with percolation thresholds.

### 7.5.4 Discussion

In graphs at the upper boundary of the percolation threshold for activation threshold 10 \((R = 5.7)\), all simulations had previously percolated at all activation threshold values in standard bootstrap percolation. After targeting 25 nodes of high degree for immunity, this had a significant impact on delaying percolation, with 90% of simulations now failing to percolate at activation threshold 10. This set of 25 nodes also had the highest values for closeness centrality and for low clustering. These targeted measures matched our expectations based on global graph properties. However, for graphs at the upper boundary of the percolation threshold for activation threshold 2 \((R = 3.0)\), the results were more ambiguous.

In graphs at \(R = 3.0\), the impact on percolation was more varied, with nodes of
high closeness having most impact, followed by high degree, low clustering, and high betweenness. This does not match our expectations based on the degree of skew seen in the graph properties. The greatest skew was seen in clustering coefficients, followed by closeness and then betweenness, with the lowest skew for degree centralisation. However, when we restrict our analysis to the chosen centrality measures, there was a good correlation between skewed centralisation and the impact on delayed percolation. Our work confirms previous research on networks that hub nodes are influential in spreading activities on networks, but also highlights the greater importance of other centrality measures on hyperbolic networks of varying edge density. It may be that some other measure, or a combination of properties, is at the heart of the influence and that it might be possible to fine tune influential node selection by observing a variety of graph properties.

7.6 Conclusion

This work has demonstrated that identifying influential nodes and targeting them for immunity has an inhibitory effect on the bootstrap percolation process on hyperbolic networks, when compared with random immunity.

This indicates that in these graphs certain nodes are highly influential in the network and warrant being protected. Given a small scale attack in a network, with 20 active seeds, our results suggest that under these conditions it is possible to immunise influential nodes and effectively grant herd immunity to the whole network. In the case of the graphs at activation threshold 10, by targeting 25 nodes with high centrality measures, we have reduced the likelihood of percolation from 1 to 0.1. On graphs at edge density 0.2, the probability of percolation for activation threshold 4 has decreased from 0.4 to 0.02, for activation threshold 3, the probability of percolation has decreased from 0.97 to 0.86. On this latter set of graphs, the rate of decline was linear, which suggests that increasing the number of immunised nodes would increase the obstructive effect on percolation.

Results from our simulations have demonstrated that targeting immunity at the
top ranked nodes for high degree, closeness and betweenness centrality and for low local node clustering coefficients has a delaying impact on the dynamics of the bootstrap percolation process on the set of hyperbolic graphs, when compared with random immunity which had no effect on the bootstrap process. This impact varies with the choice of node property selected. Node degree is commonly seen as an important indicator of influence in a network and our results confirm this, however in our graphs at the percolation threshold for activation threshold 2, closeness centralisation has the most effect in impeding the spread of the activity. In terms of global graph properties, the selected centrality measures had a good association with the level of skew of their respective graph centralisation measures, with the top ranked nodes of highest skew relating to a greater impact on delaying percolation.
Chapter 8

Conclusion

The aim of this thesis was to investigate spreading processes on models of complex networks. The graph model selected was the hyperbolic random geometric graph, and the processes examined on these graphs were Bootstrap Percolation and the evolutionary form of the Prisoner’s Dilemma. Our main focus was in observing the impact of spatial features on the spreading behaviour. Our hypotheses were that the spatial configuration of the network would impact on the outcomes for the processes. We examined these processes on a set of hyperbolic graphs of increasing edge density from 0 to 1. Our general observation is that the spatial effects undoubtedly had an impact on process outcomes, and that it is possible to identify distinct graph properties associated with these differences in outcome.

- **Transition Threshold** For both processes, as we increase edge density in the set of hyperbolic graphs, there is a distinct transition threshold above which the activity completely spreads, and below which the activity fails to spread. The transition threshold for the game of Prisoner’s Dilemma occurs at a higher edge density than for bootstrap percolation (for all of our activation thresholds from 2 to 10). This difference is not unexpected as the rules of interaction, and the update mechanisms at each time step, are different for each process.

- **Graph Properties at Transition Threshold** As edge density increases in the set of hyperbolic graphs, there is a distinct pattern of global graph proper-
ties. This pattern of properties can be correlated with the transition thresholds to help understand or identify features of the graphs which are significant in facilitating or obstructing the spread of an activity.

- **Binary Outcomes** Despite the differences in activation mechanisms, both processes display binary outcomes on the same set of hyperbolic graphs. In the Prisoner’s Dilemma, this is either the complete adoption of defection or the robustness of cooperation. In bootstrap percolation, the activity either completely percolates or fails to percolate. Outcomes for both processes display a clear transition threshold between these zones, and within reasonably close edge densities.

### 8.1 The Prisoner’s Dilemma

Our simulations on the set of hyperbolic graphs showed a clear transition threshold as edge density increased. Below this threshold, cooperation is robust and after it there is no resistance to the spread of defection. This threshold occurs at an edge density between 0.015 and 0.035, which is relatively sparse. In graphs of greater edge density, the adoption of defection was rapid and complete. Our analysis of graph properties at the transition threshold shows that the spatial impact of node degree on outcomes is an important feature of the spreading behaviour. In the majority of graphs, the presence of hub nodes facilitates the spread of defection. This contrasts with research on small world and scale free networks, reporting that the power law degree distribution and hub nodes help maintain cooperation [93, 113, 132, 102, 103, 110, 135]. An unexpected outcome was that closeness centrality also appears to be an important factor. At the transition threshold, the graphs were highly clustered with the value for transitivity decreasing. Significantly, the skew in closeness centralisation was increasing, reflecting the fact that at the threshold the graphs were becoming increasingly clustered. Typically node degree is seen as an important factor in spreading behaviours, and our results confirm this, yet in the threshold graphs, nodes with high closeness centrality had a particular impact in facilitating the spread of defection.
8.2 Bootstrap Percolation

Our simulations of bootstrap percolation on the set of hyperbolic graphs demonstrated a distinct transition threshold for percolation as edge density increased. Additionally, as the activation threshold increased, the percolation threshold was delayed to graphs of higher edge density. This reflected the fact that with a raised tipping point, inactive nodes required more edges to ensure sufficient access to activated nodes. The outcomes on individual graphs showed either complete percolation or failure to percolate, there were no examples of mixed outcomes in graphs which were at least minimally connected. This result is consistent with the mathematical investigation by Candellero et al. [24], which demonstrated that the activity either completely percolated or failed to percolate, with high probability. Our analysis of the changing pattern of graph centralisation measures at the threshold indicate that complete percolation occurs at the point where degree and closeness coincide and become increasingly skewed, whereas the betweenness centralisation shows low skew.

8.3 Bootstrap Percolation with Recovery

In the modified form of bootstrap percolation, where we introduced a mechanism for activated nodes to recover at each time step, there was some delay in percolation associated with random targeting of recovery nodes, when compared with standard bootstrap percolation, although this was marginal. However, targeting active nodes of high degree had an enhanced delaying effect on percolation, confirming the importance of node degree as an important feature in the spread of an activity, shown in research such as [53, 49, 125, 138, 137].

8.4 Bootstrap Percolation with Immunity

This set of simulations has demonstrated that identifying influential nodes and targeting them for immunity has an inhibitory effect on the bootstrap percolation process on hyperbolic networks, when compared with random immunity. Targeting nodes of
high degree had a marked inhibitory effect on percolation on graphs at both edges of the percolation threshold. However, in graphs at the lower edge of the percolation threshold, the selection of nodes with high closeness centrality had most impact. When observing global graph centralisation measures, there was a clear association between the degree of skew in these measures and the inhibitory impact of selecting immune nodes with these centrality measures. This contrasts in part with research by Pastor-Satorras and Vespignani [98] which introduced immunised nodes into a SIS (Susceptible - Infected - Susceptible) model on scale free networks. They found that random immunity had little effect, which confirms our own results, however, their research suggested that nodes of high degree should be targeted to ensure a successful outcome for immunisation strategies, whereas our own research considered other centrality measures and found that for some graphs, immunising nodes of high closeness centralisation was more effective.

8.5 Contribution

- **Graph Data Set**

  We have created a synthetic data set of 2,400 hyperbolic random geometric graphs with increasing edge density from 0 to 1, and have undertaken an analysis of graph properties in each graph. This data set facilitates our observations of the effect of graph topology on diffusion processes, as we can directly compare different activities on the same set of graphs. Additionally, we have observed graph properties for each of the graphs and can correlate these properties with different outcomes.

- **Simulation of Diffusion Processes on Hyperbolic Networks**

  We have used agent based simulation of the Prisoner’s Dilemma on the set of hyperbolic graphs and compared outcomes with graph properties. We have performed standard bootstrap percolation on the same set of graphs. Our analysis of global graph properties at the transition threshold in both sets of simulations
allows us some insight into features which facilitate or obstruct either spreading process. Our simulations of bootstrap percolation complement the research in the area, which has mainly concentrated on mathematical analysis and probabilistic methods. These mathematical analyses use a mean field approach, which simplifies a model by replacing individual interactions with an average or effective interaction. In agent based simulation of diffusion processes, agents attached to complex networks, such as hyperbolic graphs, are heterogeneous. The spread of an activity is dependent on the state of the agent, and of the state of agents within the neighbourhood of that node. This means that we can allow more individual complexity for each agent and observe the effect that local topology has on emergent behaviour in the global population.

- **Introduced modified form of Bootstrap Percolation**

  We have developed a recovery mechanism as a modification of the standard bootstrap percolation model. This introduces a stochastic element into an otherwise deterministic process. When modelling social contagion, this might reflect a change of mind scenario. Our results demonstrate that this modification has a delaying effect on the adoption of an activity. This effect is enhanced when activated hub nodes are specifically targeted for recovery.

- **Introduced Immunity to the Percolation Process**

  In our last simulations, we chose to focus on inhibiting percolation, by introducing the idea of immunity. This was modelled by a priori selection of nodes with various properties to be immune to the percolation process. These measures amount to control strategies, which seek to alter the outcomes in the process. In real life, this might relate to identifying influential nodes in a network to stop the spread of a contagious disease, or to prevent social contagion such as rumour propagation, or the spread of fake news.

- **Identifying Influential Nodes**

  Using results from the simulations of bootstrap percolation with recovery and
from bootstrap percolation with immunity, we have gained insights into the
identification of nodes which have significant impact on diffusion processes on
these networks. Our previous analysis of global and local node properties in
the set of graphs allows us to associate simulation outcomes with particular
patterns of properties. Node degree is commonly regarded as the standard
measure of influence in a network, our results show that this varies according
to the structure of the network, and to some extent we can a priori identify
node properties which are influential in different graphs, based on the measure
of skewed distribution of the property in that specific graph.

8.6 Application to Pandemic Scenarios

Following completion of the work in this thesis, real world events have suggested
potential applications which demonstrate the versatility of different aspects of this
research. The model might be applied in the following ways:

Agent based simulation on the set of hyperbolic graphs can be applied to mod-
elling the spread of a pandemic. The nodes represent members of a population and
the edges are the contacts between people. Average degree represents the average
number of contacts per person. The model can be used to find the critical point
at which an epidemic spreads within a population, based on increasing edge density.
Once this critical threshold is determined, we can further model the effect of introduc-
ing control strategies aimed at reducing the average number of contacts per person
to a point where the activity fails to spread or becomes endemic at a rate which is
manageable in terms of resources. Closely interconnected communities, such as nurs-
ing homes, residential institutions and certain workplaces may also be modelled by
community structure in the set of graphs, where it may then be possible to determine
the influential nodes in those settings and target them for control measures.

In the real world, this might be done by introducing social distancing. For ex-
ample, if social distancing is set at 2 metres then the average number of contacts of
each infected person will be less than if the distance is set at 1 metre or 1.5 metres.
Random geometric graphs are distance graphs, where pairs of nodes are connected if they lie within a specified distance of each other. The graph set can therefore be used to determine the likely outcome of the disease in the same population with different levels of contacts (modelled by graphs with differing edge density). Euclidean random geometric graphs, in particular, are ideal for modelling the effects of social distancing in various social settings such as schools or on public transport. Based on passenger capacity at different distancing measures, you can effectively model the average number of contacts each bus passenger has for significant periods during a bus ride.

Our final set of simulations introduced immune nodes into the graph, targeted at the most influential nodes based on high rankings for different centrality measures. In real life these nodes might represent people with high degree centrality, having a high number of contacts. This might represent healthcare providers who come into contact with a lot of infected people, or people with high closeness, such as a hospital administrator, or the family or other close contacts of healthcare providers. High betweenness centrality might represent an agency worker who moves between close-knit communities, such as different nursing homes.

The provision of immunity might represent the application of control strategies such as barrier techniques, aimed at physically preventing the spread of the contagious disease. This might include the use of personal protective equipment, the application of adequate hygiene measures, ensuring adequate education on disease transmission, and adequate training in the use of barrier methods.

Equally, the immunity experiments can be applied to attempts to control the spread of misinformation in times of public health emergencies such as a pandemic. It is possible that false or misleading information can cause significant negative consequences, in terms of ill-informed or reckless behaviour in social settings. Rumours that the disease is only contagious with the onset of symptoms might encourage close contacts of an infected person to believe that they are safe to go in to work unless they develop symptoms. Similarly, false or misleading information about preventative measures might abound, for example, preventative medicines which might have
harmful side effects, or even dangerous practices, such as gargling with bleach. Incorrect information can also lead to overconfidence that people are immune to the spread and might lead to risky behaviour for the individual and all their potential contacts. Trusted nodes in a population can be updated with correct information and can be relied on to resist the spread of the fake news.

Interestingly, in a pandemic situation, there are potential worldwide shortages of vital equipment and treatments which highlights a global competition for scarce resources, and may be modelled using game theory. For example, in recent times powerful governments have started buying up world supplies of personal protective equipment and ventilators. Equally, richer countries might have first access to life saving medication or vaccines, which can be very expensive. Additionally, the impact of personal behaviour during a pandemic may also be modelled by game theory, where behaviour based on personal advantage may be detrimental to the common good. Examples of this include failure to respect social distancing, such as mass gatherings for social events, or the failure to use face masks in situations where social distancing is difficult such as in supermarkets or on public transport. Examples such as these might have high relative utility for an individual but the risk of an outbreak is far more damaging to society as a whole.

Although our work has been simulated on hyperbolic random geometric graphs, our section on immunity can have wider applications, using other random graph models or models of real world networks. In Section 7.3, we developed a framework for identifying influential nodes to inhibit percolation on the hyperbolic networks.

Our approach simulated bootstrap percolation on the set of graphs of increasing edge density to determine the critical threshold for percolation. At this threshold we distinguished certain graph properties used as the basis for identifying influential nodes in the percolation process. We targeted immunity at the top ranked nodes with these properties and noticed that there was some measure of association with these targeted nodes and the inhibitory effect on percolation. This approach could also be applied to different networks to determine nodes with various properties as suitable candidates for inhibiting percolation. Nodes of high degree are often regarded
as highly influential in spreading processes, as super spreaders. Our own research confirms this, but also suggests that observing skewed distributions of other centrality measures might also be significant.

8.7 Future Work

It would be interesting to see if our findings hold for hyperbolic graphs of different sizes, and to investigate spreading phenomena on different graph models, to assess if our intuitions for the correlation between skewed distribution and inhibitory impact hold true for graphs with different global graph properties. It might also be possible to determine the degree of skew necessary to confidently predict an increased likelihood of delayed percolation.

Another interesting avenue of follow-up research is to replicate our work on all of the hyperbolic graphs within the percolation threshold zone, with a view to more detailed mapping of the relationship with edge density, the changing pattern of graph centralisation measures and outcomes in the percolation process. The threshold graphs are ideal subjects for investigation, as it is here that any potential alteration in outcomes can be readily observed. Of particular interest are those simulations which displayed significant changes in outcomes, following the introduction of inhibitory nodes. It would be useful to examine the relationship between the active seed set and the immunised nodes in greater detail. In our simulations at the percolation threshold for activation threshold 10, $R = 5.7$, only 10% of simulations completely percolated after the introduction of immunity. By investigating the spatial configuration of the paths between these sets of nodes, and the local neighbourhood structure, it might be possible to determine which local neighbourhood features have the greatest impact on global dynamics in the network.
Bibliography


[92] Colm O’Riordan and Humphrey Sorensen. Stable cooperation in the n-player prisoner’s dilemma: The importance of community structure. In *Adaptive Agents and Multi-Agent Systems III: Adaptation and Multi-Agent Learning: Adaptation and Multi-Agent Learning*, 5th, 6th, and 7th European Symposium,


162


