

Complex Contagions with Lazy Adoption



Se-Wook Oh
Christ Church College
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy

Trinity 2017

Acknowledgement

This entire thesis would not be existent without the thorough supervision of my direct academic ancestor, Professor Mason Alexander Porter. He has been one of the most influential figure in my academic life in terms of how to do science, and what it is to do science; although I cannot claim thus now I know how to do science, much less know what it is to do science. He has laid the foundation for my life as a junior researcher, although still there are countless weaknesses that I need to overcome to be a proper scientist, just like when I first started this journey for the Doctor of Philosophy (DPhil). This thesis, in that regard, is a compilation of all the struggles (and a few achievements) I have had in the journey up to this point that is yet to finish.

This journey could have been so much tougher without my respected fellow comrades (i.e., my officemates), Vladimir Murevics, the man I have spent the most time with during my life in Oxford, and Alejandro Javier Martinez, the man I have wasted the most time with during my life in Oxford. Their excellence in mathematics has been great help during the course of my study, and sharing office (and our communal house) with them has been great comfort throughout the course of my DPhil. Special thanks to Sang-Hoon Lee, whom I first met as a postdoc in our research group. As a senior researcher, he has been a great mentor with whom I could discuss both research-related things as well as things other than research. Even after he left to Korea, he has always been open to discussion whenever I requested, and he also kindly organised several seminars for me to present my work in Korea everytime I asked. He has made enormous contribution to my growth as a researcher, and I am short of words to thank him enough.

I have benefited greatly from my friends at Oxford across the disciplines through thought-provoking discussions and inspiring conversations. Also, if it had not been for my friends, I would not have had enough courage to change my discipline in my

first year in DPhil and pursue my real research interest, and would have stopped the journey of DPhil before I really get into it. Huge thanks to In-Ji Yeom and Min-Hyun Park for great advice and courage, and my sincere gratitude to Alasdair Campbell and Nicholas Letchford for their considerable help when changing my major.

There are many people whom I owe much to complete this work, who spent their precious time to read and criticize my work: Florian Klimm, Jacob Fisher, Peter Grindrod, Mikko Kivelä, James Moody, Feilx Reed-Tsochas, and Chinami Oka, who also gave me huge mental and emotional support, and has always been a comfort. Also, the work on real-data analysis in Chapter 5 could not have been done without the valuable data that I received from Thomas William Valente, George Gerald Vega Yon, and Javier Borge-Holthoefer.

For submitting the final version of my thesis after correction, I must thank Professor Philip Maini from the University of Oxford and Doctor Naoki Masuda from the University of Bristol, for their careful inspection of my thesis as well as for their thorough examination of my research work as the examiners of my viva. The three hours and a half of the viva was full of their helpful advices and sharp comments, while maintaining a relaxed atmosphere, which could have been more stifling and tense without their kind consideration.

I sincerely thank Kwanjeong Educational Foundation for their generous financial support for my study at Oxford. I would not have been able to start my life at Oxford without the grant awarded to me. I also thank Christ Church of Oxford and the members of The House for their constant care for students, which enabled me to finish my study at Oxford.

Last but not least, I would like to thank my parents and my family who have always been my biggest fans, and to whom I have always given the same answer whenever they asked when I will finish my DPhil. I am happy that this thesis will finally put an end to my DPhil, and get me prepared for the next adventure.

Contents

1	Introduction	1
1.1	Mathematics and social networks	1
1.1.1	Networks and graph theory	2
1.1.2	Dynamical processes on networks	3
1.1.2.1	Threshold model	3
1.2	Organization of this thesis	5
2	Background	6
2.1	Graph theory	6
2.1.1	Graph	6
2.1.2	Degree	8
2.1.3	Path, cycle, and connected component	9
2.1.4	Trees	10
2.1.5	Clustering coefficient	10
2.1.6	Centrality measures	11
2.1.6.1	Degree centrality	11
2.1.6.2	Geodesic node betweenness centrality	12
2.1.6.3	Eigenvector centrality	12
2.2	Random-graph models	12
2.2.1	Erdős–Rényi graphs	12
2.2.2	Configuration model	14
2.2.2.1	Configuration model with stub matching process	15
2.2.2.2	Configuration model with cliques	16
2.3	Social influence models and Watts threshold model	17
2.3.1	Numerical simulations of the WTM	20
2.3.2	Analytics of Watts threshold model	23
2.3.2.1	Probability generating functions	24

2.3.2.2	Application of probability generating functions to site percolation	25
2.3.2.3	Analytical prediction of a global cascade	26
2.4	Summary	29
3	Incorporating timers into the WTM	30
3.1	Timer models	30
3.1.1	WTM with timers	31
3.1.2	Adoption paths	32
3.2	WTM with timers on small networks	34
3.3	Timers on large random networks	39
3.3.1	Timer distribution and dynamics of the WTM with a timer .	39
3.3.2	Adoption paths in large networks	42
3.3.2.1	Stems and branches	43
3.3.2.2	Dissemination trees for synthetic networks	45
3.3.3	Adoption paths in networks with threshold $\phi = 0.16$	60
3.4	Summary	62
4	Pair-approximation analysis of timer model	63
4.1	Zero-temperature random-field Ising model on Bethe lattices	63
4.2	Pair approximation of the WTM	66
4.3	Pair approximation of the WTM with timers	68
4.3.1	Cascade condition for the WTM with a timer	72
4.4	Summary	73
5	WTM with timers on real networks	75
5.1	Adoption process for the WTM with timers on real-world networks .	75
5.2	Dissemination trees for real-world networks	81
5.3	Real-world data analysis	87
5.3.1	Three small real-world networks and their dissemination trees	88
5.3.2	The distribution of thresholds and timers in Twitter network .	94
5.4	Summary	98
6	Summary and Conclusion	99
A	Time to steady state for networks with infinitely large loops	103
B	Stems with timers larger than the mean	106

List of Figures

1.1	Example of a graph, where the number of nodes is $N = 10$. The green dots indicate nodes, and the black lines indicate edges.	2
2.1	A graph with 5 nodes and 5 edges.	7
2.2	A tree with 10 nodes.	10
2.3	An example to illustrate the way a node becomes a part of cliques or becomes adjacent to individual nodes in a configuration model with cliques. The thick lines indicate the internal stubs, and the thin lines indicate the external stubs. (a) The node has degree 7, and 3 of its stubs are chosen with probability $\gamma(7,4)$ and connect the node to a 4-clique, 2 stubs are chosen with probability $\gamma(7,3)$ to connect the node to a 3-clique, and the remaining 2 stubs connect the node to any individual nodes that are not part of a clique. (b) Portion of a network with node i connected to cliques (blue) and individual nodes (red). . .	16
2.4	ϕ - z phase diagram of WTM, where nodes have (a) homogeneous thresholds or (b) heterogeneous thresholds. Brightness indicates the final fraction ρ of activated nodes, and brighter colours indicate higher ρ values. For both panels (a) and (b), we run the WTM on ER networks of size (i.e., the number of nodes) $N = 1,000$. At time $t = 0$, a seed fraction $\rho_0 = 0.001$ of N nodes is chosen uniformly at random and their states become 1, and the rest of the nodes have state 1. We average the final fraction of activated nodes ρ over 100 realizations of ER networks for every value of (ϕ, z) . There are 50×50 number of coordinates in total, in which each axis has 50 coordinates. In panel (a), all nodes' thresholds are uniform, so $\phi_i = \langle \phi \rangle$ for all $i \in V$. In panel (b), we draw the thresholds from a Gaussian distribution with mean $\langle \phi \rangle$ and standard deviation $\sigma = 0.2$. All nodes are updated synchronously at every time step.	21

2.5	Horizontal and vertical slices of Figs. 2.4a and 2.4b. Panels (a) and (b), respectively, are horizontal slices of ρ versus ϕ with homogeneous and heterogeneous thresholds. The mean degrees are $z = 3$ (blue circle), 4.5 (green triangle), and 5.0 (red square). Panels (c) and (d), respectively, are vertical slices ρ versus z with homogeneous and heterogeneous thresholds. The mean thresholds are $\phi = 0.1$ (blue circle), 0.15 (green triangle), and 0.2 (red square).	22
2.6	ϕ - z phase diagram of WTM as in Fig.2.4 and analytical cascade boundary (red curve). All node thresholds are (a) equal to 0.2 or (b) drawn from a Gaussian distribution with mean threshold $\langle\phi\rangle = 0.2$ and standard deviation $\sigma = 0.2$. The number N of nodes is 1000, and the fraction ρ_0 of seed nodes is 0.001, where we choose seed nodes uniformly at random. We averaged over 100 realizations of ER networks and obtain the final fraction ρ of activated nodes. There are 50×50 number of coordinates in total, in which each axis has 50 coordinates.	28
3.1	Illustration of different adoption paths in the WTM with a timer. Each node v_i is assigned a timer τ_{v_i} and a threshold ϕ_{v_i} , where $i \in \{1, \dots, 5\}$. We initiate node v_0 in the state 1, so we do not need to assign it a threshold or a timer. Suppose that all nodes v_i (with $i \in \{1, \dots, 5\}$) have the same threshold $\phi = 0.5$. An adoption of any node immediately triggers the timers of its neighbours.	33
3.2	Small examples to illustrate the effect of incorporating timers in the WTM. The dark brown node v_0 is the seed node with state $s = 1$ at $t = 0$, and the orange nodes are in state $s = 0$ at $t = 0$. The thresholds are homogeneous, with $\phi = 0.1$ for each node.	35
3.3	Example of adoption paths when there is a node with a large adoption threshold.	38

3.4	(a) Adoption curves of the WTM with timers. The pink curve is without timers (i.e., the original WTM), the green curve is when the timers are homogeneous with $\tau = 4$, the orange curve is with heterogeneous timers τ that are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and the blue curve is for heterogeneous timers given by integers that we round down from a random variable that follows the Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$. Squares are the results of numerical simulations, the dashed lines mark the times at which the adoption process of the corresponding colour reaches a steady state, and the solid curves are results from an analytical approximation (see Chapter 4). (b,c) Change of the times to steady state (T_{unif} and T_{Gam} , respectively) by increasing the standard deviation σ when timers are distributed (b) uniformly at random and (c) approximately according to a Gamma distribution. We obtain our numerical results by averaging over 1,000 realizations of the WTM with timers on $G(N, p)$ ER networks with $N = 10,000$ nodes and edge probability $p = 0.0006$. To isolate the effects of incorporating timers, we use the same 1,000 ER networks for each of the 4 different cases. .	40
3.5	(a) An example network on which we run the WTM with timers. Curved arrows illustrate the spread of adoptions, and each curved arrow represents an adoption path. The brown curved arrow is a stem, and the green curved arrows are branches. (b) Graphical illustration of a dissemination tree, based on the spread of adoptions in panel (a). Node v_0 is a seed, which is the root of the stem (v_1, v_2, v_3, v_4) . Nodes v_0 , v_1 , and v_2 in the stem initiate the adoption in branches; they are the roots of the branches (v_5, v_6, v_7) , (v_8, v_7) , (v_8, v_9) , and (v_{10})	43
3.6	Illustration of a cycle in a dissemination tree.	44

3.7	Graphical representation of a dissemination tree for the WTM with (a) homogeneous timers and (b) heterogeneous timers. The edges are directed from higher levels [lv. 5 in (a) and lv. 8 in (b)] to lower levels [lv. 1 in both (a) and (b)]. We colour edges in blue if they are part of a stem and in gray if they are part of a branch. The solid blue lines indicate edges that are part of stems, and the solid gray lines indicate edges that are part of branches. We colour the nodes according to their time values, which range from 0 (white) to 8 (black). We use stars for nodes in stems and disks for nodes in branches. The number of levels in the dissemination tree with homogeneous timers is 5, and the number of levels with heterogeneous timers is 8.	45
3.8	Adoption curves of the WTM with timers and their adoption curves after we change the timers of stem nodes of the dissemination trees. The green curve is for the WTM with homogeneous timers with $\tau = 4$, the orange curves are for the WTM with heterogeneous timers τ that are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and the blue curves are for heterogeneous timers determined from the Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and then rounding down to an integer. To isolate the effects of different distributions of timers, in each case, we run the WTM using the same networks with the same seed nodes. The dark orange and blue curves are before we change the timers in dissemination trees, and the corresponding light-coloured curves are after we change those timer values (see the main text). The squares (for the dark curves) and disks (for the light curves) are the results of numerical simulations, and the dashed lines mark the times at which the adoption process of the corresponding colour reaches a steady state. We obtain our numerical results by averaging over 1,000 realizations of the WTM with timers on configuration-model networks with $N = 10,000$ nodes and a Poisson degree distribution with mean $z = 6$. For each realization, we generate an independent network. We also determine the seed node (uniformly at random) and timer values separately for each realization.	55

3.9	Adoption curves of the WTM with timers on generalised configuration-model networks we used in Table 3.3 and 3.4, and adoption curves after we adjust the timers of stem nodes of dissemination trees. We use (a,b) generalised configuration-model networks with 3-cliques and (c,d) generalised configuration-model networks with 4-cliques. We consider different values of the edge-clique ratios α and β ; we use the parameter values $\alpha = 0.5$ in panel (a), $\alpha = 1$ in panel (b), $\beta = 0.5$ in panel (c), and $\beta = 1$ in panel (d). For all networks, each node has a homogeneous threshold value of $\phi = 0.1$. The green curve is for the WTM with homogeneous timers with $\tau = 4$, the orange curves are for the WTM with heterogeneous timers τ that are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and the blue curves are for heterogeneous timers determined from the Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and then rounding down to an integer. To isolate the effects of different distributions of timers, in each case, we run the WTM using the same networks with the same seed nodes. The dark orange and blue curves are before we change the timers in dissemination trees, and the corresponding light-coloured curves are after we change those timer values (see Section 3.3.2.2). The squares (for the dark curves) and disks (for the light curves) are the results of numerical simulations, and the dashed lines mark the times at which the adoption process of the corresponding colour reaches a steady state. Our numerical results are means over 1,000 realizations. For each realization, we generate an independent network. We also determine the seed node (uniformly at random) and timer values separately for each realization.	58
4.1	An example of (a) a Bethe lattice with $w = 3$, and (b) a 3-Cayley tree. Nodes in the boundary of the Cayley tree are coloured in black. . . .	64
4.2	An illustration of level-by-level spreading of adoption behaviour in a network.	67
4.3	(Continue on the following page.)	70

4.3	Adoption curves of the WTM with timers on configuration-model networks with $N = 10,000$ nodes and a Poisson degree distribution with mean $z = 6$. We use (a–c) standard configuration-model networks, (d–i) generalised configuration-model networks with 3-cliques and (j–o) generalised configuration-model networks with 4-cliques. We consider different values of the edge–clique ratios α and β , where α determines the edge–clique ratio for a node of degree $k \geq 3$ in Congen-3 networks and β determines the edge–clique ratio for a node of degree $k \geq 4$ in Congen-4 networks. We use the parameter values $\alpha = 0.5$ in panels (d–f), $\alpha = 1$ in panels (g–i), $\beta = 0.5$ in panels (j–l), and $\beta = 1$ in panels (m–o). For all networks, each node has a homogeneous threshold value of $\phi = 0.1$. The green curve is for the WTM with homogeneous timers with $\tau = 4$, the orange curves are for the WTM with heterogeneous timers τ that are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and the blue curves are for heterogeneous timers determined from the Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and then rounding down to an integer. To isolate the effects of different distributions of timers, in each case, we run the WTM using the same networks with the same seed nodes. Dashed lines mark the times at which the adoption process of the corresponding colour reaches a steady state. Our numerical results are means over 1,000 realizations. For each realization, we generate an independent network. We also determine the seed node (uniformly at random) and timer values separately for each realization.	71
5.1	(Continue on the following page.)	77

- 5.1 Adoption curves of the WTM with timers on the nine FACEBOOK100 networks [130, 131]. The green crosses are for the WTM with homogeneous timers with $\tau = 4$, the orange squares are for the WTM with timers that are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and the blue diamonds are for timers determined from the Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and then rounding down to an integer. The dashed lines mark the times at which the adoption process of the corresponding colour reaches a steady state, and the solid curves are results from an analytical approximation (see Section 4.3). To isolate the effects of different distributions of timers, in each case, we run the WTM using the same networks with the same seed nodes. Our numerical results are means over 1,000 realizations. We also determine the seed node (uniformly at random) and timer values separately for each realization. 78
- 5.2 Cumulative degree distributions of the nine FACEBOOK100 networks: from left to right, (a) Reed College, (b) Simmons College, (c) California Institute of Technology, (d) Haverford College, and (e) Swarthmore College. The vertical axes are plotted on a logarithmic scale. 80

5.3	Adoption curves of the WTM with timers on FACEBOOK100 networks and adoption curves after we adjust the timers of stem nodes of dissemination trees. The green curve is for the WTM with homogeneous timers with $\tau = 4$, the orange curves are for the WTM with heterogeneous timers τ that are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and the blue curves are for heterogeneous timers determined from the Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and then rounding down to an integer. To isolate the effects of different distributions of timers, in each case, we run the WTM using the same networks with the same seed nodes. The dark orange and blue curves are before we change the timers in dissemination trees, and the corresponding light-coloured curves are after we change those timer values (see the main manuscript). The squares (for the dark curves) and disks (for the light curves) are the results of numerical simulations, and the dashed lines mark the times at which the adoption process of the corresponding colour reaches a steady state. Our numerical results are means over 1,000 realizations. For each realization, we generate an independent network. We also determine the seed node (uniformly at random) and timer values separately for each realization.	86
5.4	Cumulative distributions of threshold (red) and timer (blue) of (a,b) the physicians network, (c,d) the Brazilian farmers network, and (e,f) Korean women network. We calculate thresholds as the adopted fraction of out-neighbours, and we calculate timers as the time difference between a node and its most recently adopted out-neighbours. Data on the networks and the times of adoptions of nodes are from an R package, called NETDIFFUSER, developed by Vega Yon et al [136]. . .	91
5.5	Cumulative distribution of threshold (red) and timer distribution (blue) of the 15M network. We use 0.01 as the bin width of the threshold distribution, and we use 1 day (i.e., 86,400 seconds) as the bin width of the timer distribution. We calculate thresholds as the adopted fraction of out-neighbours, and we calculate timers as the time difference between a node and its most recently adopted out-neighbours minus 1.	95

5.6	Scatter plot of thresholds and timers of nodes in the 15M network. Each green data point corresponds to each node in the network, and each data point is positioned along the vertical axis by its threshold, and along the horizontal axis by its timers. We calculate thresholds as the adopted fraction of out-neighbours, and we calculate timers as the time difference (in the units of second) between a node and its most recently adopted out-neighbours minus 1.	96
5.7	Change of the cumulative distributions of (a) thresholds, and (b) timers for the 15M network. We assume that a node decides to adopt when the x^{th} from the last out-neighbour to adopt adopts, where $x \in \{0, 1, 2, \dots\}$, and we calculate thresholds and timers according to (5.3) and (5.4) We increase x from 0 to 10 and then to 100, and we investigate the ensuing changes in the distributions of thresholds and timers. We determine the time that a node as the bin width of threshold distribution is 0.01, and the bin width of timer distribution is 1 day, which corresponds to 86,400 seconds.	98
A.1	Examples of networks in which a seed node v_0 is adjacent to (a) an infinitely large odd-sized loop with nodes $v_i \in \{v_1, v_2, \dots, v_{2N+1}\}$ and (b) an infinitely large even-sized loop with nodes $v_j \in \{v_1, v_2, \dots, v_{2N+2}\}$. We use X to denote a path $(v_1, v_2, \dots, v_{N+1})$, and we use Y to denote $(v_1, v_{N+2}, \dots, v_{2N+1})$	104
B.1	Examples of networks with larger mean stem-node timers than mean branch-node timers.	107

Chapter 1

Introduction

Over the decades, and especially recently amidst the surge in data availability and richness, it has become increasingly popular to take quantitative approaches to the study of sociological questions [17, 70, 82, 91, 138]. Modelling efforts have drawn from computational, statistical, and mathematical approaches [114]; and the study of mechanistic models that incorporate data in a meaningful way (see [53] for an example of data-driven mechanistic modelling) can give insight into both existing observations and forecasting of future dynamics. Through this thesis, we introduce a *timer model* and study its behaviour on networks both numerically and analytically, and we also examine its application on real-world network data to scrutinize the possible use of the model for understanding societal behaviour of people reflected on the real-world data.

1.1 Mathematics and social networks

There are mainly two paths for research in networks: study on the network structures, and study on the dynamical processes on networks.¹ Thanks to its firm basis on mathematical axioms, mathematical language is often employed for modelling in the study of networks, because it can interpret things that may seem ambiguous in ordinary languages. For this reason, in the field of social networks, mathematical models are exploited to understand the social network structures and the social phenomena in simpler but more precise manner. If fortunate enough, such models can investigate and interpret any “hidden order” that is present by fitting observed statistical

¹Adaptive network models incorporate the network structures and dynamical processes, because the evolution of a network structure is fundamentally entangled with the dynamical processes on the network [63]. An adaptive network is a “dynamic network” which is a network whose structure varies over time [23].

regularities via data analysis and then reproducing such regularities with long-time or equilibrium dynamics from some generative model, which might even be used to predict similar happenings in the future [53].

1.1.1 Networks and graph theory

To model the structure of social networks, researchers have used ideas from *graph theory* [105]. Since its introduction in 1736 by Euler [13], *graph theory* has been actively studied by many mathematicians. An ordinary graph consists of two primary elements: nodes (i.e., vertices) and edges (i.e., links). See Fig. 1.1. Because of its rich mathematical tractability as well as intuitive visual representation, the study of graphs is widely adapted as a tool to understand properties of systems that are composed of entities and interactions between them, such as biological systems (e.g., neural networks [122], cellular networks [5], and genomic networks [37]), infrastructure (e.g., urban network [10], communication networks [111], transportation systems [65]), and of course, human social networks [2, 15, 62, 81, 124, 143, 146]. We will introduce the mathematical concepts of graph theory later in Section 2.1.

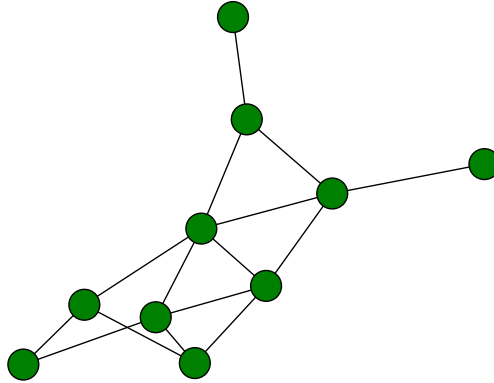


Figure 1.1: Example of a graph, where the number of nodes is $N = 10$. The green dots indicate nodes, and the black lines indicate edges.

1.1.2 Dynamical processes on networks

The field of social networks has used ideas from dynamical processes to model social phenomena [105]. For example, there have been numerous studies of the spread of opinions, actions, memes, information, misinformation, “alternative facts”, and other phenomena in populations in disciplines such as sociology, economics, computer science, physics, and many others [18, 29, 72, 113, 148]. By analogy with the spread of infectious diseases in a population, spreading phenomena — including the spread of defaults of banks, norms in populations, and products or new practices in populations — are often modelled as contagion processes on a network. To distinguish between different mechanisms in social and biological contagions, the former are often construed as examples of “complex contagions”, and the latter are often construed as examples of “simple contagions” [85, 100, 144].

The quantitative study of the spread of behaviour in social networks has a long history that dates back (at least) several decades [7, 42, 60, 125, 126, 134]. In recent years, studies of social influence have tended to focus on large social and/or communication networks, taking advantage of the increased availability of microblogging data sets (e.g., using data from Twitter) with relational information that enables one to incorporate network effects into models [58, 110]. In studying social influence, one often explores what conditions yield *cascades* [102], in which a small seed of activity leads to a large change in a network. In the study of spreading models, a common way to quantify cascades is to examine when an infinitesimally small seed fraction of adopted nodes generates a nonvanishing mean cascade size as the total number N of nodes in a network becomes infinite ($N \rightarrow \infty$) [48, 113]. In practical applications with empirical data, one often measures cascade sizes in other ways (such as by calculating how long it takes for a given fraction of the nodes in a network to adopt). See the discussion of cascade conditions in [113].

1.1.2.1 Threshold model

A particularly popular type of model for the spread of behaviour in social networks are so-called *threshold models*, in which nodes update their states if the amount of peer pressure from their neighbouring nodes (usually just nearest neighbours) exceeds some personal threshold. The simplest such model is the Watts threshold model (WTM) [137], which uses a linear updating rule that is very similar² to the one

²In [60], which does not incorporate network structure (it assumes that each node is adjacent to all other nodes), Granovetter did not define the adoption threshold of a node in terms of either the fraction or the total number of its adopted neighbours. Because of the all-to-all connectivity in [60],

introduced by Granovetter [60] and which was previously examined on networks by Valente [133, 134]. (It is also related to bootstrap percolation [28].) The WTM uses a “threshold” to represent the latent tendency of an individual to adopt an innovation (or become infected, if one wants to use more biological terminology) when at least some fraction of its neighbours has adopted the innovation [76]. Threshold models of adoption were first studied by sociologists, and the idea of a threshold comes from a sociological theory [60] that articulates that a person exhibits inertia in adopting an innovation as a way to reduce cost in making decisions. The WTM is appealing to study in part because of its mathematical tractability [50, 52, 113] and in part because it incorporates simple notions of peer pressure, social reinforcement (because multiple neighbours who have adopted an idea increase the peer pressure for a node to adopt), and personal resistance to peer pressure.³ It thereby provides a simple model of social influence, which occurs when an individual is influenced by others and adopts their behaviour [84, 118].

The WTM and its generalizations have been studied from many perspectives. This includes a considerable amount of work — both analytical and numerical — on WTM dynamics on random networks with various characteristics, including local clustering [66, 67, 96], community structure [48], degree–degree correlations [34], and communities with intercommunity correlations [98]. The WTM has also been generalized to dynamics on temporal networks [74] and multiplex networks [77, 83]. In the study of networks constructed from empirical data, the WTM has been used to examine phenomena such as protest recruitment [58] and adoption of technology [134]. There are also many variants of the WTM that caricature adoption behaviour in different ways; these variants include thresholds that rely on the total number of neighbours [26], a multi-stage threshold model [99], on–off thresholds [33], threshold models with memory [35, 55, 127], and a threshold model that incorporates node states (through “synergistic” effects) from nodes other than nearest neighbours [73].

There are many possible reasons why somebody may wait before adopting an idea, buying a product, etc. Possibilities include unawareness of an innovation, awareness

Granovetter’s linear-threshold update rule does not distinguish between the fraction (as in [137]) and number (as in [26]) of adopted neighbours. There is also a slight difference between Granovetter’s update rule and the one in the WTM: in the former, a node will not adopt under any circumstance if its threshold is 1; in the latter, a node with threshold 1 adopts if all of its neighbours have adopted. Finally, although threshold models are used to model social contagions (and for so-called “complex contagions”), Granovetter wrote in [60] that “In the broader context of threshold models, the idea of ‘contagion’ seems inappropriate, since much more is involved than mere imitation of the last person observed.”

³In some applications, one sometimes uses terms like “infected” or “active” rather than “adopted”.

but not yet deciding to adopt something, a “decision” to adopt something but laziness before changing behaviour [69], and so on. Different social-influence models have different adoption rules that codify behavioural latency (i.e., a delay before adopting a behaviour) in different ways, and different adoption rules generate different patterns of growth of the fraction of adopters over time. The study of different patterns in an adoption process is an important aspect of research on dynamical processes on networks [20, 58, 135]. For example, in the WTM, the time at which a node adopts an innovation depends both on the node’s threshold and on the adoption status of its neighbours (and hence on network architecture), and this affects the temporal evolution of the fraction of adopters in a network. In the present thesis, our goal is to incorporate response times into spreading models such as threshold models: even if a threshold is met or exceeded, there is often a delay until a behaviour is adopted. This can be due to personal inertia or to other factors. In this thesis, we introduce a *timer* to model the tendency of individuals to wait some amount of time between deciding to adopt a behaviour and actually adopting it, and we investigate how the incorporation of timers (especially ones that are heterogeneous in a population) changes qualitative dynamics of the WTM.

1.2 Organization of this thesis

This thesis is composed of 6 chapters, including this introductory chapter. We begin Chapter 2 by laying out the mathematical theories including a brief introduction to graph theory, the Erdős-Rényi random graph model, probability generating function, site-percolation, and Watts cascade model. In Chapter 3, we introduce what we deem a *timer model*, and illustrate the effects of incorporating a timer on small networks and large random-graph ensembles in the chapter. Chapter 4 is concerned with analysis of a timer model, for which we employ a *pair approximation* that is originally motivated by a random-field Ising model on Bethe lattice. Chapter 5 presents the application of the timer model for the analysis of real-world network data. We conclude in Chapter 6 and discuss the plan to further our research. We include further discussions and calculations in appendices.

Chapter 2

Background

In this chapter, we present some mathematical background that is necessary to understand our work. We begin this chapter with a brief introduction to graph theory, where we introduce the idea of a *tree*, which we will use extensively for analytical calculations in Chapter 4. We then introduce the main random network models that we use in this thesis (Erdős–Rényi networks and configuration model). We also introduce the main dynamical process (the Watts threshold model) that we will generalise. We follow the discussions in [105, 117] when introducing graph theory and Erdős–Rényi networks, and we follow the discussion in [137] when introducing the Watts threshold model.

2.1 Graph theory

2.1.1 Graph

A graph¹ G is an ordered pair (V, E) , where V indicates a set of nodes and E indicates a set of edges [105, 117]. The set V of nodes is a non-empty set, and E is a set of 2-element subsets of V , such that $E = \{\{i, j\}, i, j \in V\}$. We define the *size* of a graph as the number of nodes in the network (i.e., $|V|$).

If there exists an edge $e \in E$, nodes i and j are adjacent to each other through an edge, and we say that i and j are *adjacent* in the graph G , and the two nodes i and j are *neighbours*. We use $\Gamma(i)$ to indicate the set of neighbours of node i . Therefore, if node j is the neighbour of node i , the node $j \in \Gamma(i)$. Additionally, we say that e and

¹There are different kinds of graphs — including weighted graphs, directed graphs, and graphs with multi-edges or self-edges. However, unless otherwise specified, we will use the term *graph* as an unweighted, undirected, and *simple* graph, which means that it does not have any multi-edges or self-edges.

i , as well as e and j , are *incident* because the nodes i and j are adjacent through an edge e .

Adjacency and incidence are useful concepts to express the structure of graphs in mathematical language. An *adjacency matrix* A is a matrix that encapsulates information of nodes and edges using adjacency relations:

$$A_{ij} = \begin{cases} 1, & \text{if } \{i, j\} \in E, \\ 0, & \text{otherwise.} \end{cases}$$

An *incidence matrix* B is also a matrix with information of nodes and edges, but using incidence relations:

$$B_{ij} = \begin{cases} 1, & \text{if } i \text{ and } e_j \text{ are incident,} \\ 0, & \text{otherwise.} \end{cases}$$

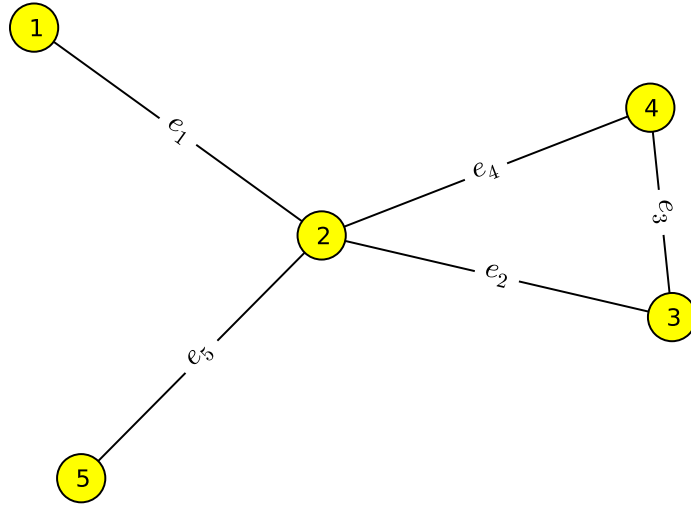


Figure 2.1: A graph with 5 nodes and 5 edges.

For an example graph, see Fig. 2.1. It shows a graph $G = (V, E)$ with node set $V = \{1, 2, 3, 4, 5\}$ and edge set $E = \{e_1, e_2, e_3, e_4, e_5\}$, where $e_1 = \{1, 2\}$, $e_2 = \{2, 3\}$, $e_3 = \{3, 4\}$, $e_4 = \{2, 4\}$, and $e_5 = \{2, 5\}$. The adjacency matrix is

$$A = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix} \end{matrix}, \quad (2.1)$$

and the incidence matrix is

$$B = \begin{matrix} & \begin{matrix} e_1 & e_2 & e_3 & e_4 & e_5 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{matrix}. \quad (2.2)$$

As one can see from equation (2.1), the adjacency matrix is symmetric because $A_{ij} = A_{ji}$ for all i and j ; an adjacency matrix of an undirected graph is always symmetric, while an adjacency matrix of a directed graph can be asymmetric. All elements are either 0 or 1, because we study simple unweighted graphs (which do not have multi-edges). Furthermore, the trace of A is 0 because the graph is simple and hence does not have self-edges. However, the trace of the incidence matrix (2.2) need not be 0, as diagonal elements are not necessarily 0 in this case. Additionally, an incidence matrix is not typically a square matrix, as the number of nodes and the number of edges in networks are usually different. As an edge connects two nodes, every column in the incidence matrix, which corresponds to different edges in the network, must contain exactly two ‘1’ entries.

2.1.2 Degree

In an undirected network, degree k_i of a node i is the number of adjacent neighbours of i . We can express it as

$$k_i = \sum_{j \in V} A_{ij} = \sum_{j \in V} A_{ji}. \quad (2.3)$$

A *degree sequence* $(k_1, k_2, \dots, k_{|V|})$ is a sequence of degrees of nodes in a network. With degree sequence, one can examine a distribution of degrees that is called a

degree distribution — a probability distribution of degrees of a network. Instead of degree distribution, *complementary cumulative degree distribution*² is often used to characterize a network by its structure. The complementary cumulative degree distribution, evaluated at k , is the probability that a node picked uniformly at random has degree larger than, or equal to, k .

Assume that we arrive at a neighbour along an edge of a node. If the neighbour we arrived at has k other edges than the edge we followed, the total degree of the neighbour is $k + 1$. We call the k other edges than the one we arrived along as *excess degree* k , and the distribution of excess degree (i.e., *excess degree distribution*) has a distribution

$$q_k = \frac{(k + 1)P_{k+1}}{\langle k \rangle}, \quad (2.4)$$

where P_{k+1} is degree distribution.

One can also define degrees in a directed network; out-degree k_i^{out} of a node i is the number of outward adjacent neighbours (i.e., neighbours that are adjacent via an out-going edge) of i , and in-degree k_i^{in} of a node i is the number of inward adjacent neighbours (i.e., neighbours that are adjacent via an in-going edge) of i . One can calculate the out-degree as $k_i^{\text{out}} = \sum_{j \in V} A_{ij}$, and one can calculate the in-degree as $k_i^{\text{in}} = \sum_{j \in V} A_{ji}$.

2.1.3 Path, cycle, and connected component

Note that the graph in Fig. 2.1 includes a *cycle*, which we define as a *path* whose last node is adjacent to the first node of the path. A path in a graph is a sequence $(v_1, v_2, \dots, v_n)$ of nodes, in which all nodes are distinct, and for $i \in \{1, \dots, n\}$, node v_i and v_{i+1} are adjacent to each other³. Therefore, the cycle in Fig. 2.1 is $(2, 3, 4)$ because node 4 is adjacent to node 2. Also, the graph in Fig. 2.1 is *connected* because there is a path between every pair of nodes. If a graph is composed of multiple components each of which is connected, then each component is called a *connected component* (or *cluster*; see vulnerable cluster in Section 2.3.1).

²Some authors mistakenly call a complementary cumulative degree distribution as a cumulative degree distribution. However, the cumulative distribution function of a random variable X , evaluated at x , means the probability that X has value less than or equal to x , which is not the same as the complementary cumulative distribution function.

³The length of a path $(v_1, v_2, \dots, v_n)$ is the number of edges in the path, that is, $n - 1$.

2.1.4 Trees

A tree is a graph in which any two nodes are connected by exactly one path. Therefore, a tree must be connected, and must not contain any cycle (i.e., must be *acyclic*). Figure 2.2 is an example of a tree with 10 nodes. We can designate node 0 as the root of the tree, and we can refer to such a tree as a *rooted tree*. In a rooted tree, we define a *leaf* as a node with degree 1. In Fig. 2.2, nodes 3, 4, 6, 8 and 9 are leaves, and they are coloured in green.

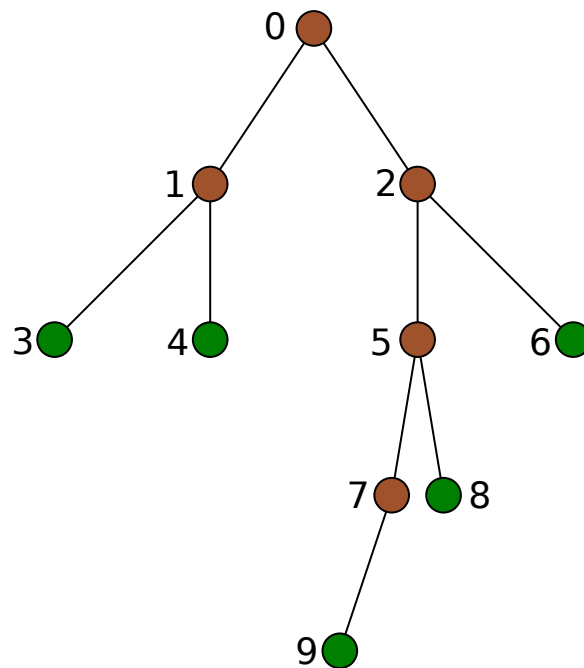


Figure 2.2: A tree with 10 nodes.

2.1.5 Clustering coefficient

Many networks are not trees, but they have clusterings. A clustering coefficient is a measurement to quantify the extent to which nodes in a network form densely connected groups [105]. There is more than one way to calculate the clustering coefficient of networks. In this thesis, we use the one defined by Watts and Strogatz

[141], which is widely used. The global clustering coefficient C is

$$C = \frac{\text{number of triangles}}{\text{number of connected triples}}. \quad (2.5)$$

In contrast to the global clustering coefficient, we define the *local clustering coefficient* C_i of a node i as the number of triangles to which the node is connected divided by the number of connected triples that include the node:

$$C_i = \frac{\text{number of triangles connected to node } i}{\text{number of connected triples that include node } i}. \quad (2.6)$$

We can then calculate the *degree-dependent clustering coefficient* C_k of nodes with degree k in a network as the mean of the local-clustering coefficients of nodes with degree k in a network:

$$C_k = \frac{\sum_{i \in V_k} C_i}{|V_k|}, \quad (2.7)$$

where V_k is a set of nodes with degree k .

2.1.6 Centrality measures

Understanding the structure of a network is important for studying a network system. One key research question is to find out which node in a network plays the most central role for a dynamical process on that network, such as a biological network [45, 79] or a social network [88]. Here, we introduce the notions of degree centrality, (geodesic node) betweenness centrality, and eigenvector centrality.

2.1.6.1 Degree centrality

Degree centrality measures how many connections a node has. The degree centrality, z_i , of a node i is simply calculated as the degree (i.e., the number of neighbours) of the node:

$$z_i = k_i, \quad (2.8)$$

where k_i is the degree of node i .

2.1.6.2 Geodesic node betweenness centrality

Geodesic node betweenness centrality, b_i , ranks a node i based on the number of geodesic paths — the paths with the shortest length — that pass through the node:

$$b_i = \sum_{j,l \in G} \frac{u_{j,l}(i)}{u_{j,l}}, \quad (2.9)$$

where nodes $i, j, l \in G$ are different nodes in a graph G , $u_{j,l}$ denotes the number of geodesic paths between nodes j and l , and $u_{j,l}(i)$ denotes the number of geodesics between nodes j and l that pass through node i .

2.1.6.3 Eigenvector centrality

The eigenvector centrality, g_i , of a node i determines not only how well the node i itself is connected, but also how well i is connected to well-connected nodes in a network. It is calculated using the largest eigenvalue κ of the adjacency matrix A of the network:

$$g_i = \frac{1}{\kappa} \sum_{j \in \Gamma(i)} g_j, \quad (2.10)$$

where g_j is the eigenvector centrality of neighbours $j \in \Gamma(i)$.

2.2 Random-graph models

Different generative network models have different algorithms that characterise the rule of interaction between nodes [15, 21, 140]. Thus, the structural properties of those network models depend on the algorithms that are used to generate them. Complicated network generative algorithms make a network difficult to analyse mathematically. Thus, we often begin with simple random graph models that are well studied. In our work, we will use two simple, but popular, random graph models: the *Erdős–Rényi (ER) graph model* [38], and configuration models [41].

2.2.1 Erdős–Rényi graphs

The Erdős–Rényi model is an attractive model because its mathematical properties have been studied extensively in numerous papers, and it has a simple generative algorithm. The generative mechanism to create an ER graph is as follows [38]: let $n = |V|$ be the number of nodes, and suppose that each pair of nodes is adjacent to each other with independent probability p . This model gives a random-graph ensemble

$G(N, p)$ that is a probability distribution on graphs.⁴ In other words, $G(N, p)$ defines an ensemble of networks, not an individual instance of the ensemble. In the ensemble $G(N, p)$, the probability to create a graph G_m with m edges is

$$P(G_m) = p^m (1 - p)^{\binom{N}{2} - m},$$

where $\binom{N}{2} - m$ is the number of pairs of nodes without an edge between them. As edges are independent and identically distributed, the global clustering coefficient⁵ C of the ER model is

$$C = \frac{\text{number of triangles}}{\text{number of connected triples}} = \frac{\binom{N}{3} p^3}{\binom{N}{3} p^2} = p. \quad (2.11)$$

Considering how interactions between entities occur in a real social network, we see that an ER model is not a good model of a social network. Unlike in the ER model, interactions between individuals do not happen completely at random in real-world networks [95]. Instead, interactions often occur for a reason, such as similarity between individuals. Furthermore, ER graphs have a binomial degree distribution,⁶ which is not true in real-world social networks. Therefore, the ER model is not a good model of a social network, as it is unrealistic in many respects. Nevertheless, an ER model is still a good model to test a dynamical process, as its structural properties are well studied, so it is a convenient choice to use at first. Therefore, it is reasonable to start with ER graphs as a toy model of a social network, due to their mathematical tractability.

Mean degree and degree distribution. The probability of drawing a network with m edges from the ensemble $G(N, p)$ is [46]

$$P(m) = \binom{\binom{N}{2}}{m} p^m (1 - p)^{\binom{N}{2} - m}, \quad (2.12)$$

⁴The model was first invented by Anatol Rapoport and Ray Solomonoff [116, 129], and extensively studied by Gilbert [46]. Erdős and Rényi developed a different random graph model, denoted $G(N, m)$, in which N is the number of nodes in a graph and m is the number of edges [38]. However, we also call $G(N, p)$ an Erdős–Rényi graph, as Erdős and Rényi contributed a lot to the study of $G(N, p)$ as well [86, 105].

⁵A clustering coefficient is a measurement to quantify the extent to which nodes in a network form densely connected groups [105]. There is more than one way to calculate the clustering coefficient of a network; for example, the one defined by Watts and Strogatz is widely used [141].

⁶ER random graphs are sometimes called “Poisson random graphs”, because the binomial degree distribution is approximated by the Poisson degree distribution in the $N \rightarrow \infty$ limit.

which is a binomial distribution. The expected number $\langle m \rangle$ of edges of a graph drawn from the ensemble $G(N, p)$ is therefore

$$\langle m \rangle = \sum_m \binom{N}{2} P(m) = \binom{N}{2} p. \quad (2.13)$$

Let k denote the mean degree of a node in an ER graph. The total number of edges m in the graph is then $m = \frac{Nk}{2}$. By inserting this expression into (2.13), we obtain the mean degree of the ensemble $G(N, p)$ of networks. That is,

$$\langle k \rangle = \frac{2 \langle m \rangle}{N} = (N - 1)p. \quad (2.14)$$

Using (2.14), we can also calculate the degree distribution of $G(N, p)$. The probability for a node to have degree k is

$$p_k = \binom{N-1}{k} p^k (1-p)^{(N-1)-k}. \quad (2.15)$$

In many cases, we consider large networks, which generally entails considering $N \rightarrow \infty$ nodes in analytical calculations [105]. One can then approximate $\binom{N-1}{k} p^k$ by $\frac{(N-1)^k p^k}{k!} = \frac{\langle k \rangle^k}{k!}$ using the relation between $\langle k \rangle$ and $(N-1)p$ from (2.14). To calculate $(1-p)^{(N-1)-k}$, we write

$$\ln[(1-p)^{(N-1)-k}], \quad (2.16)$$

which we can then expand using a Taylor series:

$$\ln[(1-p)^{(N-1)-k}] = [(N-1)-k](-p) \approx [(N-1)-k] \frac{\langle k \rangle}{N-1}. \quad (2.17)$$

Using $N \rightarrow \infty$, we obtain $\frac{(N-1)-k}{N-1} \rightarrow 1$. We thus rewrite equation (2.15) as

$$(1-p)^{(N-1)-k} \approx e^{\langle k \rangle}. \quad (2.18)$$

Therefore, (2.15) reduces

$$p_k = \langle k \rangle^k \frac{e^{\langle k \rangle}}{k!}. \quad (2.19)$$

2.2.2 Configuration model

Structural properties of real-world networks are often compared to the properties of an ensemble of random network models to quantify which structural properties are expected (i.e., common consequences of particular degree distribution), and which structural properties are not expected [41]. In this respect, ER networks do not

capture the structural properties that are commonly observed in real-world networks such as heavy-tailed degree distribution [3, 4], and community structure [131]. Furthermore, as we saw in Section 2.2.1, one obtains a Poisson degree distribution as $N \rightarrow \infty$, although there is no *a priori* restriction on the degree distribution of networks in the real world [68]. Because many structural properties are constrained by the distribution of node degrees, it is more appropriate to draw a graph from a space of graphs in which the degrees of all nodes are fixed, but where the edges are placed between nodes uniformly at random [41].

We can resolve this problem using *configuration model*, which was first introduced by Bender and Canfield [11], and further studied by Bollobás [16] and Wormald [147]. A configuration model is a uniform distribution of graphs with specific *degree sequence*, a sequence of node degrees. Therefore, using a configuration model, we can create a network with any desired degree sequence.

There have been many variants of configuration models [41]. Depending on which graph space one draws a graph from — either stub- or node-labelled graph space — a configuration model can give graphs with different properties, especially if a resulting graph contains multi-edges or self-loops [41]. Here we use a configuration model based on stub-labelled graphs, because it is the graph space we use for configuration models in this thesis.

2.2.2.1 Configuration model with stub matching process

The algorithm of stub matching process is somewhat simple: first, each node i is assigned exactly k_i stubs (i.e., half-edges) according to a given degree sequence. One then chooses pairs of stubs uniformly at random and connects them to each other. This process continues until there are no remaining unpaired stubs. The sum of all degrees in a degree sequence must be even so that there is no unpaired stub after the pairing process terminates. We will call a configuration model produced via stub matching as the standard configuration model.

The standard configuration model preserves specific degree sequence, but it randomizes all other structural properties. There has been work that develops random graph models that incorporate particular structural properties, such as clusters [75, 104] and cliques (fully-connected subgraphs) [49]. In Section 2.2.2.2, we will discuss a model that incorporates cliques, which is introduced by James Gleeson in [49].

2.2.2.2 Configuration model with cliques

We start with a standard configuration model with degree distribution $\{P_k\}$, but there is also a joint distribution $\gamma(k, c)$ that specifies the probability that a node chosen uniformly at random has degree k and is in a clique of c nodes (i.e., a c -clique).

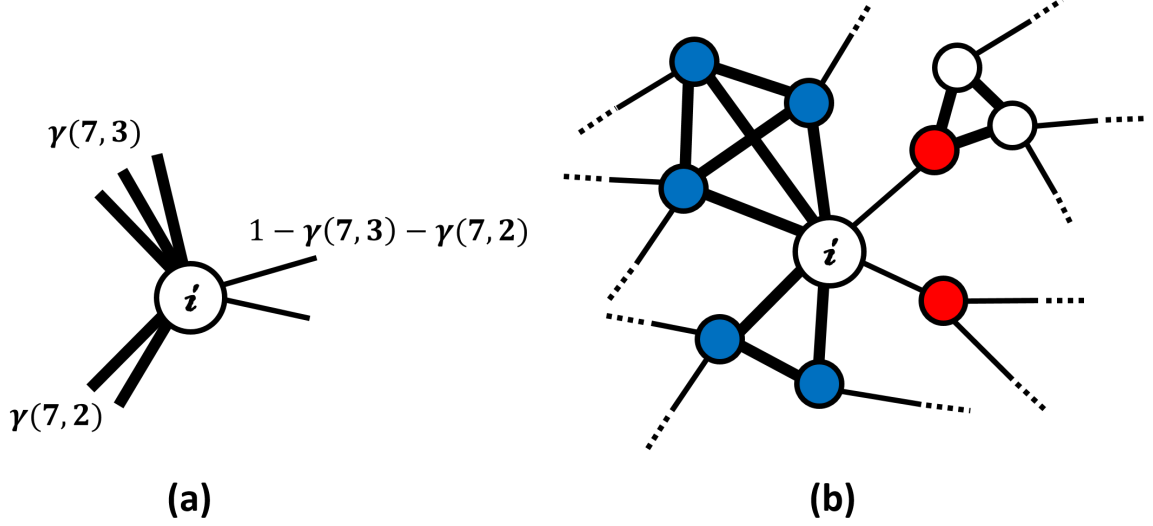


Figure 2.3: An example to illustrate the way a node becomes a part of cliques or becomes adjacent to individual nodes in a configuration model with cliques. The thick lines indicate the internal stubs, and the thin lines indicate the external stubs. (a) The node has degree 7, and 3 of its stubs are chosen with probability $\gamma(7,4)$ and connect the node to a 4-clique, 2 stubs are chosen with probability $\gamma(7,3)$ to connect the node to a 3-clique, and the remaining 2 stubs connect the node to any individual nodes that are not part of a clique. (b) Portion of a network with node i connected to cliques (blue) and individual nodes (red).

Figure 2.3 shows the way a node in a configuration model with cliques becomes a part of cliques or becomes adjacent to individual nodes. For ease of understanding, we use different thicknesses of stubs depending on whether a stub connects a node to a clique or not. The thicker stubs are *internal stubs* that connect nodes to other members of the cliques to which the nodes belong, and the thinner stubs are *external stubs* that connect nodes to nodes other than the members of the clique. Node i in Fig. 2.3a has degree 7. Three of its stubs are chosen with probability $\gamma(7,4)$ and connect the node to a 4-clique, 2 stubs are chosen with probability $\gamma(7,3)$ to connect the node to a 3-clique, and the final 2 stubs connect the node to any individual nodes that are not part of a clique. Such a node becomes a part of a network (see Fig. 2.3b). The degree k of a node should be at least as large as the number of internal stubs $c - 1$ of a clique c to which the node belongs; therefore, $\gamma(k, c) = 0$ for $k < c - 1$.

One can calculate the degree distribution P_k of a network by summing all $\gamma(k, c)$ over all possible c :

$$P_k = \sum_c^{k+1} \gamma(k, c). \quad (2.20)$$

Assume a node i with degree k_i is a part of c -clique; it means the node is a part of $\binom{c-1}{2}$ triangles, and therefore the local clustering coefficient C_i of the node i is

$$C_i = \frac{\sum_{c \in Q_i} \binom{c-1}{2}}{\binom{k_i}{2}} = \frac{\sum_{c \in Q_i} (c-1)(c-2)}{k_i(k_i-1)}. \quad (2.21)$$

where Q_i is the set of cliques that has node i . We can also calculate the degree-dependent clustering coefficient C_k of nodes with degree k in a network as the following:

$$C_k = \sum_c \frac{\gamma(k, c)}{P_k} \frac{(c-1)(c-2)}{k(k-1)}. \quad (2.22)$$

2.3 Social influence models and Watts threshold model

In social networks, individuals are influenced by their friends, neighbours, and other contacts for various reasons. For example, in industry, a company adopts an innovation introduced by other companies; in school, various rumours spread from one student to another. *Social influence* occurs when an individual is influenced by other individuals (e.g., neighbours, friends, or colleagues) and changes his/her behaviour by adopting — and possibly also adapting — the others' behaviour [84, 118]. According to Polansky et al. [112], there are two different kinds of social influence: *behavioural contagion* and *direct influence*. A behavioural contagion occurs when the original influencer actually does not have an intention to influence the others. In short, it happens when an individual only passively (or indirectly) influences the others. In contrast, direct influence occurs when an actor actively and intentionally influences the others.

*Social influence models*⁷ are mathematical models to portray social influence. These include the SI (susceptible–infected) model, voter models [24, 71, 87], and the Watts threshold model (WTM) [137]. Specifically, the SI model and WTM are

⁷A social influence model is also often called a “social-contagion model” [84], because the spreading characteristic of social influence is similar to a biological contagion process.

contagion models that caricature the way that individuals in society influence each other [113]. The WTM was created to explain large and rare cascade phenomena that are triggered by a small number of initiators, such as fads [14] and the spread of information in social networks [81].

Nodes in the WTM have two main attributes: a *state* and a *threshold*. The state $s_i(t)$ indicates the state of node i at time t ; it is either 1 (adopted) or 0 (unadopted). There are two main versions of the WTM:⁸ to activate a node i from state 0 to state 1, the threshold ϕ_i of the node should be smaller than or equal to⁹

1. the absolute number of neighbours in state 1 (absolute threshold [26, 44]), or
2. the fraction of neighbours in state 1 (fractional threshold [137]).

The main difference between the absolute-threshold model and the fractional-threshold model is whether nodes are influenced by both activated and non-activated neighbours or only by activated neighbours [26]. In the absolute-threshold model, it is only a node's activated neighbours that determine whether it becomes activated. A node becomes activated when the number of activated neighbours is greater than or equal to a certain level. In contrast, in the fractional-threshold model, a node becomes activated when the ratio of a node's activated neighbours to the total number of neighbours is greater than or equal to a certain level. As the activated fraction indicates the relative number of activated neighbours versus unactivated neighbours, both the number of activated neighbours and the number of non-activated neighbours matter.

We give the basic algorithm of Watts threshold dynamics in Algorithm 1. Initially, a small fraction ρ_0 of nodes are chosen uniformly at random and are activated to state 1 at time $t = 0$, while all other nodes are in state 0. We call the activated nodes at $t = 0$ the *seed nodes*, and we call ρ_0 the seed fraction. We then perform the cascade algorithm synchronously¹⁰, until no further nodes in the network can be activated, reaching a *steady state*.

Node thresholds can be either heterogeneous or homogeneous. In the homogeneous case, all nodes have the same threshold, whereas in the heterogeneous case, nodes have different thresholds. In real-world situations, nodes with low thresholds

⁸There are also various generalizations of these two threshold models [33, 43, 94, 99, 109].

⁹The WTM has *monotone* dynamics [50]; once a node “adopts”, its state changes from 0 to 1, and its state cannot change back to 0. This property is sometimes called the *permanently active property* [52].

¹⁰In synchronous updating, a node changes its state based on its neighbours' states at the previous time step [113]. One can also consider asynchronous cascade models [48, 51].

Algorithm 1 Algorithm for Watts threshold dynamics. See the text for details.

```

 $t == 0$ 
while  $N \times \rho_0 > \sum_{i \in V} s_i(t)$  do
    Draw  $i \in V$  uniformly at random
     $s_i(t) == 1$ 
end while
 $\rho(t) == \sum_{i \in V} s_i(t)$ 
 $t == 1$ 
while  $t > 0$  do
    for  $i \in V$  do
        if  $\phi_i(t-1) < \frac{\sum_{j=1}^N A_{ij} s_j(t-1)}{k_i}$  then
             $s_i(t) == 1$ 
        else
             $s_i(t) == 0$ 
        end if
    end for
     $\rho(t) == \sum_{i \in V} s_i(t)$ 
    if  $\rho(t) = \rho(t-1)$  then
        Break
    end if
     $t = t + 1$ 
end while

```

may represent people who are easily influenced, while nodes with high threshold may represent people who are more resistant against external influence. In marketing literatures [101], consumers are divided into different groups, such as innovators, early adopters, late adopters, depending on how quickly a consumer adopts a new technology, consumes a new product, etc. Innovators are those who buy a new product and adopt it very shortly after the new product is released, early adopters are people who consume the product and adopt it later than innovators but faster than other people, and late majorities are those who wait until a market cools down. In terms of thresholds, innovators can be interpreted as seed nodes whose state $s = 1$ at time $t = 0$, so they buy the new product when nobody else buys it. Their consumption behaviour influences others, and those who consume the product a time step after a few of their neighbours buy it can be interpreted as early adopters. The rest, with high thresholds, who consume goods after a market stabilizes can be interpreted as late adopters.

2.3.1 Numerical simulations of the WTM

We investigate how the fraction of activated nodes at steady state changes depending on the heterogeneity of the threshold values. In this section, we duplicate numerical results from Watts' original paper [137]. An important thing to note in this numerical study is that the properties of dynamical systems on ER networks cannot be determined by a single realization of a network, because an ER network model defines an ensemble of networks (many realizations of networks). For our figures, we therefore indicate how many realizations of ER networks that we run. Also, we indicate other initial conditions.

In our numerical calculations, we use the $G(N, p)$ model with $|V| = N$ and edge probability p . Each node is assigned a threshold ϕ drawn from a probability distribution $f(\phi)$ that is normalized such that $\int f(\phi)d\phi = 1$. Depending on the mean and variance of $f(\phi)$, some nodes can have negative threshold values. They are innovators, and they are activated at $t = 0$ regardless of their number of activated neighbours.

Figure 2.4a is a z - ϕ phase diagram of WTM on ER networks, where the thresholds are homogeneous, and Fig. 2.4b is a phase diagram with heterogeneous thresholds. For heterogeneous thresholds, the thresholds are distributed as a Gaussian with mean ϕ and standard deviation of $\sigma = 0.2$. We call the border between the area where the *global cascade*¹¹ occurs (bright region in Fig. 2.4) and the area where the global cascade does not occur (dark region in Fig. 2.4) as the *cascade boundary*, and the bright area enclosed by the cascade boundary is called the *cascade region* [137].

Both panels of Fig. 2.4 share a similar characteristic that the range of the mean threshold ϕ in which a global cascade occurs becomes narrower with increasing mean degree z . With large mean degree, a node chosen uniformly at random has a small fraction of seed neighbours. In other words, the influence of the seed nodes decreases if we increase the mean degree of ER network while keeping the seed fraction ρ_0 constant. Hence, it is natural that the range of the mean threshold ϕ in which a global cascade can occur decreases with increasing mean degree z in both panels of Fig. 2.4.

The difference between the two panels in Fig. 2.4 is that in the homogeneous-threshold case (Fig. 2.4a), the shape of a cascade boundary is stair-like, whereas if

¹¹The definition of a cascade is different in different papers. Some define it to have all of the nodes in the largest connected component — the connected component that has the largest number of nodes in a network — to be in adopted state. Some loosely use the term ‘cascade’ as a phenomenon in which some fraction of nodes in networks adopt [47, 89]. In this thesis, we define (*global*) *cascade* as a phenomenon in which a small seed fraction of adopters leads to a much larger adopted fraction that scales with the size of networks.

the thresholds are distributed heterogeneously via a Gaussian distribution (Fig. 2.4b), the cascade boundary is smoother than in the homogeneous threshold case. Additionally, the size of cascade region is larger than the case of homogeneous threshold. In particular, for Gaussian distributed thresholds, cascade occurs in networks with high mean threshold; it is because even if the mean threshold is high, there can exist nodes with low thresholds for Gaussian distributed thresholds. Therefore, for constant mean degree z in ER networks, the range of ϕ for which a global cascade can occur is wider in the heterogeneous threshold case than in the homogeneous threshold case.

We speculate that we can apply the argument to other networks, because a heterogeneous threshold leads to a larger *vulnerable cluster*, which we will discuss in Section 2.3.2.3.

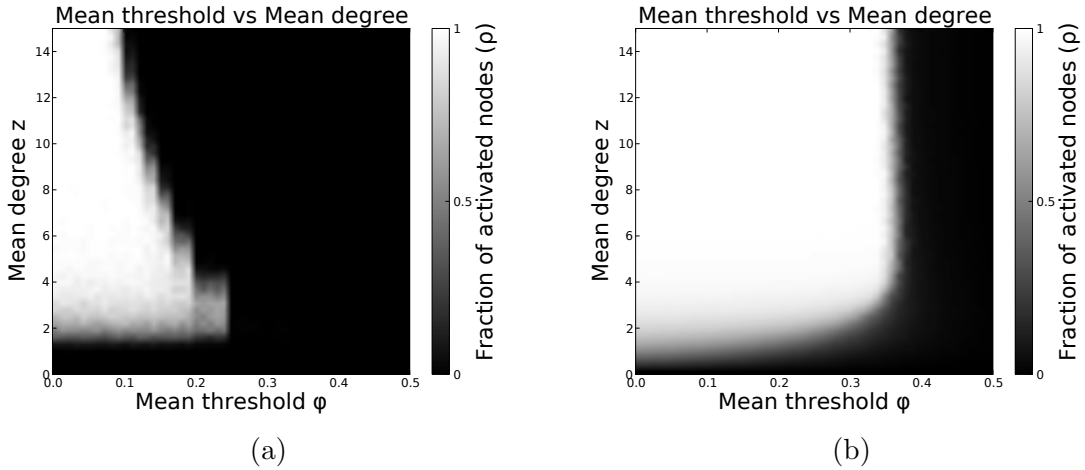


Figure 2.4: ϕ - z phase diagram of WTM, where nodes have (a) homogeneous thresholds or (b) heterogeneous thresholds. Brightness indicates the final fraction ρ of activated nodes, and brighter colours indicate higher ρ values. For both panels (a) and (b), we run the WTM on ER networks of size (i.e., the number of nodes) $N = 1,000$. At time $t = 0$, a seed fraction $\rho_0 = 0.001$ of N nodes is chosen uniformly at random and their states become 1, and the rest of the nodes have state 1. We average the final fraction of activated nodes ρ over 100 realizations of ER networks for every value of (ϕ, z) . There are 50×50 number of coordinates in total, in which each axis has 50 coordinates. In panel (a), all nodes' thresholds are uniform, so $\phi_i = \langle \phi \rangle$ for all $i \in V$. In panel (b), we draw the thresholds from a Gaussian distribution with mean $\langle \phi \rangle$ and standard deviation $\sigma = 0.2$. All nodes are updated synchronously at every time step.

To do a more detailed inspection, we slice both panels in Fig. 2.4 through the vertical lines $\phi = 0.1$, $\phi = 0.15$, and $\phi = 0.2$ and the horizontal lines $z = 3$, $z = 4.5$,

and $z = 6$. By doing so, we can trace how the final activated fraction ρ changes by varying only the mean threshold ϕ or only the mean degree z .

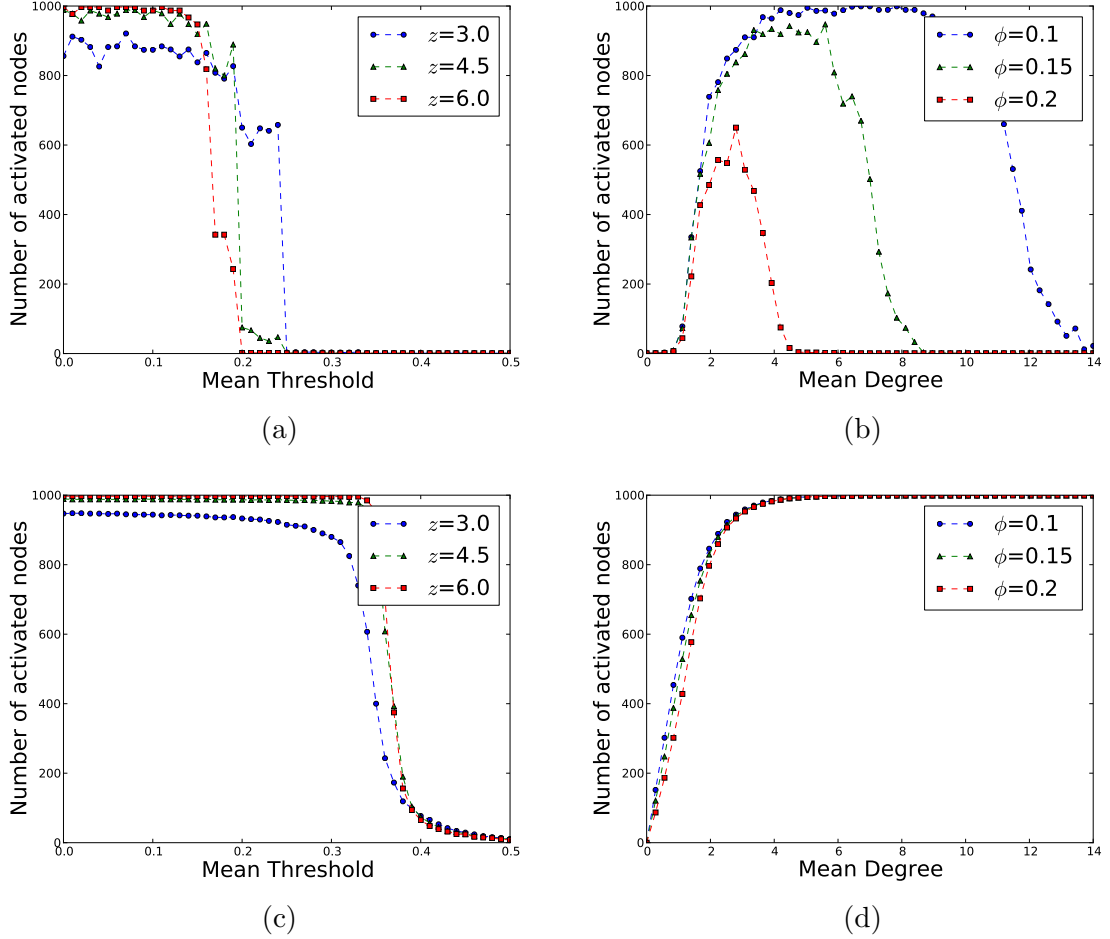


Figure 2.5: Horizontal and vertical slices of Figs. 2.4a and 2.4b. Panels (a) and (b), respectively, are horizontal slices of ρ versus ϕ with homogeneous and heterogeneous thresholds. The mean degrees are $z = 3$ (blue circle), 4.5 (green triangle), and 5.0 (red square). Panels (c) and (d), respectively, are vertical slices ρ versus z with homogeneous and heterogeneous thresholds. The mean thresholds are $\phi = 0.1$ (blue circle), 0.15 (green triangle), and 0.2 (red square).

From Figs. 2.5a and 2.5c, we see for the WTM on ER networks that the size of the activated fraction decays earlier with increasing mean threshold for homogeneous thresholds than for Gaussian-distributed heterogeneous thresholds. In this case, if the ensemble of ER networks has a high mean degree, the activated fraction tends to decay earlier than when it has low mean degree (see Fig. 2.5a). However, in the heterogeneous-threshold case, if the ensemble of ER networks has a high mean degree, the activated fraction tends to decay later than when it has low mean degree (see Fig.

2.5c). A fully activated component can occur when the ensemble of ER networks has a high mean degree, because the number of small components (the components that are not connected to the largest connected component) is smaller if the mean degree z is larger.

For plots of ρ versus z , one can see that lower mean threshold leads to a higher probability of a global cascade for both homogeneous and heterogeneous thresholds. For $z \in (0, 15)$, the fraction of activated nodes in ER networks with heterogeneous thresholds does not decay with increasing mean degree, whereas it does for networks with homogeneous thresholds. As we have discussed previously, the mean ratio of a node's seed neighbours to neighbours that are not seeds decreases as the mean degree increases. Moreover, if the thresholds are heterogeneous, then there can exist a node that has low enough threshold or a node that has threshold less than or equal to 0 to become activated at time 1 even if the node has a very small number of seed neighbours compared to the number of neighbours that are not seeds. The number of such nodes increases as the mean threshold decreases. However, in the homogeneous-threshold case, all nodes have the same threshold. Therefore, if both the threshold and the mean degree of nodes in a network are too high, then there does not exist a single node to extend the influence from the seed nodes. That is the reason that global cascades in ER networks with homogeneous threshold decay earlier than in ER networks with a heterogeneous threshold. We will show this mathematically in Section 2.3.2.

2.3.2 Analytics of Watts threshold model

As we can see from our numerical simulations, the mean degree of ER networks and the existence of nodes with small threshold are important factors for triggering a cascade in ER networks. We will now see how the previous results can be examined analytically — in particular, to forecast whether or not a global cascade occurs. We follow the discussion and derivation in [106, 137].

Before we go into our analysis of the WTM, we consider *site percolation* in graphs. Site percolation is useful for undertaking an analytical study of the WTM, because of the similarity between site percolation and cascade dynamics of the WTM [137]. In the simplest problem in site percolation, a node (site) is occupied with probability q and empty with probability $1 - q$, independently of others. A question of interest is to calculate the probability that there is a *giant percolating component* — a giant component, which scales with the size of a network, with occupied nodes only — for a given probability q . Let p_k denote the probability that a node has degree k , and

let q_k be the probability that a node with degree k is occupied. The quantity $q_k p_k$ is then the probability of a node having degree k and being in the occupied state. We can use *probability generating functions* [145] to calculate a ‘percolating cluster’ — a connected component that is composed only of occupied nodes.

2.3.2.1 Probability generating functions

Probability generating functions are formal power series whose coefficients encode a given probability distribution [145]. Let X be a non-negative, discrete random variable with probability mass function $p_r = \mathbb{P}(X = r)$. The probability generating function $G(z)$ for X is defined as

$$G(z) = \langle z^X \rangle = \sum_{r=0}^{\infty} p_r z^r, \quad (2.23)$$

where p_r denotes the probability that X is equal to r . Here, we normalize the quantity p_r such that $G(1) = \sum_{r=0}^{\infty} p_r = 1$.

As the degree distribution of a network also follows some probability distribution, we use the generating function (2.23) to describe it [106]. We write

$$G_0(z) = \sum_{k=0}^{\infty} p_k z^k, \quad (2.24)$$

where p_k is the probability that a node has degree k . Thus, the set of coefficients of (2.24) represents the *degree distribution*. The probability p_k is normalized such that $G_0(1) = 1$. The mean degree of a node can be calculated by taking the first derivative of (2.24):

$$G'_0(1) = \sum_k k p_k = \langle k \rangle = z. \quad (2.25)$$

Now we would like to examine the degree distribution of the *first neighbours*¹² of a node. We define first neighbours of a node as neighbours that can be reached by following the edges of a node. Suppose that we choose a node i uniformly at random and suppose that its degree is k . Then, we follow its edges to its k first neighbours; those k first neighbours will have each of their neighbours (*second neighbours*¹³ of node i), the number of which follows the distribution of remaining outgoing edges of the first neighbour. A node with high degree has a higher probability to be a

¹²In some literature, a first neighbour is called an *immediate neighbour* (or simply neighbour) [106].

¹³Following the definition of the first neighbours, the second neighbour means the first neighbours’ first neighbours. Thus, the degree distribution of the first neighbours is equal to the probability distribution of the number of second neighbours of a node that is chosen uniformly at random.

neighbour of the node chosen uniformly at random, and the probability of finding such a neighbour is positively related to its degree k . The distribution of outgoing edges of first neighbours is generated by the function

$$G_1(x) = \frac{\sum_k k p_k x^{k-1}}{\sum_k k p_k} = \frac{G'_0(x)}{G'_0(1)} = \frac{1}{z} G'_0(x). \quad (2.26)$$

We use $\sum_k k p_k$ in the denominator to normalize the distribution.

2.3.2.2 Application of probability generating functions to site percolation

In this section, we present the use of generating functions for site-percolation problems in graph theory. Note that we follow the discussion in [22], so this is not our derivation. Using (2.23), we write the probability generating function for the distribution of $q_k p_k$ as

$$F_o(x) = \sum_{k=0}^{\infty} q_k p_k x^k. \quad (2.27)$$

As in (2.26), we can also calculate the probability generating function for the distribution of $q_k p_k$ for the first neighbours:

$$F_1(x) = \frac{\sum_k k q_k p_k x^{k-1}}{\sum_k k p_k} = \frac{F'_o(x)}{G'_0(1)} = \frac{1}{z} F'_o(x). \quad (2.28)$$

In site percolation, a *percolating cluster* is a connected component that is composed only of occupied nodes. Let $H_1(x)$ be the generating function for the probability that an edge chosen uniformly at random is connected to a percolating cluster. Therefore,

$$H_1(x) = \sum_s r_s x^s, \quad (2.29)$$

where s is the size¹⁴ of the finite percolating cluster and r_s is the probability that the size of a percolating cluster at the end of the edge is s . If a node at the end of the edge is unoccupied, the cluster includes 0 nodes with probability $1 - F_1(1)$. The probability that one end of the edge is connected to an occupied node is $F_1[H_1(x)]$. Therefore, $H_1(x)$ satisfies the following self-consistency condition:

$$H_1(x) = 1 - F_1(1) + x F_1[H_1(x)]. \quad (2.30)$$

¹⁴Analogous to our terminology for the size of a graph, we use the term ‘size’ as the number of nodes in the percolation cluster.

Similarly, the generating function $H_0(x)$ for the probability that a node chosen uniformly at random belongs to a finite percolating cluster is

$$H_0(x) = \sum_s u_s x^s, \quad (2.31)$$

where u_s is the probability that the node belongs to a finite percolating cluster of size s . The analogue of (2.30) for $H_0(x)$ is

$$H_0(x) = 1 - F_0(1) + x F_0[H_1(x)]. \quad (2.32)$$

Note that the term $F_0[H_1(x)]$ has $H_1(x)$ as an argument.

As we will discuss in Section 2.3.2.3, equations (2.30) and (2.32) are useful for calculating a condition to trigger a cascade in the WTM.

2.3.2.3 Analytical prediction of a global cascade

In this section, we conduct real analysis to calculate the cascade boundary of WTM on undirected unweighted networks in the $N \rightarrow \infty$ limit. To analyse cascade phenomena in the WTM, we define a *vulnerable node* as a node with $\phi_i < \frac{1}{k_i}$, where ϕ_i is the threshold of a node i and k_i is the degree of a node i . The defining characteristic of a vulnerable node is that it becomes activated if at least one of its neighbours is an innovator whose state $s = 1$ at time $t = 0$.

Whether or not a cascade can occur depends on the size of the *vulnerable cluster*, which is a connected component that is composed of vulnerable nodes. To calculate the size of a vulnerable cluster, we use the same approach as we did when considering site percolation in Section 2.3.2.2. Specifically, we use generating functions [106, 137, 145]. Let q_k be the probability of having a threshold lower than $\frac{1}{k}$ [i.e., $q_k = \mathbb{P}(\phi < \frac{1}{k})$]. The generating function of a vulnerable node degree is then

$$G_0(x) = \sum_{k=0}^{\infty} q_k p_k x^k, \quad (2.33)$$

where

$$q_k = \begin{cases} 1, & \text{if } k = 0, \\ F\left(\frac{1}{k}\right), & \text{if } k > 0. \end{cases}$$

The function $F(\frac{1}{k})$ is the fraction of nodes in which the threshold is less than or equal to $\frac{1}{k}$. Thus,

$$F\left(\frac{1}{k}\right) = \int_{-\infty}^{\frac{1}{k}} f(\phi) d\phi. \quad (2.34)$$

We can see that (2.33) is the same equation as the generating function for an occupied node given degree k [see equation (2.27)]. In equation (2.27), q_k was the probability that a node is occupied for a given k , but now, in (2.33), q_k is the probability that a node is vulnerable for a given k . Therefore, if we follow the derivation in Section 2.3.2.2, then the WTM can be reduced to a percolation problem, and we can go through the same process as we did for site percolation. We can thus use the equations that we derived from (2.27) for $H_0(x)$ and $H_1(x)$. The meaning of $H_0(x)$ and $H_1(x)$ is slightly different now: $H_0(x)$ represents the probability generating function for a node chosen uniformly at random to belong to a vulnerable cluster of finite size s , and $H_1(x)$ is the probability generating function for an edge chosen uniformly at random to lead to a vulnerable cluster of finite size s . Again, $H_0(x)$ and $H_1(x)$ satisfy self-consistency conditions:

$$H_1(x) = 1 - F_1(1) + xF_1[H_1(x)], \quad (2.35)$$

$$H_0(x) = 1 - F_0(1) + xF_0[H_1(x)]. \quad (2.36)$$

Because we can obtain the mean degree $z = \langle k \rangle$ by taking the first derivative of the generating function $G_0(x)$ for the degree distribution, we can calculate the mean size of a finite vulnerable cluster by taking the first derivative of the generating function for the finite vulnerable cluster size [i.e., $\langle s \rangle = H'_0(1)$]. By inserting (2.35) into (2.36), we obtain

$$\langle s \rangle = G_0(1) + \frac{[G'_0(1)]^2}{z - G''_0(1)} = G_0(1) + \frac{z^2}{z - G''_0(1)}, \quad (2.37)$$

where $G_0(1)$ is the vulnerable fraction of the nodes. Equation (2.37) diverges when

$$G''_0(1) = \sum_k^{\infty} k(k-1)q_k p_k = z, \quad (2.38)$$

where $G''_0(1)$ is the mean number of vulnerable second neighbours [106], and $z = G'_0(1)$ is the mean degree of the vulnerable nodes. If $G''_0(1) < z$, then the finite size of the vulnerable cluster is not sufficiently large for a global cascade to occur. The mean size $\langle s \rangle$ of a vulnerable cluster diverges when $G''_0(1) = z$, which implies that the size of the vulnerable cluster becomes infinite. Also, if $G''_0(1) > z$, then a finite-size vulnerable

cluster is absent from a network¹⁵, implying that the size of the vulnerable cluster approaches infinity as the network size tends to infinity. Therefore, the size of the vulnerable cluster becomes infinite if $G_0''(1) \geq z$, and a global cascade occurs. We obtain a *cascade condition* based on satisfying the following inequality:

$$\sum_k^{\infty} k(k-1)q_k p_k \geq z. \quad (2.39)$$

We now analytically calculate the boundary of the cascade region in Fig. 2.4. The network is an ER graph, where p_k can be approximated to be $p_k = e^{-z} \frac{z^k}{k!}$ if N is large enough. The thresholds are Gaussian distributed with a standard deviation of $\sigma = 0$ for homogeneous thresholds and a standard deviation of $\sigma = 0.2$ for the heterogeneous thresholds that we have used in our example. The quantity q_k for $k > 0$ is given by the cumulative distribution function for the Gaussian distribution:

$$q_k = F\left(\frac{1}{k}\right) = \frac{1}{2} \left[1 + \operatorname{erf}\left(\frac{\frac{1}{k} - \mu}{\sqrt{2}\sigma}\right) \right], \quad (2.40)$$

where $\operatorname{erf}(x)$ is the error function.

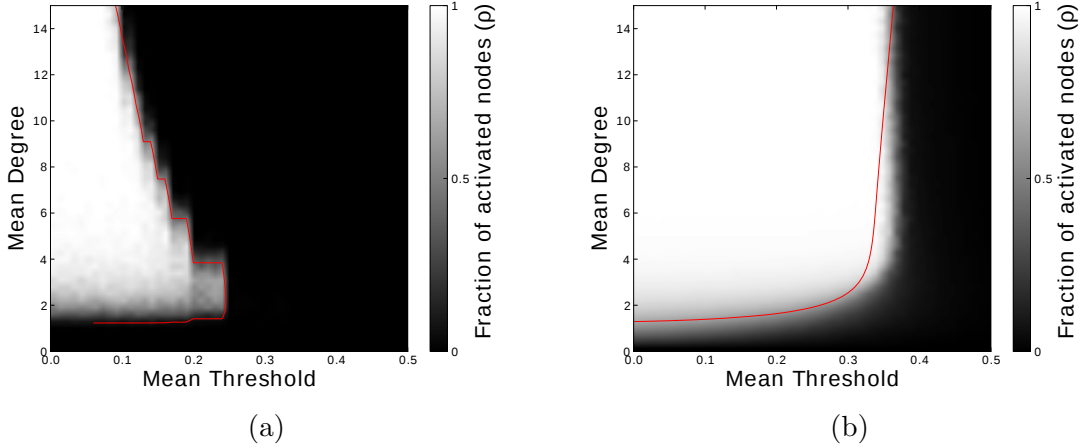


Figure 2.6: ϕ - z phase diagram of WTM as in Fig.2.4 and analytical cascade boundary (red curve). All node thresholds are (a) equal to 0.2 or (b) drawn from a Gaussian distribution with mean threshold $\langle \phi \rangle = 0.2$ and standard deviation $\sigma = 0.2$. The number N of nodes is 1000, and the fraction ρ_0 of seed nodes is 0.001, where we choose seed nodes uniformly at random. We averaged over 100 realizations of ER networks and obtain the final fraction ρ of activated nodes. There are 50×50 number of coordinates in total, in which each axis has 50 coordinates.

¹⁵For $G_0''(1) \geq z$, the mean size $\langle s \rangle$ of a finite vulnerable clustering is less than 0, but this is not a quantitative argument. It demonstrates the bidirectionality of the scaling of the size of the finite vulnerable clustering.

In Fig. 2.6 we show the analytical cascade boundary (red curve) calculated from (2.38) on top of the phase diagram of ϕ - z for Fig. 2.4. In Fig. 2.6a, the threshold is homogeneous (i.e., $\sigma = 0$), whereas $\sigma = 0.2$ in Fig. 2.6b. Equation (2.38) assumes that $N \rightarrow \infty$, and it allows us to assume that on average the ER networks include only a small number of short cycles, as we discussed earlier. However, in our numerical simulations, because the sizes of the ER networks are finite (specifically, $N = 1000$), the number of cycles is non-negligible. Therefore, the cascade boundary of the phase diagram does not completely agree with the analytical cascade boundary. Nevertheless, the actual cascade boundary and the analytical cascade boundary calculated by (2.38) are very similar to each other.

2.4 Summary

In this chapter, we covered the mathematical background that is necessary to understand the rest of the thesis. We started by introducing graph theory, where we defined some measures of network structure. We then introduced random network models — Erdős–Rényi networks and configuration-model networks — which are the primary types of network models for our work in this thesis. Additionally, we introduced the Watts threshold model (WTM). In the next chapter, we will generalise the WTM, introduce the idea of a *timer*, incorporate timers into the WTM and examine novel features that arise in WTM dynamics.

Chapter 3

Incorporating timers into the WTM

A great deal of effort has gone into trying to model social influence — including the spread of behaviour, norms, and ideas — on networks. Most models of social influence tend to assume that individuals react to changes in the states of their neighbours without any time delay, but this is often not true in social contexts, where (for various reasons) different agents can have different response times. To examine such situations, we introduce the idea of a timer into threshold models of social influence. The presence of timers on nodes delays the adoption — i.e., change of state — of each agent, which in turn delays the adoptions of its neighbours. With a homogeneously-distributed timer, in which all nodes have the same amount of delay, adoption delays are also homogeneous, so the adoption order of nodes remains the same. However, heterogeneously-distributed timers can change the adoption order of nodes and hence the “adoption paths” (see Section 3.1.2 and 3.2) through which state changes spread in a network. Using a threshold model of social contagions, we illustrate that heterogeneous timers can either accelerate or decelerate the spread of adoptions compared to an analogous situation with homogeneous timers, and we investigate the relationship of such acceleration or deceleration with respect to timer distribution and network structure. We derive an analytical approximation for the temporal evolution of the fraction of adopters by modifying a pair approximation of the WTM, and we find good agreement with numerical computations.

3.1 Timer models

According to the Oxford English Dictionary, a “timer” is an automatic mechanism for activating an object at a preset time [32]. We apply this concept to a discrete-stage

social-influence model on a network by defining a “timer model” as a model in which a node adopts an innovation after a preset number of discrete time steps, where a countdown starts after some other condition (e.g., peer pressure on the node matching or exceeding some other threshold) has been met. In a timer model, each node in a network has an associated timer that is drawn from some probability distribution. Once the timer of a node is “triggered” (e.g., when the node meets some adoption condition), its counter starts counting down to 0 the next time the node is updated, and it changes its state when its timer hits 0. We use a discrete-time setting, so each timer starts decrementing one time unit after all other adoption conditions are satisfied. For example, if node v_i ’s timer $\tau_{v_i} \in \mathbb{Z}_{\geq 0}$ is triggered at time $t = t'_{v_i}$, then it changes its state at time $t = t'_{v_i} + \tau_{v_i} + 1$, as that is one time unit after the timer countdown reaches 0. This is the first time that node v_i is considered for updating of its state after its “timer-adoption condition” (at time $t = t'_{v_i} + \tau_{v_i}$) occurs.

If there is no adoption condition other than a timer-adoption condition, then the timers of all nodes are triggered at time step $t = 0$ (so they start decrementing at $t = 1$), and the adoption process terminates when the largest timer hits 0. Therefore, the time of adoption of the node is equal to the timer of a node, and the cumulative distribution function of the timer distribution describes the adoption process; network structure plays no role in this case. Such a naive timer model already illustrates Everett Rogers’s ideas about the spread of innovations [119, 120], in which different adopter categories (innovators, early adopters, early majority, late majority, and laggards) are determined only by their different adoption times.

We are interested in incorporating the idea of timers (especially heterogeneous ones) into models of spreading, and in this thesis we use the WTM for concreteness.

3.1.1 WTM with timers

One can add a timing mechanism to any social-influence model in which nodes adopt an innovation based on the states of other nodes in a network. Let us consider what happens if we add a timing mechanism to the WTM. The WTM is a binary-state model (see Section 2.3) in which a node can have state $s \in \{0, 1\}$. The WTM has monotonic dynamics, as a node’s state can change from 0 to 1, but any node that attains state 1 remains at that state for all time. When a node first changes its state from 0 to 1, we say that it “adopts” some behaviour or idea. The adoption condition of a node in the WTM is that at least a fraction ϕ_{v_i} of its neighbours have previously adopted the behaviour. The parameter ϕ_{v_i} is the “threshold” for node v_i , and the

condition that at least this fraction of v_i 's neighbours have adopted is the threshold-adoption condition. In the WTM with timers, a node v_i must meet its threshold-adoption condition and also its timer-adoption condition to adopt a behaviour: the fraction of adopted neighbours of node v_i must be at least its threshold ϕ_{v_i} and then its timer τ_{v_i} must hit 0. Update rules for nodes can either be synchronous or asynchronous [113]; in our study, we use synchronous updating, in which all nodes are updated simultaneously during each discrete time step.

3.1.2 Adoption paths

We study the WTM with timers on undirected, unweighted networks; and we trace what we call *adoption paths*, which are directed paths in a network through which an adoption is transmitted (see Fig. 3.1). A directed path in a network is a sequence $(v_1, v_2, \dots, v_n)$ of nodes in which all nodes are distinct, and for $i \in \{1, \dots, n-1\}$, node v_i is adjacent to v_{i+1} via an edge from v_i . The length $l = n-1$ of a path is the number of edges that comprise the path. In an adoption path, node v_i is adjacent to v_{i+1} only if the adoption of v_i triggers the timer $\tau_{v_{i+1}}$ of v_{i+1} . We call v_1 the “root” of the adoption path, as it initiates the spread of adoptions.¹ The time t'_{v_i} at which the timer of a node v_i is triggered is the sum of timers of all preceding nodes in the adoption path plus the number of synchronous time steps to trigger the timers of preceding nodes:

$$t'_{v_i} = \sum_{j=1}^{i-1} \tau_{v_j} + i - 1. \quad (3.1)$$

In Section 3.3.2, we use the unidirectional property of adoption paths to scrutinize how adoptions spread on large networks for the WTM with timers.

An adoption path in the WTM with timers terminates when

- (i) the last node to adopt has degree 1, or
- (ii) all neighbours of the last node to adopt have their timers triggered before the last node adopts.

In Fig. 3.1, we show an example of different adoption paths that terminate after meeting either condition (i) or condition (ii). The dark brown node v_0 in the centre is a seed node that has state 1 at time $t = 0$, and the other nodes are in state 0 at $t = 0$. Each node v_i is assigned a timer τ_{v_i} and a threshold ϕ_{v_i} , where $i \in \{1, \dots, 5\}$. Node

¹We use the term “root” in a different way from its conventional usage in graph theory.

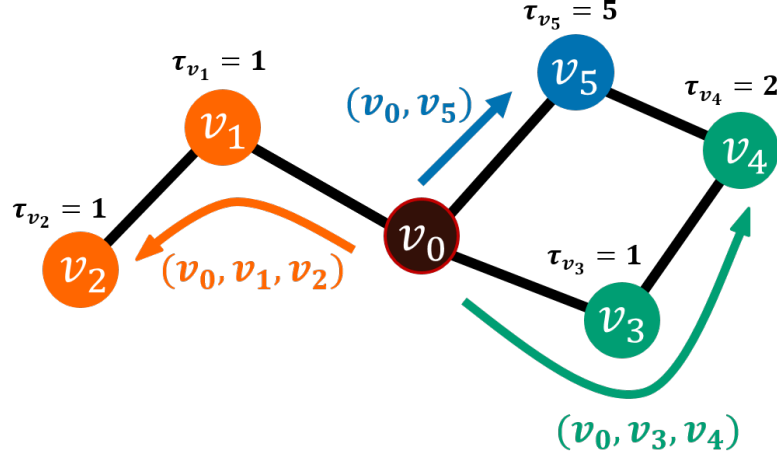


Figure 3.1: Illustration of different adoption paths in the WTM with a timer. Each node v_i is assigned a timer τ_{v_i} and a threshold ϕ_{v_i} , where $i \in \{1, \dots, 5\}$. We initiate node v_0 in the state 1, so we do not need to assign it a threshold or a timer. Suppose that all nodes v_i (with $i \in \{1, \dots, 5\}$) have the same threshold $\phi = 0.5$. An adoption of any node immediately triggers the timers of its neighbours.

v_0 is the seed node, so it starts in the adopted state. Nodes v_i (for $i \in \{1, \dots, 5\}$) have a threshold $\phi = 0.5$, so an adoption of any node immediately triggers the timers of its neighbours. The number written above each node indicates its timer value. If we run the WTM with timers on this network, we obtain three adoption paths: (v_0, v_1, v_2) , (v_0, v_3, v_4) , and (v_0, v_5) . The lengths of these three adoption paths are 2, 2, and 1, respectively. All adoption paths grow from the seed v_0 , which is the root of all adoption paths. Each coloured arrow represents the spread of an adoption through a distinct adoption path. The orange curved arrow terminates at time $T_{(v_0, v_1, v_2)} = 4$; the last node to adopt has degree $k = 1$, so it has no more neighbouring nodes to influence (i.e., condition (i) is satisfied). The green arrow terminates at time $T_{(v_0, v_3, v_4)} = 5$, and the blue arrow terminates at time $T_{(v_0, v_5)} = 6$. Each of these adoption paths exhibits one of the two possible scenarios to satisfy condition (ii): for the blue adoption path, all neighbours of the last node to adopt are in state 1; for the green adoption path, all neighbours of the last node to adopt are in state 0, but the timers of all neighbours have already been triggered. The former scenario is also the condition for the termination of an adoption path in the original WTM, but the latter scenario is a novel feature of the WTM with timers.

3.2 WTM with timers on small networks

Clearly, adding timers to the WTM in general will delay adoption processes. The presence of a homogeneous timer merely delays the adoption of each node for exactly the same number of time steps. Suppose that we run the WTM without timers on an arbitrary network, and it takes T_{WTM} time steps to reach a steady state, in which no more nodes can adopt. On the same network, if we run the WTM with homogeneous timers τ_{hom} , it now takes $T_{\text{WTM}}(\tau_{\text{hom}} + 1)$ time steps to reach a steady state, as every node is delayed by the same amount of time. However, if timers are heterogeneous, different nodes have their adoptions delayed by different amounts of time, and it is less straightforward to calculate the time to steady state in relation to T_{WTM} .

We calculate the time T_{hom} to steady state for the WTM with homogeneous timers and the mean time $\langle T_{\text{het}} \rangle$ to steady state for the WTM with heterogeneous timers in three small example networks in Fig. 3.2. Suppose that all nodes have a homogeneous threshold of $\phi = 0.1$, which is small enough so that any node in Fig. 3.2 adopts the state $s = 1$ once even one of its neighbours is in the adopted state². Using a homogeneous threshold enables us to disentangle the effect of timers from the effect of a heterogeneous threshold. Because all nodes have the same positive threshold, we need a seed (node v_0 in Fig. 3.2) in state 1 at time $t = 0$ to initiate the spread of adoptions.

In Fig. 3.2a, we show a one-dimensional (1D) lattice with a seed node at its left end, so adoption occurs from left to right. The time T to steady state is

$$T = 3 + \tau_{v_1} + \tau_{v_2} + \tau_{v_3}, \quad (3.2)$$

where τ_{v_i} is the timer of node v_i . We need to add 3 because there are 3 nodes other than the seed, and it takes 1 synchronous time step to trigger the timer of a node. In Table 3.1, we show results for this network when the nodes have a homogeneous timer with $\tau_{\text{hom}} = 4$ and heterogeneous timers $\tau_{\text{het}} \in \{2, 4, 6\}$ (note that the mean timer value is the same as in the homogeneous case), and we compare the time T_{hom} to steady state for the WTM with homogeneous timers and the mean time $\langle T_{\text{het}} \rangle$

²We use $\phi = 0.1$ in the small example networks in Fig. 3.2 for consistency with the threshold of nodes in large networks in Section 3.3, where we use the characteristics of the WTM with timers on small example networks to explain some behaviour of the WTM with timers on large networks. In the example networks presented in Fig. 3.2, a node changes its state once one of its neighbours adopts, which may seem to be identical to a simple contagion model where a node can change its state once one of its neighbours adopts. However, in large networks in which nodes have threshold $\phi = 0.1$ and varying degrees, a node may not adopt by the adoption of only one neighbour if its degree is larger than 10, and this property makes the WTM different from a simple contagion model.

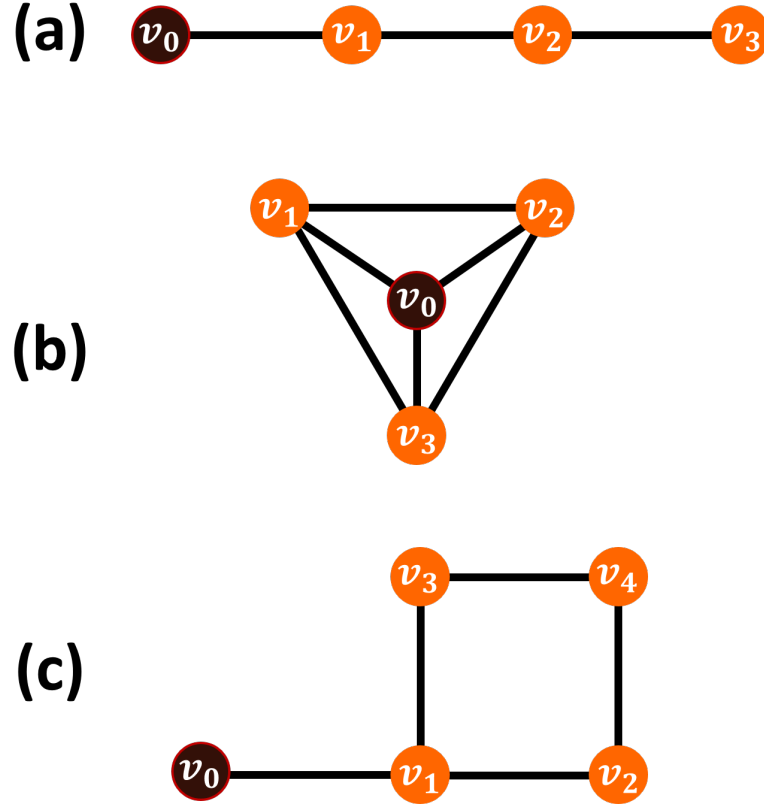


Figure 3.2: Small examples to illustrate the effect of incorporating timers in the WTM. The dark brown node v_0 is the seed node with state $s = 1$ at $t = 0$, and the orange nodes are in state $s = 0$ at $t = 0$. The thresholds are homogeneous, with $\phi = 0.1$ for each node.

to steady state for all possible timer configurations of the WTM with heterogeneous timers. In this example, the time to steady state is simply a function of the sum of timers of all nodes because adoptions spread from each node to its neighbour on the right. Therefore, for a given mean timer value, the time to steady state is the same regardless of whether the timer values are distributed homogeneously or heterogeneously. That is, the ratio $\langle T_{\text{het}} \rangle / T_{\text{hom}} = 1$. Note that neither the adoption order of the nodes nor the adoption path (v_0, v_1, v_2, v_3) can change even if the timers are distributed heterogeneously in the 1D lattice. Adoption always starts from the left end, and it spreads to the right one node at a time, regardless of how the timers are distributed.

Using the same homogeneous threshold value $\phi = 0.1$ as above, let us now consider what happens in a 4-clique. In a 4-clique, all nodes are adjacent to each other, so v_1 , v_2 , and v_3 are triggered simultaneously by the seed node v_0 at $t = 0$. Therefore, with

Table 3.1: Comparison between the time it takes for all nodes to adopt a behaviour (i.e., for a network to reach the fully-adopted state) for the small networks in Fig. 3.2 for the WTM with homogeneous timers $\tau_{\text{hom}} = 4$ and heterogeneous timers $\tau_{\text{het}} \in \{2, 4, 6\}$ if the number of non-seed nodes (i.e., the orange nodes in Fig. 3.2) is 3, and $\tau_{\text{het}} \in \{1, 3, 5, 7\}$ if the number of non-seed nodes is 4. The mean value of the heterogeneous timers is $\langle \tau \rangle = 4 = \tau_{\text{hom}}$. We calculate the mean time $\langle T_{\text{het}} \rangle$ to steady state for heterogeneous timers by averaging over all possible configurations of timers with the given set of heterogeneous timers. In comparing homogeneous and heterogeneous timers in these networks, we also indicate if the adoption orders of nodes and/or adoption paths can change (i.e., if they can be different in the two scenarios) from what occurs in the WTM without timers.

	$T_{\text{WTM}}^{\text{a}}$	$T_{\text{hom}}^{\text{b}}$	$\langle T_{\text{het}} \rangle^{\text{c}}$	$\frac{\langle T_{\text{het}} \rangle}{T_{\text{hom}}}$	Change of adoption order	Change of adoption paths
(a)	3	15	15	1	✗	✗
(b)	1	5	7	1.4	✓	✗
(c)	3	15	13.33	0.91	✓	✓

^a Time to steady state in the original WTM (i.e., without timers)

^b Mean time to reach the fully-adopted state when the timers are homogeneous

^c Mean time to reach the fully-adopted state averaged over all possible configurations of heterogeneous timers

timers, the time T to steady state is

$$T = 1 + \max\{\tau_{v_1}, \tau_{v_2}, \tau_{v_3}\}. \quad (3.3)$$

We need to add 1 because it takes 1 synchronous time step to start a countdown after a timer is triggered. In a 4-clique, the adoption paths (v_0, v_1) , (v_0, v_2) , and (v_0, v_3) do not change for any assignment of timers. Moreover, the spread of adoptions in different adoption paths are independent of each other. Therefore, the adoption path that includes the node with the largest timer terminates last, determining the time to steady state. Hence, T_{het} is always larger than T_{hom} for a 4-clique if the mean of the timer distribution is the same for the homogeneous and heterogeneous timers. In Table 3.1, we show an example with the same homogeneous and heterogeneous timers as in Fig. 3.2a, and we see that $\langle T_{\text{het}} \rangle / T_{\text{hom}} > 1$. We can generalise this to k -cliques with any value of k ; the time to steady state in a k -clique is determined by the largest timer, so $\langle T_{\text{het}} \rangle / T_{\text{hom}} > 1$ if the mean of the timer distribution is kept constant.

The example in Fig. 3.2c has a seed node adjacent to a square. The adoption of node v_1 triggers the timers of nodes v_2 and v_3 simultaneously, and whichever one

adopts earlier triggers the timer of node v_4 . The time T to steady state is

$$T = \begin{cases} \tau_{v_1} + \max\{2 + \tau_{v_3}, 3 + \tau_{v_2} + \tau_{v_4}\}, & \text{if } \tau_{v_2} \leq \tau_{v_3}, \\ \tau_{v_1} + \max\{2 + \tau_{v_2}, 3 + \tau_{v_3} + \tau_{v_4}\}, & \text{if } \tau_{v_2} > \tau_{v_3}. \end{cases}$$

We need to add 2 if adoption paths (v_0, v_1, v_3) or (v_0, v_1, v_2) determine the time to steady state, and we need to add 3 if adoption paths (v_0, v_1, v_2, v_4) or (v_0, v_1, v_3, v_4) determine the time to steady state. If the timers are homogeneous, then the adoption of node v_1 triggers the timers of both nodes v_2 and v_3 simultaneously, and the simultaneous adoption of v_2 and v_3 triggers the timer of v_4 , so the adoption paths are (v_0, v_1, v_2, v_4) and (v_0, v_1, v_3, v_4) . However, if the timers are heterogeneous and the timers of v_2 and v_3 are different, then there are changes in both the adoption order of the nodes and the adoption paths. Suppose, for example, that the timer of node v_2 is smaller than that of node v_3 . In this case, node v_2 adopts before v_3 , and it triggers the timer of v_4 ; therefore, the adoption paths are (v_0, v_1, v_2, v_4) and (v_0, v_1, v_3) , and the one that takes the longer time to terminate determines the time to steady state. In Table 3.1, we compare the time T_{hom} to steady state for a homogeneous timer $\tau_{\text{hom}} = 4$ and the mean time $\langle T_{\text{het}} \rangle$ to steady state for heterogeneous timers $\tau_{\text{het}} \in \{1, 3, 5, 7\}$, and we see that $\langle T_{\text{het}} \rangle < T_{\text{hom}}$, in contrast to the other examples in Fig. 3.2.

In fact, a network with a fixed seed node and fixed threshold assignments gives different adoption paths for different distributions of timers only if the network includes at least one cycle with 4 or more nodes. Consider a node v_i that is triggered by the adoption of node v_j in adoption path X for one timer distribution and by the adoption of node v_k (with $v_k \neq v_j$) in adoption path Y for another timer distribution. All adoption paths — regardless of the timer distribution — must share the same root (the seed node), because all adoptions spread from the seed. Therefore, adoption paths X and Y must share at least two nodes: v_i and the seed node. Therefore, the network has a cycle that includes the seed node, v_i , and the other nodes in X and Y . Therefore, a network that has different adoption paths for different distributions of timers must have a cycle of length at least 4. We show in Appendix A that for infinitely large loop with N nodes, the mean time $\langle T_{\text{het}} \rangle$ to steady state for heterogeneous timers is smaller than or equal to T_{hom} if N is even, and $\langle T_{\text{het}} \rangle$ is larger than or equal to T_{hom} if N is odd.

Because adoption paths can be different for heterogeneous timers than for a homogeneous timer, if the thresholds of all nodes in a network are sufficiently small so that any node adopts once at least one of its neighbours has adopted, an adoption path with nodes that have small timers tends to be long, and vice versa. Suppose

that a node v_i adopts if any one of its neighbours $v_j \in \Gamma(v_i)$ adopts, and suppose further that each neighbour belongs to a different adoption path of the same length. The time t''_{v_i} at which node v_i 's timer is triggered is then

$$t''_{v_i} = \min_{v_j \in \Gamma(v_i)} (t'_{v_j} + \tau_{v_j}) + 1, \quad (3.4)$$

where t'_{v_j} is the time that the timer of v_j is triggered (see Eq. (3.1)). Node v_i thereby becomes part of the adoption path that has the smallest sum of node timers, and this adoption path becomes longer than the other ones. Thus, in networks with cycles of length at least 4, adoption paths with small mean timers tend to be long, and the adoption paths with large mean timers tend to be short. For example, in Fig. 3.2c, the node with a smaller timer (v_2 in this case) is part of the longer adoption path, and the node with the larger timer (v_3 in this case) is part of the shorter adoption path. We will use the relationship between the length of an adoption path and the timer values of its nodes in Section 3.3.2.2 when we investigate long adoption paths with small mean timer values in large random networks. To compel nodes with small timers to be part of long adoption paths, it is important for their thresholds to be sufficiently small to adopt once at least one of their neighbours adopts. Otherwise, nodes with large timers tend to be part of longer adoption paths. In Fig. 3.3, we illustrate an example network in which the last node to adopt (v_2 in this case) has a sufficiently large threshold to adopt after both nodes v_1 and v_3 adopt. Therefore, when the timers of v_1 and v_3 are different, the node that adopts later triggers the adoption of node v_2 and becomes part of the longer adoption path. In our example in Fig. 3.3, v_2 has a larger timer than v_3 , so v_2 yields a longer adoption path [given by (v_0, v_1, v_2)] than v_3 . Accordingly, if we abandon the premise that a node adopts if one neighbour adopts, then nodes with timers larger than the others can become part of long adoption paths.

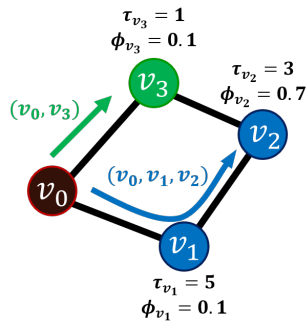


Figure 3.3: Example of adoption paths when there is a node with a large adoption threshold.

3.3 Timers on large random networks

We now incorporate both homogeneous and heterogeneous timers into a spreading process on large networks. Specifically, we examine the WTM with a timer on the largest connected component (LCC) of random networks with $N = 10,000$ nodes (i.e., of “size” 10,000). For many of our random networks, we use a homogeneous threshold ϕ such that a node whose degree is equal to the graph’s mean degree needs only a single adopted neighbour to be triggered (as in the small networks in Fig. 3.2). Because all nodes have the same threshold, we choose a seed adopted node uniformly at random at $t = 0$ to initiate the spread of adoptions. In our simulations, we report sample means of many realizations to give an idea of ensemble expectations in the $N \rightarrow \infty$ limit.

3.3.1 Timer distribution and dynamics of the WTM with a timer

We first consider the WTM with a timer on the LCC of $G(N, p)$ Erdős–Rényi (ER) networks with $N = 10,000$ nodes and edge probability $p = 0.0006$ (and thus an expected mean degree of $z = 6$), and we compare the adoption curves — the progress of the adopted fraction $\rho(t)$ of nodes at time t — between homogeneous and heterogeneous timers. Note that the results we illustrate in this section also arise when we use other values of p and ϕ .

In Fig. 3.4a, the pink curve shows the adoption process of the WTM without a timer, and the green curve shows the adoption process of the WTM with a homogeneous timer $\tau = 4$. As the figure shows, at each point that adoption occurs, the green curve is simply delayed for 4 time steps from the pink curve. The green adoption curve thus has a stair-like shape. If we employ asynchronous updating rather than synchronous updating (so that we choose some number of nodes uniformly at random at each time step to update their states [113]), the change in dynamics is not simply a delay for each adoption, as the randomness in node choice changes the adoption order³. See [39] for a discussion of discrete versus continuous dynamics in contagion models on networks.

The orange curve in Fig. 3.4a is the adoption curve of the WTM with heterogeneous timers on ER networks of size $N = 10,000$ in which we select a timer

³If we add a timer to any stochastic process on a network (e.g., with a non-deterministic update rule), a homogeneous timer does not in general simply delay the adoption process, because the adoption order of adoption can change due to the randomness.

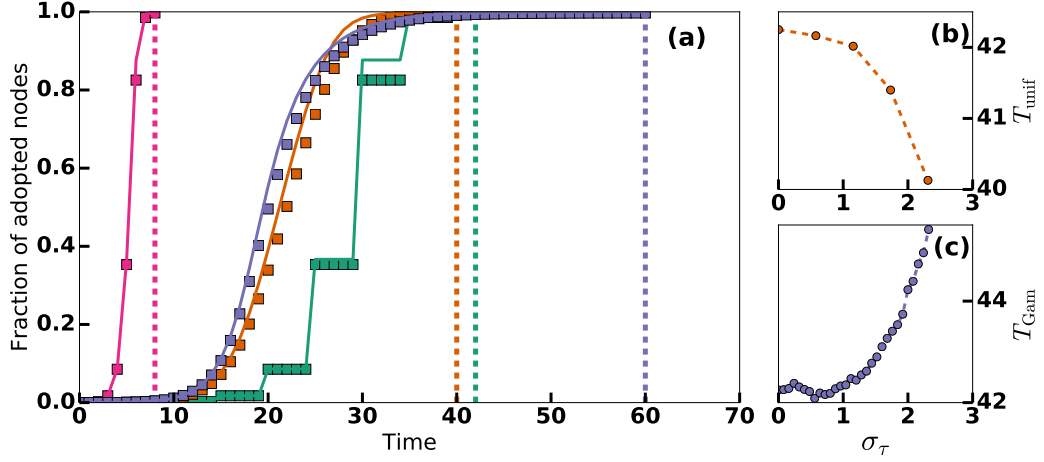


Figure 3.4: (a) Adoption curves of the WTM with timers. The pink curve is without timers (i.e., the original WTM), the green curve is when the timers are homogeneous with $\tau = 4$, the orange curve is with heterogeneous timers τ that are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and the blue curve is for heterogeneous timers given by integers that we round down from a random variable that follows the Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$. Squares are the results of numerical simulations, the dashed lines mark the times at which the adoption process of the corresponding colour reaches a steady state, and the solid curves are results from an analytical approximation (see Chapter 4). (b,c) Change of the times to steady state (T_{unif} and T_{Gam} , respectively) by increasing the standard deviation σ when timers are distributed (b) uniformly at random and (c) approximately according to a Gamma distribution. We obtain our numerical results by averaging over 1,000 realizations of the WTM with timers on $G(N, p)$ ER networks with $N = 10,000$ nodes and edge probability $p = 0.0006$. To isolate the effects of incorporating timers, we use the same 1,000 ER networks for each of the 4 different cases.

$\tau \in \{0, 1, \dots, 8\}$ uniformly at random. The mean μ_τ of the timers is 4, which is the same as for the homogeneous timer in the figure. The obvious difference from the heterogeneous timers is that the adoption curve is now much smoother (compare the orange and green curves). This arises amidst the change in adoption order from the heterogeneous timers; nodes that adopt simultaneously when timers are homogeneous now adopt at different times, and there are now fewer nodes that adopt simultaneously. (Note that some nodes that adopted at different times with a homogeneous timer now adopt simultaneously.) We also see that the time T_{unif} to steady state (orange dashed line) when the heterogeneous timers are distributed uniformly at random is earlier than the time T_{hom} to steady state (green dashed line) for homogeneous timers. In ER networks, we thus see that uniformly random heterogeneous timers ac-

Table 3.2: Time to reach certain fractions of adopted nodes for the WTM on ER networks with homogeneous timers, heterogeneous timers distributed uniformly at random, and timers determined using a Gamma distribution and then rounded down to an integer.

ρ^{*a}	t_{hom}^b	t_{unif}^c	$\frac{t_{\text{unif}}}{t_{\text{hom}}}$	t_{Gam}^d	$\frac{t_{\text{Gam}}}{t_{\text{hom}}}$
0.5	29.31	21.97	0.75	20.05	0.68
0.6	29.52	23.18	0.79	21.21	0.72
0.7	29.74	24.48	0.82	22.59	0.76
0.8	29.95	25.97	0.87	24.43	0.82
0.9	34.48	28.10	0.81	27.56	0.80
ρ_{st}^e	42.41	40.21	0.95	60.44	1.43

^a Fraction of adopted nodes.

^b Time to reach ρ^* when timers are homogeneous.

^c Time to reach ρ^* when the timers are distributed uniformly at random.

^d Time to reach ρ^* when the timers are determined using a Gamma distribution and then rounded down to an integer.

^e Fraction of adopted nodes at steady state.

celerate the adoption process compared to the case of homogeneous timers. Increased heterogeneity in the distribution of the timers accelerates the adoption process even further. As one can observe from Fig. 3.4b, increasing the standard deviation σ_τ of the uniformly randomly distributed timers accelerates the time to steady state for ER networks.

Importantly, it is not true that heterogeneously-distributed timers necessarily accelerate an adoption process compared to homogeneous timers. For example, the blue curve in Fig. 3.4 shows the adoption process of the WTM with timers given by integers that we determine by rounding down from numbers drawn uniformly at random from a Gamma distribution [142] with the same mean $\mu_\tau = 4$ as the uniformly randomly distributed timers and with standard deviation $\sigma_\tau = 4$. The blue dashed line marks the time T_{Gam} at which the blue adoption curve reaches a steady state, and we observe that it is located to the right of the green dashed line T_{hom} . In Fig. 3.4c, we show that the increase of standard deviation of the Gamma distribution decelerates the time to steady state on ER networks, in stark contrast to our observations in Fig. 3.4b for timers that are distributed uniformly at random.

Although the time to steady state for heterogeneous timers can become either larger or smaller than T_{hom} , depending on the timer distribution, we observe that the majority of nodes adopt noticeably earlier when the timers are distributed either

uniformly at random or using a Gamma distribution (and then rounded down to an integer)⁴ than they do for homogeneous timers. In Table 3.2, we compare the time t when the adopted fraction $\rho(t)$ reaches at least a certain fraction ρ^* of nodes in networks when incorporating a homogeneous timer (t_{hom}), heterogeneous timers distributed uniformly at random (t_{unif}), and heterogeneous timers determined using a Gamma distribution (t_{Gam}). We update nodes synchronously, so we do not in general have exactly the adoption fraction $\rho(t) = \rho^*$ at any time. In this case, we find the time T' at which the fraction $\rho(T')$ of adopted nodes first exceeds ρ^* . We then estimate the time $t_{\rho=\rho^*}$ at which the fraction of adopted nodes reaches ρ^* as

$$t_{\rho=\rho^*} = T' - 1 + \frac{\rho^* - \rho(T' - 1)}{\rho(T') - \rho(T' - 1)}. \quad (3.5)$$

As we show in Table 3.2, the fraction $\rho(t)$ of adopted nodes tends to evolve faster when incorporating either uniformly randomly distributed timers or Gamma-distributed timers than for homogeneous timers, although Gamma-distributed timers take a longer time to reach the steady-state adoption fraction ρ_{st} . This illustrates that a global cascade can occur earlier when one considers heterogeneous timers than for homogeneous timers. In Section 3.3.2, we investigate adoption paths and give evidence for how incorporating heterogeneous timers in the WTM can make the majority of nodes adopt earlier than when considering homogeneous timers.

3.3.2 Adoption paths in large networks

In Section 3.1, we argued that the WTM with a homogeneous timer τ_{hom} has exactly the same adoption paths as the adoption paths of the WTM without timers, because homogeneous timers do not change the adoption order of nodes but instead merely delays adoption times uniformly by τ_{hom} ; therefore, the time T_{hom} to achieve a steady state is delayed to $T_{\text{hom}} = T_{\text{WTM}}(1 + \tau_{\text{hom}})$, where T_{WTM} is the time to steady state when there are no timers. However, if the timers are distributed heterogeneously, the adoption order of nodes can change, so adoption paths can also change (depending on the network structure). Furthermore, the relationship between the time T_{het} to steady state for heterogeneous timers and T_{WTM} is more complicated than the relationship between T_{hom} and T_{WTM} . In Section 3.3.1, we observed from simulations on ER networks that the WTM with timers distributed uniformly at random and timers determined using a Gamma distribution yield earlier adoptions for the majority of

⁴Henceforth, we will usually state that these timers are determined using a Gamma distribution rather than always stating explicitly that we round down to an integer.

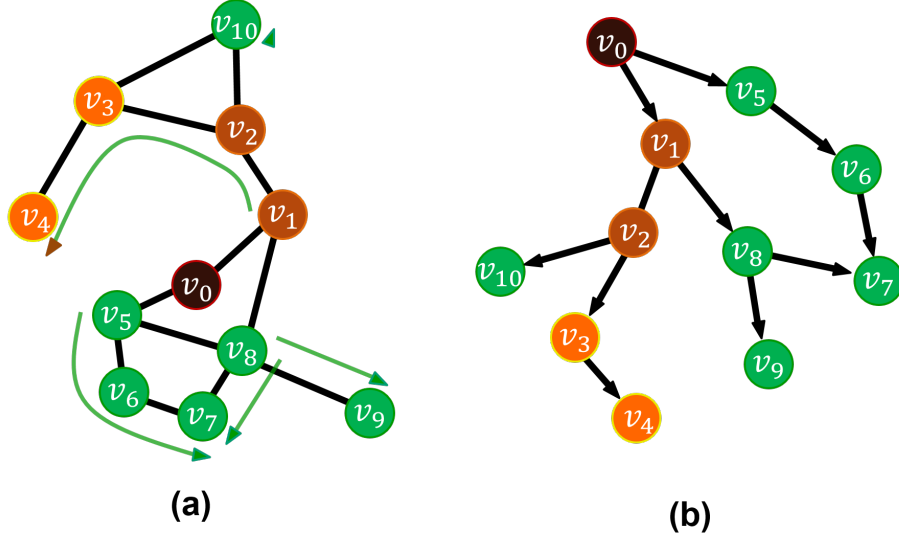


Figure 3.5: (a) An example network on which we run the WTM with timers. Curved arrows illustrate the spread of adoptions, and each curved arrow represents an adoption path. The brown curved arrow is a stem, and the green curved arrows are branches. (b) Graphical illustration of a dissemination tree, based on the spread of adoptions in panel (a). Node v_0 is a seed, which is the root of the stem (v_1, v_2, v_3, v_4). Nodes v_0, v_1 , and v_2 in the stem initiate the adoption in branches; they are the roots of the branches (v_5, v_6, v_7) , (v_8, v_7) , (v_8, v_9) , and (v_{10}) .

nodes than is the case for a homogeneous timer with the same mean (see Table 3.2). In this section, we explore this issue in depth by investigating adoption paths of the WTM with timers for both synthetic and real-world networks.

3.3.2.1 Stems and branches

As we discussed in Section 3.1, an adoption path is a sequence of directed edges in which each edge indicates the flow of adoption from a node at one end to a node at the other end. All adoption paths grow from the seed, which we recall is the node that initiates the spread of adoptions in a network. Therefore, the seed is the root of all adoption paths. Among the adoption paths, which have various lengths, we pick a longest adoption path at steady state. We do not include the root as part of the adoption path, which we call a *stem* of adoption spreading (or simply a “stem”). Nodes in a stem are called *stem nodes*. If there are two or more adoption paths that both have the largest length, we consider all of them to be stems. For the other adoption paths, we exclude the stem nodes; their remaining nodes are *branches* of adoption spreading (or simply “branches”). The main difference between stems and branches is that stems grow from a seed node, whereas branches grow from stems.

We give an example of a stem and its branches in Fig. 3.5. Suppose that we run the WTM with timers on the network in Fig. 3.5a with node v_0 as a seed and that we obtain five adoption paths: $(v_0, v_1, v_2, v_3, v_4)$, (v_0, v_5, v_6, v_7) , (v_0, v_1, v_8, v_7) , (v_0, v_1, v_8, v_9) , and (v_0, v_1, v_2, v_{10}) . Among the adoption paths, $(v_0, v_1, v_2, v_3, v_4)$ has the largest length, so the stem is (v_1, v_2, v_3, v_4) . The branches are (v_5, v_6, v_7) , (v_8, v_7) , (v_8, v_9) and (v_{10}) . Note that v_7 and v_8 each appear in two different adoption paths, because the adoption of v_8 triggers the timers of v_7 and v_9 , and the timer of v_7 is simultaneously triggered by v_6 and v_8 .

With the adoption paths, we can construct a tree (see Fig. 3.5b), which demonstrates how adoptions spread in the original network in Fig. 3.5a. We use the term *dissemination tree* for a network composed of adoption paths.⁵ Because an adoption path is directed, a dissemination “tree” does not have a path back to the root and is thus technically a directed acyclic graph (DAG) [105]. However, as one can see in Fig. 3.5b, the underlying undirected graph of a dissemination tree can have a cycle of length at least 4; in Fig. 3.5b, this cycle is $(v_0, v_1, v_8, v_7, v_6, v_5, v_0)$. However, a dissemination tree’s underlying undirected graph cannot have triangular clustering (i.e., 3-node cycles). For a 3-cycle to exist in a dissemination tree’s underlying undirected graph, a node’s timer needs to be triggered by the simultaneous adoption of its two neighbours. However, the two neighbours cannot be adjacent to each other in the dissemination tree, because any two adjacent nodes in a dissemination tree cannot adopt simultaneously; a pair of nodes are neighbours in a dissemination tree if and only if one triggers the timer of the other. Therefore, there cannot exist a 3-cycle in a dissemination tree.

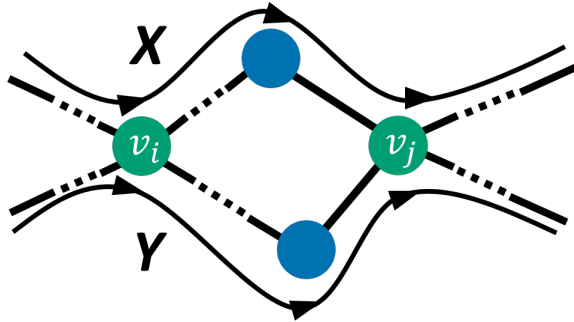


Figure 3.6: Illustration of a cycle in a dissemination tree.

⁵Dissemination trees have been studied previously, though using other terminology [57, 103, 128, 132]. Reference [56] used the term “diffusion tree”, but we use the word “dissemination” instead of “diffusion” to distinguish diffusion from other types of spreading dynamics (e.g., non-conservative contagion dynamics).

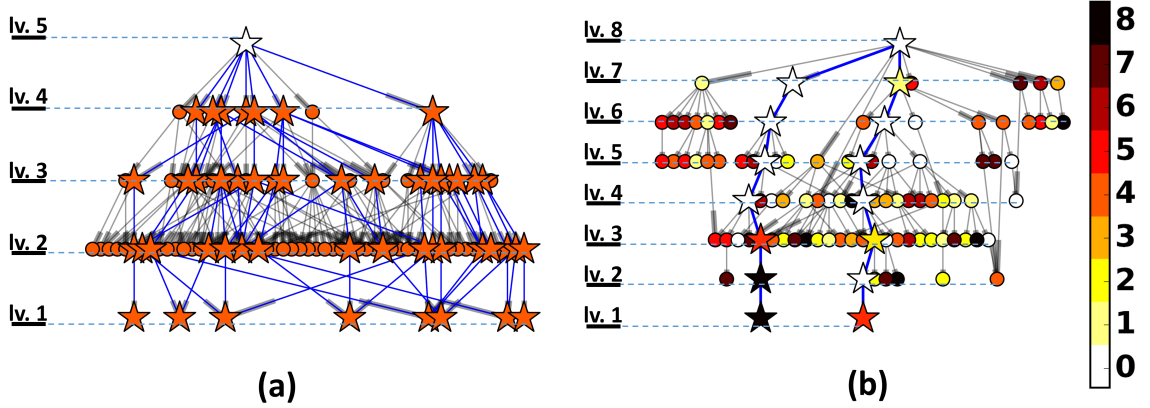


Figure 3.7: Graphical representation of a dissemination tree for the WTM with (a) homogeneous timers and (b) heterogeneous timers. The edges are directed from higher levels [lv. 5 in (a) and lv. 8 in (b)] to lower levels [lv. 1 in both (a) and (b)]. We colour edges in blue if they are part of a stem and in gray if they are part of a branch. The solid blue lines indicate edges that are part of stems, and the solid gray lines indicate edges that are part of branches. We colour the nodes according to their time values, which range from 0 (white) to 8 (black). We use stars for nodes in stems and disks for nodes in branches. The number of levels in the dissemination tree with homogeneous timers is 5, and the number of levels with heterogeneous timers is 8.

In contrast, a dissemination tree can have cycles with 4 or more nodes when two nodes have a common predecessor in each of their adoption paths. In particular, for the WTM with heterogeneous timers, any k -cycle with $k \geq 4$ can exist; for a homogeneous timer, k must be even. Assume that there is a cycle in a dissemination tree (see Fig. 3.6). For a dissemination tree, having a cycle implies that there exist two different adoption paths (X and Y in Fig. 3.6) with at least two common nodes (v_i and v_j) in their adoption paths. Suppose that v_i adopts before v_j . If the timers are homogeneous, the adoption spreads from v_i to v_j at the same rate in X and Y . Therefore, the cycle sizes for a homogeneous timer can only be even. However, for heterogeneous timers, the rates of adoption spreading in X and Y can be different. Therefore, any k -cycle with $k \geq 4$ can exist for the WTM with heterogeneous timers.

3.3.2.2 Dissemination trees for synthetic networks

In Fig. 3.7, we show graphical representations of dissemination trees that we obtain from running the WTM with timers on an ER network of size $N = 100$, mean degree $z = 6$, and a homogeneous threshold of $\phi = 0.1$ for each node. We give examples with both homogeneous and heterogeneous timers, and we examine adoption paths and combine them to create a dissemination tree in each case. We place the

root of the stem (i.e., the seed node) at the top of the tree, and we place the rest of the nodes in the subsequent levels based on their distance from the root. In Fig. 3.7a, we show a dissemination tree when timers are distributed homogeneously with $\tau = 4$. In Fig. 3.7b, we show a dissemination tree when the timers are distributed heterogeneously (and, specifically, drawn uniformly at random from $\{0, 1, \dots, 8\}$). We colour the nodes based on their timer values, which range from 0 (white) to 8 (black). We use stars for nodes in stems and disks for nodes in branches. We colour the edges of stems in blue, and we use gray for edges in branches. When the timers are homogeneous, there are more stems (many different ones with the same maximal length), whose length is smaller than in the example with heterogeneous timers. When the timers are heterogeneous, we also observe that the timers of nodes in a stem tend to be small (lighter colours). As we discussed in Section 3.1.2, the timer of a node in an adoption path gets triggered instantaneously by the adoption of its neighbour that is adjacent via an in-edge (i.e., an edge that directs the node from its neighbours). Thus, the time to terminate an adoption path is determined by the sum of timers of nodes in the adoption path [see Eq. (3.1)]. Consequently, if the mean of the node timers in an adoption path is small, an adoption tends to be transmitted through the path in a short amount of time. The mean timer value of the nodes in an adoption path thus helps determine how fast an adoption spreads in an adoption path. Therefore, the mean timer value of stem nodes being smaller than the mean value of branch-node timers suggests that adoption should tend to spread at a faster rate along stems than along the branches. In Table 3.3, we show results for several families of random networks, and we observe qualitatively similar phenomena as in Fig. 3.7.

In Table 3.3, we compare the characteristics of dissemination trees for the WTM with a homogeneous timer $\tau_{\text{hom}} = 4$ versus dissemination trees for the WTM with heterogeneous timers on different random-graph models. For the WTM with heterogeneous timers, we first consider timers drawn uniformly at random from $\{0, 1, \dots, 8\}$, and we then consider timers drawn from a Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and then rounded down to an integer. We show results for 13 diagnostics in Table 3.3: the time T to steady state, the time $t_{0.9}$ that it takes for at least 90% of the nodes to adopt, the mean number r of adopted nodes in the networks, the mean number a of adoption paths, the mean μ_l and the standard deviation σ_l of adoption-path lengths, *structural virality* [56] v (which is defined as the mean shortest-path length in a dissemination tree), and several stem-specific and branch-specific quantities. These latter quantities are the mean percentage f_{as}

of stems and the mean percentage f_{ab} of branches among adoption paths, the mean percentage f_{r_s} of stem nodes and the mean percentage f_{r_b} of branch nodes among adopted nodes, and the mean stem length l_s and mean branch length l_b .

The network models that we use in Table 3.3 are a configuration model [41] and generalised configuration models that include cliques [49, 66]. We construct configuration-model networks by specifying a degree distribution $\{P_k\}$ and then connecting stubs (i.e., ends of edges) uniformly at random. To construct networks using a generalised configuration model, we embellish the above configuration model by incorporating cliques [49, 66]. We start with a configuration-model network with degree distribution $\{P_k\}$, but there is also a joint distribution $\gamma(k, c)$ that specifies the probability that a node chosen uniformly at random has degree k and is in a clique of c nodes (i.e., a c -clique). Note that $\gamma(k, c) = 0$ for $k < c - 1$, as a node with degree k can only be a member of a c -clique if its degree is large enough to link to all $c - 1$ neighbours in the clique.

For all networks in Table 3.3, we use configuration-model networks with $N = 10,000$ nodes and degrees drawn from the Poisson distribution $P_k = z^k e^{-z} / k!$ with mean $z = 6$. In the networks, each node has a homogeneous adoption threshold of $\phi = 0.1$, so that a node with the mean degree adopts once one of its neighbours adopts. We use “Config” to denote a standard configuration model; “Congen-3” to denote a generalised configuration model with 3-cliques and with joint distribution $\gamma(k, c) = [(1 - \alpha)\delta_{c,1} + \alpha\delta_{c,3}] P_k$ for $k \geq 3$ (where the parameter α determines the edge-clique ratio for a node of degree $k \geq 3$); and “Congen-4” to denote a generalised configuration model with 4-cliques and with joint distribution $\gamma(k, c) = [(1 - \beta)\delta_{c,1} + \beta\delta_{c,4}] P_k$ for $k \geq 4$ (where β determines the edge-clique ratio for a node of degree $k \geq 4$).

As we discussed in Section 3.3, the time T to steady state for the WTM with heterogeneous timers can be either shorter (e.g., for timers distributed uniformly at random) or longer (e.g., for Gamma-distributed timers) than for a homogeneous timer, but both choices of heterogeneous timer distributions have a smaller value than with a homogeneous timer for the time $t_{0.9}$ for at least 90% of the nodes to adopt. Additionally, for both distributions of heterogeneous timers, we observe a smaller mean number n of adoption paths than for a homogeneous timer. This, in turn, results in a shorter mean adoption path length μ_l for homogeneous timers than for heterogeneous ones, because for a fixed network size N , having a larger number of adoption paths leads to shorter mean adoption path lengths. Note that the number n of adoption paths of a network can become larger than the number r of adopted

nodes in a network, because a node can be part of different adoption paths if the node triggers the timers of multiple nodes at the same time.

If timers are homogeneous, adoptions spread at the same rate for every adoption path, and adoption paths that terminate at the same time have the same lengths. However, for heterogeneous timers, adoption paths that terminate at the same time can have different lengths, because adoptions can spread at different rates. Therefore, the lengths of adoption paths can be more diverse when using heterogeneous timers than for homogeneous ones. As we see in Table 3.3, the standard deviation σ_l of adoption path lengths is larger for the WTM with heterogeneous timers than with homogeneous timers. That is, $\sigma_{l,\text{unif}} > \sigma_{l,\text{hom}}$ and $\sigma_{l,\text{Gam}} > \sigma_{l,\text{hom}}$ in our simulations.

Table 3.3: Comparison between the characteristics of stems and branches from running the WTM with timers on configuration-model networks (“Config”), generalised configuration-model networks with 3-cliques (“Congen-3”), and generalised configuration-model networks with 4-cliques (“Congen-4”). For all networks in this table, we use configuration-model networks with $N = 10,000$ nodes and degrees drawn from a Poisson distribution $P_k = z^k e^{-z}/k!$ with mean $z = 6$. All nodes have a homogeneous threshold of $\phi = 0.1$, so a node with the mean degree adopts once one of its neighbours adopts. For the generalised configuration models, we consider different values of the edge-clique ratios α and β , where α determines the edge-clique ratio for a node of degree $k \geq 3$ in Congen-3 and β determines the edge-clique ratio for a node of degree $k \geq 4$ in Congen-4. The notation x_{hom} indicates that we calculate the quantity x for homogeneous timers, the notation x_{unif} indicates that we calculate the quantity x for timers that are distributed uniformly at random, and the notation x_{Gam} indicates that we calculate the quantity x when the timers are determined using a Gamma distribution. The quantity T is the time to steady state, and t_{ρ^*} is the time that it takes for adopted fraction ρ^* to reach a certain fraction of adopted nodes. (In this table, we use $\rho^* = 0.9$.) The quantity μ_l is the mean adoption path length, σ_l is the standard deviation of the lengths of adoption paths, v is the *structural virality* (which is defined as the mean shortest path length in a dissemination tree), a is the mean number of adoption paths, f_{as} is the mean percentage of stems among adoption paths, f_{ab} is the mean percentage of branches among adoption paths, r is the mean number of adopted nodes, f_{rs} is the mean percentage of stem nodes, f_{rb} is the mean percentage of branch nodes, l_s is the mean stem length, and l_b is the mean branch length. To compare the effects of different timer distributions, we augment the WTM with the different timer distributions on the same network with the same seed node and the same adoption-threshold distribution for each realization. Each reported value is a mean over 100 simulations of the WTM with timers on networks generated independently for each simulation.

Config	Homogeneous timer												
	T_{hom}	$t_{0.9,\text{hom}}$	r_{hom}	a_{hom}	$\mu_{l,\text{hom}}$	$\sigma_{l,\text{hom}}$	v_{hom}	f_{rs} (%)	f_{rb} (%)	f_{as} (%)	f_{ab} (%)	$l_{s,\text{hom}}$	$l_{b,\text{hom}}$
Congen-3	42.76	32.97	9999.43	17505.69	7.72	0.36	7.15	4.44	95.56	1.07	98.93	9.55	4.13
($\alpha = 0.5$)	44.06	34.30	9998.99	16754.55	7.91	0.37	7.31	4.70	95.30	1.25	98.75	9.81	4.30
Congen-3	46.32	35.43	9993.95	15770.20	8.16	0.37	7.54	3.02	96.98	0.62	99.38	10.26	4.57
($\alpha = 1.0$)	45.41	35.03	9987.32	17316.51	8.05	0.41	7.48	3.46	96.54	0.88	99.12	10.08	4.56
Congen-4													
($\beta = 0.5$)													

Table 3.3: (Continued from the previous page.)

Congen-4 ($\beta = 1.0$)	52.17	38.78	9996.23	17250.30	8.72	0.43	8.17	1.66	98.34	0.31	99.69	11.43	5.57
Uniformly random distribution of timers													
	T_{unif}	$t_{0.9,\text{unif}}$	r_{unif}	a_{unif}	$\mu_{l,\text{unif}}$	$\sigma_{l,\text{unif}}$	v_{unif}	$f_{r_s}(\%)$	$f_{r_b}(\%)$	$f_{a_s}(\%)$	$f_{a_b}(\%)$	$l_{s,\text{unif}}$	$l_{b,\text{unif}}$
Config	40.39	27.26	9999.43	7935.59	9.02	0.46	10.52	0.34	99.66	0.05	99.95	15.90	6.57
Congen-3 ($\alpha = 0.5$)	41.76	28.34	9998.99	7763.81	9.19	0.49	10.79	0.35	99.65	0.35	99.65	16.13	6.75
Congen-3 ($\alpha = 1.0$)	44.10	29.77	9993.95	7511.16	9.38	0.47	11.16	0.35	99.65	0.06	99.94	16.53	6.92
Congen-4 ($\beta = 0.5$)	42.81	27.13	9987.32	7798.33	9.29	0.54	11.03	0.35	99.65	0.06	99.94	16.30	6.84
Congen-4 ($\beta = 1.0$)	49.60	32.81	9996.23	7500.79	9.87	0.54	12.05	0.34	99.66	0.05	99.95	17.35	7.36
Gamma distribution (and then rounding down to an integer) of timers													
	T_{Gam}	$t_{0.9,\text{Gam}}$	r_{Gam}	a_{Gam}	$\mu_{l,\text{Gam}}$	$\sigma_{l,\text{Gam}}$	v_{Gam}	$f_{r_s}(\%)$	$f_{r_b}(\%)$	$f_{a_s}(\%)$	$f_{a_b}(\%)$	$l_{s,\text{Gam}}$	$l_{b,\text{Gam}}$
Config	61.19	29.89	9999.43	7786.92	8.51	0.44	10.45	0.50	99.50	0.09	99.91	13.15	6.13
Congen-3 ($\alpha = 0.5$)	61.92	30.93	9998.99	7630.97	8.72	0.47	10.75	0.48	99.52	0.08	99.92	13.52	6.32
Congen-3 ($\alpha = 1.0$)	63.67	32.23	9993.95	7400.31	8.98	0.47	11.14	0.44	99.56	0.07	99.93	14.09	6.60
Congen-4 ($\beta = 0.5$)	62.69	31.75	9987.32	7701.32	8.88	0.52	11.00	0.45	99.55	0.08	99.92	13.85	6.46
Congen-4 ($\beta = 1.0$)	66.99	35.23	9996.23	7493.46	9.60	0.53	12.08	0.41	99.59	0.07	99.93	15.26	7.12

Goel et al. [56] studied how viral contents that spread from peer to peer in large networks have different spreading patterns from content that does not go viral, and they introduced the idea of structural virality to try quantify the virality of content from its spreading pattern. Structural virality is one quantity that has been used to try to forecast the eventual size of a cascade, so it is useful for studies of information dissemination and social influence (see [107] for a comparison of different ways — including ones that are based on cascades — to measure social influence in a network). They defined structural virality v as the mean shortest-path length in a dissemination tree, and a larger v indicates that the mean distance between any two nodes chosen uniformly at random from a dissemination tree is larger. If a meme goes viral, it is reasonable that the mean distance between randomly-chosen nodes should be larger than for memes that do not become viral. We show structural virality of dissemination trees in Table 3.3, and the results provide evidence that v is larger for the WTM with heterogeneous timers than with homogeneous ones.

For the WTM with each of the distributions of timers that we examine, the stem-specific quantities f_{r_s} and f_{a_s} are smaller than the branch-specific quantities f_{r_b} and f_{a_b} . In particular, for the two types of heterogeneous timers, both f_{r_s} and f_{a_s} in Table 3.3 are less than 1, and they are smaller than the corresponding quantities when we use a homogeneous timer. For a homogeneous timer, all adoption paths grow at the same rate, so the adoption path (or paths, if there is a tie) that grows for the longest time is the longest adoption path at steady state. Therefore, in this situation, all adoption paths that grow until steady state are stems. However, adoption paths grow at different rates for heterogeneous timers. Therefore, even if a stem terminates when a simulation reaches a steady state, not all adoption paths that grow until a steady state need to be stems. (A stem can terminate before a branch if the sum of timers of the stem nodes is smaller than the sum of branch-node timers.) We observe in our simulations that the mean percentage f_{a_s} of stems is smaller for heterogeneous timers than for a homogeneous timer. We speculate that for most network structures the number of stems should be smaller for the WTM with heterogeneous timers than for the WTM with a homogeneous timer.

Table 3.4: Comparison between the mean timer of stems and branches for the WTM model with a homogeneous timer, uniformly-randomly-distributed timers, and timers determined using a Gamma distribution. The networks that we use are a configuration model (Config), a generalised configuration model with 3-cliques (Congen-3), and a generalised configuration model with 4-cliques (Congen-4). For all networks, we use configuration-model networks with $N = 10,000$ nodes and degrees drawn from a Poisson distribution $P_k = z^k e^{-z}/k!$ with mean $z = 6$. Additionally, each node has a homogeneous threshold of $\phi = 0.1$, so a node with the mean degree adopts once one of its neighbours adopts. For the generalised configuration models, we consider different values of the edge-clique ratios α and β , where α determines the edge-clique ratio for a node of degree $k \geq 3$ in Congen-3 and β determines the edge-clique ratio for a node of degree $k \geq 4$ in Congen-4. The notation x_{hom} indicates that we calculate the quantity x for homogeneous timers, the notation x_{unif} indicates that we calculate the quantity x for timers that are distributed uniformly at random, and the notation x_{Gam} indicates that we calculate the quantity x for timers determined using a Gamma distribution. The quantity τ_s is the mean stem-node timer, τ_b is the mean branch-node timer, and $\langle \tau \rangle$ is the mean of all timers.

	Homogeneous timer		Uniformly random timers		Gamma-distributed timers	
	$\frac{\tau_s, \text{hom}}{\langle \tau \rangle}$	$\frac{\tau_b, \text{hom}}{\langle \tau \rangle}$	$\frac{\tau_s, \text{unif}}{\langle \tau \rangle}$	$\frac{\tau_b, \text{unif}}{\langle \tau \rangle}$	$\frac{\tau_s, \text{Gam}}{\langle \tau \rangle}$	$\frac{\tau_b, \text{Gam}}{\langle \tau \rangle}$
Config	1.0	1.0	0.29	1.00	0.42	1.01
Congen-3 ($\alpha = 0.5$)	1.0	1.0	0.30	1.00	0.40	1.01
Congen-3 ($\alpha = 1.0$)	1.0	1.0	0.33	1.00	0.40	1.00
Congen-4 ($\beta = 0.5$)	1.0	1.0	0.32	1.00	0.39	1.00
Congen-4 ($\beta = 1.0$)	1.0	1.0	0.35	1.00	0.44	1.00

To better understand the difference between the stems and branches of the networks in Table 3.3, we show calculations of (1) the ratio $\tau_s/\langle\tau\rangle$ of the mean τ_s of timer values of nodes in stems to the mean $\langle\tau\rangle$ of timer values of all nodes, and (2) the ratio $\tau_b/\langle\tau\rangle$ of the mean τ_b of timer values of nodes in branches to $\langle\tau\rangle$ in Table 3.4. For the WTM with a homogeneous timer, both $\frac{\tau_{s,\text{hom}}}{\langle\tau\rangle} = 1$ and $\frac{\tau_{b,\text{hom}}}{\langle\tau\rangle} = 1$, so stems and branches grow at the same rate. When the timers are distributed uniformly at random, stems tend to grow at faster rates than the mean rate because $\frac{\tau_{s,\text{unif}}}{\langle\tau\rangle} < 1$, whereas branches grow at slower rates than the mean rate because $\frac{\tau_{b,\text{unif}}}{\langle\tau\rangle}$ is (slightly) larger than 1. Similarly, when the timers are determined using a Gamma distribution, $\frac{\tau_{s,\text{Gam}}}{\langle\tau\rangle} < 1$ and $\frac{\tau_{b,\text{Gam}}}{\langle\tau\rangle}$ is slightly larger than 1. The reason that there is only slight difference between τ_b and $\langle\tau\rangle$ for the heterogeneous timers that we study is that the percentages r_b of branch nodes for heterogeneous timers are larger than 99% of all nodes in networks, as we see in Table 3.3.

To better understand why the mean τ_b is larger than $\langle\tau\rangle$ and τ_s is smaller than $\langle\tau\rangle$, it is useful to revisit our discussion from the end of Section 3.2 about the relationship between the mean timer size for nodes in an adoption path and the length of the adoption path. We know that a network can have different adoption paths for different timer distributions. Consider a scenario in which all nodes in a given network are vulnerable nodes — nodes that adopt if even one of their neighbours adopts. The nodes with small timers tend to be part of long adoption paths, and the nodes with large timers tend to be part of short adoption paths. From Table 3.3, we know that the adoption paths for a homogeneous and heterogeneous timers are different for different distributions of timers, because the number a of adoption paths entails different values for different timer distributions. The nodes in the networks in Table 3.3 have a homogeneous adoption threshold of $\phi = 0.1$, so nodes with degree of 10 or less will adopt once a single neighbour adopts. For example, in the configuration-model networks, the expected mean of the degree distribution is $z = 6$, so there are a larger number of nodes with degree less than or equal to 10 than with degree larger than 10, and most of the nodes in these networks adopt if any of their neighbours adopt. Accordingly, long adoption paths of dissemination trees in Table 3.3 are more likely to consist of nodes with small timers. Thus, the ratio $\tau_s/\langle\tau\rangle$ of the mean timer value of stem nodes to the mean timer value $\langle\tau\rangle$ of all nodes is smaller than 1 for the WTM with heterogeneous timers, as we can see in Table 3.4. (Note that it is possible for a stem to consist of nodes with large timers depending on network structure; see Appendix B for more details.)

Recall that (1) rates of adoption spreading in stems are smaller for the WTM with heterogeneous timer distributions than with a homogeneous timer distribution and (2) the rates of adoption spreading in branches are only slightly larger for heterogeneous timer distributions than for a homogeneous timer distribution. Therefore, in our calculations, it seems that stems play a significant role in spreading adoptions to the majority of nodes faster for heterogeneous timers than for a homogeneous timer. To support our speculation, we conduct the following experiment on the dissemination trees in Tables 3.3 and 3.4. We know that each node in a dissemination tree has its own timer (i.e., the timer that is determined before the simulation starts) and it adopts at a certain time. Suppose that we change the timers of stem nodes of the dissemination tree to the mean value $\langle \tau \rangle$ of the timers (which is $\mu_\tau = 4$ in our example). The times that the stem nodes' neighbours in the dissemination tree adopt then also change, as the time of adoption of a node in an adoption path is determined by the sum of the node's timer and the time of adoption of the node's predecessor node. (Note that we are not rerunning the WTM dynamics.) In Fig. 3.8, we investigate how the adoption curves change because of the change of timers of stem nodes. The orange and the blue curves are both adoption curves for the WTM with heterogeneous timers on configuration-model networks with a Poisson degree distribution with mean $z = 6$. For the orange curve, the timers are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$. For the blue curve, the timers are determined from a Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ that we then round down to an integer. The curves with lighter colours are the adoption curves after we change the timers of the stem nodes of dissemination trees to the mean timer value. The light orange curve is for timers distributed uniformly at random, the light blue curve is for timers determined from a Gamma distribution, and the green curve (which we include to facilitate our comparison) is for the WTM with a homogeneous timer. We observe that intersections between the light-coloured curves and the green curve occurs earlier than the corresponding ones between the dark-coloured curves and the green curve. We also show in Table 3.5 how the time to reach certain fractions ρ^* of nodes calculated using Eq. (3.5) changes by adjusting the timers of stem nodes. From both Fig. 3.8 and Table 3.5, we observe that the adoption processes for the WTM with heterogeneous timers is delayed by changing the timers of stem nodes, even though they constitute fewer than 1% of all nodes in the networks.

In Fig. 3.9, we show our results for generalised configuration-model networks with 3-cliques (Congen-3) and a generalised configuration-model networks with 4-cliques

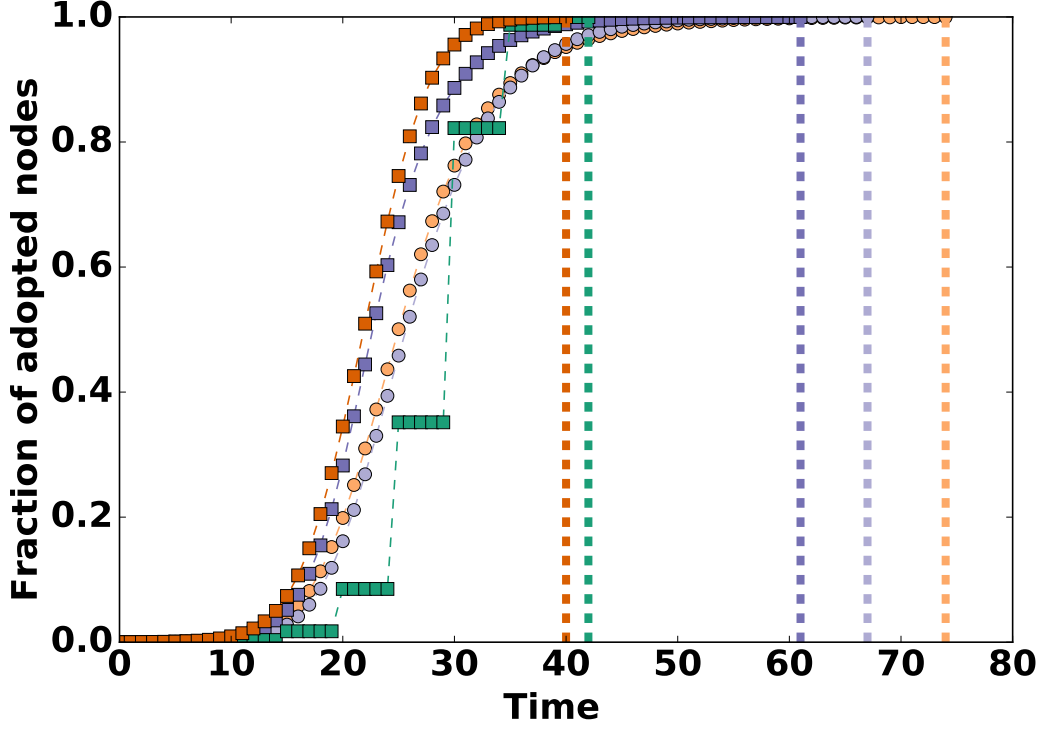


Figure 3.8: Adoption curves of the WTM with timers and their adoption curves after we change the timers of stem nodes of the dissemination trees. The green curve is for the WTM with homogeneous timers with $\tau = 4$, the orange curves are for the WTM with heterogeneous timers τ that are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and the blue curves are for heterogeneous timers determined from the Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and then rounding down to an integer. To isolate the effects of different distributions of timers, in each case, we run the WTM using the same networks with the same seed nodes. The dark orange and blue curves are before we change the timers in dissemination trees, and the corresponding light-coloured curves are after we change those timer values (see the main text). The squares (for the dark curves) and disks (for the light curves) are the results of numerical simulations, and the dashed lines mark the times at which the adoption process of the corresponding colour reaches a steady state. We obtain our numerical results by averaging over 1,000 realizations of the WTM with timers on configuration-model networks with $N = 10,000$ nodes and a Poisson degree distribution with mean $z = 6$. For each realization, we generate an independent network. We also determine the seed node (uniformly at random) and timer values separately for each realization.

Table 3.5: Time to reach certain fractions of adopted nodes for the WTM with timers on configuration-model networks and the changes of the time to reach certain fractions of adopted nodes after we adjust the timers of stem nodes of dissemination trees for the WTM with timers. We consider configuration-model networks with $N = 10,000$ nodes and a Poisson-degree distribution with mean $z = 6$. All nodes have a homogeneous adoption threshold of $\phi = 0.1$. For timer distributions, we use homogeneous timers with $\tau = 4$, heterogeneous timers distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and timers determined using a Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and then rounding down to an integer. To isolate the effects of different distributions of timers, in each case, we run the WTM using the same networks with the same seed nodes. We average over 1,000 realizations, and we determine the seed node (uniformly at random) and timer values separately for each realization. We also generate an independent network for each realization.

ρ^{*a}	t_{hom}^b	$t_{\text{unif,WTM}}^c$	$t_{\text{unif,dis}}^d$	$t_{\text{Gam,WTM}}^e$	$t_{\text{Gam,dis}}^f$
0.5	29.02	22.01	25.17	22.76	25.70
0.6	29.82	23.10	26.71	23.89	27.24
0.7	30.42	24.24	28.48	25.21	29.04
0.8	31.26	25.54	30.79	26.97	31.36
0.9	32.97	27.26	34.59	29.88	35.01
ρ_{st}^g	42.76	40.40	74.55	61.19	67.55

^a Fraction of adopted nodes.

^b Time to reach ρ^* for the WTM with a homogeneous timer.

^c Time to reach ρ^* for the WTM with timers distributed uniformly at random.

^d Time to reach ρ^* after we change the timers of stem nodes for timers that are distributed uniformly at random.

^e Time to reach ρ^* for the WTM with timers determined using a Gamma distribution.

^f Time to reach ρ^* after we change the timers of stem nodes for a timers determined using a Gamma distribution.

^g Fraction of adopted nodes at steady state.

(Congen-4). We consider different values of the edge-clique ratios α and β , where α determines the edge-clique ratio for a node of degree $k \geq 3$ in Congen-3 and β determines the edge-clique ratio for a node of degree $k \geq 4$ in Congen-4 [49, 66]. We use $\alpha = 0.5$ in Fig. 3.9a, $\alpha = 1.0$ in Fig. 3.9b, $\beta = 0.5$ in Fig. 3.9c, and $\beta = 1.0$ in Fig. 3.9d. In each figure, the orange and blue curves both represent adoption curves for the WTM with heterogeneous timers. For the orange curve, timers are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$; for the blue curve, timers are determined from a Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and are then rounded down to an integer. The curves with lighter colours are the adoption curves after we change the timers of stem nodes of dissemination

trees. The light-orange curve is for uniformly-randomly-distributed timers, and the light-blue curve is for Gamma-distributed timers. The green curve, which we include for comparison, is for a WTM with a homogeneous timer. We observe that the light-coloured curves intersect the green curve earlier than do the dark-coloured curves.

In Table 3.6, we show the change in times to achieve certain fractions ρ^* of nodes — calculated with Eq. (3.5) from Section 3.3.1 — after adjusting the timers of stem nodes. From both Fig. 3.9 and Table 3.6, we observe that adoption processes for heterogeneous timers are delayed by changing the timers of stem nodes, which constitute fewer than 1% of all nodes in the networks.

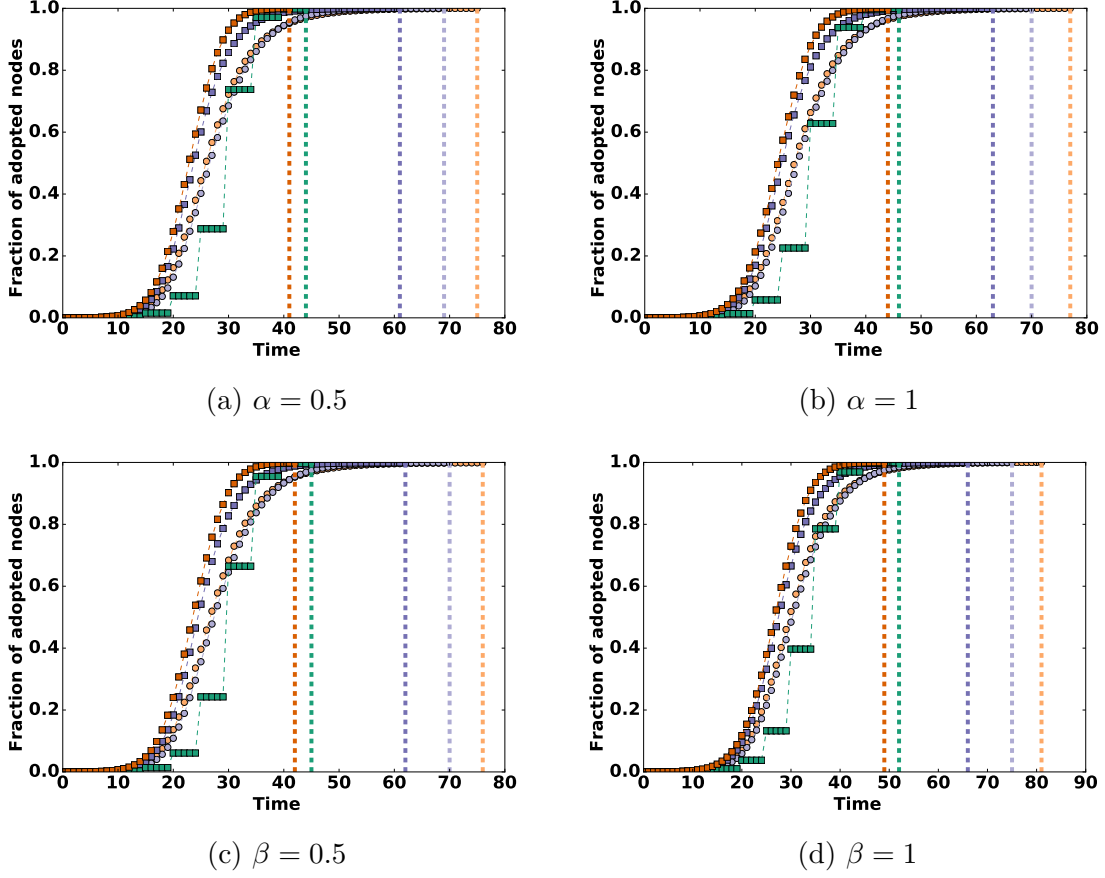


Figure 3.9: Adoption curves of the WTM with timers on generalised configuration-model networks we used in Table 3.3 and 3.4, and adoption curves after we adjust the timers of stem nodes of dissemination trees. We use (a,b) generalised configuration-model networks with 3-cliques and (c,d) generalised configuration-model networks with 4-cliques. We consider different values of the edge-clique ratios α and β ; we use the parameter values $\alpha = 0.5$ in panel (a), $\alpha = 1$ in panel (b), $\beta = 0.5$ in panel (c), and $\beta = 1$ in panel (d). For all networks, each node has a homogeneous threshold value of $\phi = 0.1$. The green curve is for the WTM with homogeneous timers with $\tau = 4$, the orange curves are for the WTM with heterogeneous timers τ that are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and the blue curves are for heterogeneous timers determined from the Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and then rounding down to an integer. To isolate the effects of different distributions of timers, in each case, we run the WTM using the same networks with the same seed nodes. The dark orange and blue curves are before we change the timers in dissemination trees, and the corresponding light-coloured curves are after we change those timer values (see Section 3.3.2.2). The squares (for the dark curves) and disks (for the light curves) are the results of numerical simulations, and the dashed lines mark the times at which the adoption process of the corresponding colour reaches a steady state. Our numerical results are means over 1,000 realizations. For each realization, we generate an independent network. We also determine the seed node (uniformly at random) and timer values separately for each realization.

Table 3.6: Time to reach certain fractions of adopted nodes for the adoption curves in Fig. 3.9. We use subscript t_{WTM} to indicate the WTM with timers, and we use subscript t_{dis} to indicate the dissemination tree with adjusted timers of stem nodes. We use subscript t_{hom} to indicate a homogeneous timer, t_{unif} to indicate timers distributed uniformly at random, and t_{Gam} to indicate timers determined using a Gamma distribution. The quantity ρ^* is the fraction of adopted nodes, and t is the time to reach corresponding ρ^* .

Congen-3										
$\alpha = 0.5$					$\alpha = 1.0$					
ρ^*	t_{hom}	$t_{\text{unif,WTM}}$	$t_{\text{unif,dis}}$	$t_{\text{Gam,WTM}}$	$t_{\text{Gam,dis}}$	t_{hom}	$t_{\text{unif,WTM}}$	$t_{\text{unif,dis}}$	$t_{\text{Gam,WTM}}$	$t_{\text{Gam,dis}}$
0.5	29.82	22.94	26.05	23.71	26.63	30.56	24.17	27.18	24.87	27.67
0.6	30.43	24.05	27.37	24.87	28.19	31.52	25.32	28.76	26.07	29.26
0.7	31.27	25.23	29.37	26.22	30.02	32.70	26.53	30.58	27.46	31.19
0.8	32.53	26.56	31.67	27.99	32.37	34.25	27.91	32.95	29.27	33.49
0.9	34.30	28.34	35.47	30.93	36.07	35.43	29.77	36.88	32.23	37.49
1.0	44.06	41.76	75.70	61.92	69.01	46.32	44.10	77.67	63.67	70.87

Congen-4										
$\beta = 0.5$					$\beta = 1.0$					
ρ^*	t_{hom}	$t_{\text{unif,WTM}}$	$t_{\text{unif,dis}}$	$t_{\text{Gam,WTM}}$	$t_{\text{Gam,dis}}$	t_{hom}	$t_{\text{unif,WTM}}$	$t_{\text{unif,dis}}$	$t_{\text{Gam,WTM}}$	$t_{\text{Gam,dis}}$
0.5	30.26	23.60	26.75	24.47	27.39	33.57	26.81	29.81	27.58	30.38
0.6	31.08	24.74	28.32	25.65	28.96	34.71	28.04	31.45	28.87	32.05
0.7	32.14	25.93	30.12	27.02	30.80	35.69	29.33	33.32	30.33	33.98
0.8	33.62	27.29	32.46	28.81	33.18	36.83	30.81	35.72	32.21	36.46
0.9	35.05	29.11	36.35	31.75	36.91	38.78	32.81	39.68	35.23	40.35
1.0	45.43	42.79	76.63	62.86	70.14	52.17	49.60	81.77	66.99	75.53

3.3.3 Adoption paths in networks with threshold $\phi = 0.16$

In the previous sections, we used $\phi = 0.1$ for large networks with mean degree 6, so most of the nodes in the networks adopt if any of their neighbours adopt. Here we use a homogeneous threshold $\phi = 0.16$ so that the number of nodes which adopt if any of their neighbours adopt is smaller than when $\phi = 0.1$.

In Table 3.7 we examine the timers of nodes in stems and branches when we use $\phi = 0.16$. As a result, we find that the mean timer, τ_s , of stem nodes is larger than that of the stem nodes in Table 3.4. The reason why τ_s is larger in Table 3.7 than in Table 3.4 is because nodes with large timers can be part of long adoption paths if we use a large threshold, as we discussed in Fig. 3.3. Therefore, the mean timer of stem nodes becomes larger if we use a larger node threshold.

However, τ_s is still smaller than the mean timer $\langle\tau\rangle$ of all nodes in the networks. Hence, although the time T to steady state for heterogeneous timers is either longer or shorter than for a homogeneous timer (i.e., $T_{\text{hom}} < T_{\text{Gam}}$ and $T_{\text{hom}} > T_{\text{unif}}$), the time $t_{0.9}$ that the majority (90%) of nodes adopt is longer for homogeneous timers than for both heterogeneous timers (i.e., $t_{0.9,\text{hom}} > t_{0.9,\text{Gam}}$ and $t_{0.9,\text{hom}} > t_{0.9,\text{unif}}$). The time to reach 90% of adopted nodes is delayed significantly if we change the timers of stem nodes — which only constitute a small percentage of all nodes in the networks — to the mean timer value $\langle\tau\rangle$.

Table 3.7: Comparison between the mean timer of stems and branches for the WTM model with a homogeneous timer, uniformly-randomly-distributed timers, and timers determined using a Gamma distribution. The networks that we use are a configuration model (Config), a generalised configuration model with 3-cliques (Congen-3), and a generalised configuration model with 4-cliques (Congen-4). For all networks, we use configuration-model networks with $N = 10,000$ nodes and degrees drawn from a Poisson distribution $P_k = z^k e^{-z}/k!$ with mean $z = 6$. Additionally, each node has a homogeneous threshold of $\phi = 0.16$. For the generalised configuration models, we consider different values of the edge-clique ratios α and β , where α determines the edge-clique ratio for a node of degree $k \geq 3$ in Congen-3 and β determines the edge-clique ratio for a node of degree $k \geq 4$ in Congen-4. The notation x_{hom} indicates that we calculate the quantity x for homogeneous timers, the notation x_{unif} indicates that we calculate the quantity x for timers that are distributed uniformly at random, and the notation x_{Gam} indicates that we calculate the quantity x for timers determined using a Gamma distribution. The quantity T is the time to steady state, and t_{ρ^*} is the time that it takes for adopted fraction ρ^* to reach a certain fraction of adopted nodes. (In this table, we use $\rho^* = 0.9$.) The quantity τ_s is the mean stem-node timer, τ_b is the mean branch-node timer, and $\langle \tau \rangle$ is the mean of all timers. To compare the effects of different timer distributions, we augment the WTM with the different timer distributions on the same network with the same seed node and the same adoption-threshold distribution for each realization. Each reported value is a mean over 100 simulations of the WTM with timers on networks generated independently for each simulation.

	Homogeneous timer				Uniformly random timers				Gamma-distributed timers			
	T_{hom}	$t_{0.9, \text{hom}}$	$\frac{\tau_{s, \text{hom}}}{\langle \tau \rangle}$	$\frac{\tau_{b, \text{hom}}}{\langle \tau \rangle}$	T_{unif}	$t_{0.9, \text{unif}}$	$\frac{\tau_{s, \text{unif}}}{\langle \tau \rangle}$	$\frac{\tau_{b, \text{unif}}}{\langle \tau \rangle}$	T_{Gam}	$t_{0.9, \text{Gam}}$	$\frac{\tau_{s, \text{Gam}}}{\langle \tau \rangle}$	$\frac{\tau_{b, \text{Gam}}}{\langle \tau \rangle}$
Config	72.58	63.19	1.0	1.0	70.89	58.69	0.51	1.02	90.54	62.35	0.52	1.03
Congen-3 ($\alpha = 0.5$)	72.66	62.17	1.0	1.0	71.03	57.41	0.47	1.01	91.33	60.73	0.49	1.01
Congen-3 ($\alpha = 1.0$)	72.19	60.90	1.0	1.0	69.72	55.14	0.45	1.01	89.74	58.73	0.49	1.01
Congen-4 ($\beta = 0.5$)	75.42	65.38	1.0	1.0	73.10	59.37	0.47	1.00	93.34	63.12	0.49	1.01
Congen-4 ($\beta = 1.0$)	76.47	62.76	1.0	1.0	72.98	56.34	0.48	0.99	91.85	59.88	0.50	1.01

The sizes of many real-world networks fall between the small networks in Section 3.2 and the large networks used in Section 3.3. However, we can imagine that the dynamical characteristics of the WTM with timers on large networks that we presented throughout this section are not going to be much different from networks of smaller or even larger sizes; by taking the mean of many realizations, the idea of ensemble expectations in the $N \rightarrow \infty$ limit does not change by changing network sizes. Therefore, we only present the dynamical characteristics of the WTM with timers on networks with $N = 10,000$, however it does not confine our analysis only to networks of this size; instead, we can generalise our findings to the families of networks we used in this section. In order to test if our investigations are more broadly relevant, we test our model on real-world networks in Chapter 5.

3.4 Summary

In this chapter, we studied some novel features that a timer brings to the dynamics of the WTM. We first studied the WTM with timers on small networks in which each node adopts once one of its neighbours adopts, and we found that adoption paths and adoption order can change depending on the network structure and the distribution of timers. We then considered larger networks, and examined the WTM with different distributions of timers on these networks. As a result, we discovered that the different distributions of timers not only produce different shapes of adoption curves, but they also take different times to reach steady state. However, we noticed in our examples that the majority of nodes in networks adopt earlier when timers are distributed heterogeneously. We investigated adoption paths to examine how adoptions spread in networks, and we observed that stems — the longest adoption paths in networks — play an important role in spreading adoptions faster for heterogeneous timers than for a homogeneous timer.

We also conducted numerical simulations to understand the behaviour of the WTM with timers. In Chapter 4, we study the WTM with timers analytically by using a pair approximation.

Chapter 4

Pair-approximation analysis of timer model

We present an analytical approximation for the temporal evolution of the fraction of adopted nodes of the WTM with timers. To do this, we use a pair approximation, whose application to the WTM and its variants have been studied at length [48, 52, 66, 99, 113]. A pair approximation of the WTM was first developed by Gleeson and Cahalane [52], who built on a method to study the zero-temperature random-field Ising model on Bethe lattices [31]. Gleeson and Cahalane’s pair approximation agrees well with the temporal evolution of the WTM, and it takes into account pairwise interaction between nodes [54].

In Section 4.1, we introduce the analytical model for the single-spin flip dynamics of the random field Ising model (RFIM) on a Bethe lattice at zero temperature. This motivated the development of pair approximation for the WTM on networks that we discuss in Section 4.2. We generalise the pair approximation in Section 4.3 to examine the temporal evolution of the adopted fraction of nodes in the WTM with timers, and we compare the analytical calculations with numerical results.

4.1 Zero-temperature random-field Ising model on Bethe lattices

Dhar et al. studied single-spin flip dynamics of the random field Ising model (RFIM) on a Bethe lattice at zero temperature [31]. A Bethe lattice is an infinite connected acycle graph in which each node is connected to w neighbours, where w is called the coordination number [80]. Figure 4.1a shows an example of a Bethe lattice with $w = 3$. In the RFIM on a Bethe lattice, each node has a spin state, and it can be in either of two states: up or down. The spin S_i of node i can change from down to up

if the external magnetic field h is larger than the local quenched random field h_i of node i .

To solve statistical mechanical problems (see, for example, [27, 36, 80]) on the Bethe lattice, one often employs a Cayley tree. An n -Cayley tree is a tree in which each non-leaf node has a constant n number of edges. Figure 4.1b shows a 3-Cayley tree. If we increase the size of an n -Cayley tree to infinity, then the topology of the deep interior part of the n -Cayley lattice can approximate the topology of a Bethe lattice with $w = n$. Therefore, the behaviour of a dynamical model on a Bethe lattice can be analogous to the behaviour of the model deep inside the Cayley tree [8, 123]. In fact, many correlation functions deep inside a Cayley tree are found to be the same as in the Bethe lattice [31].

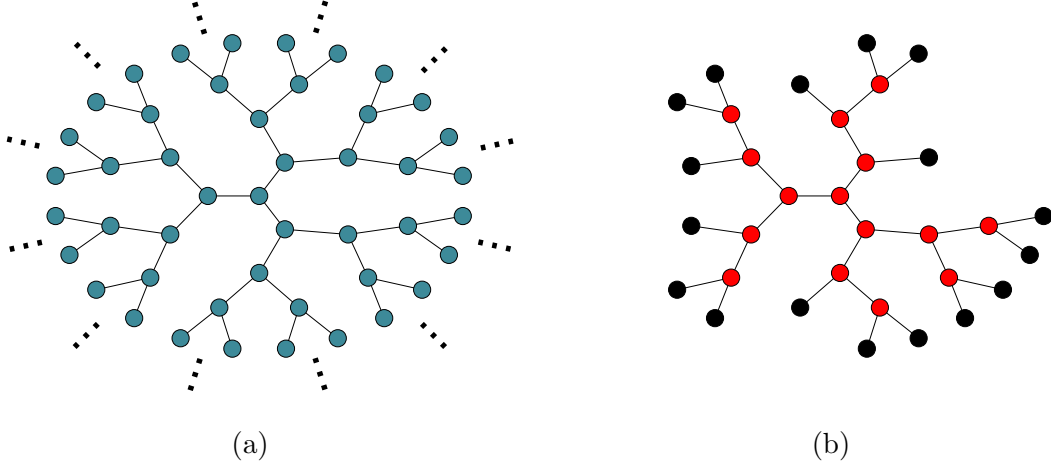


Figure 4.1: An example of (a) a Bethe lattice with $w = 3$, and (b) a 3-Cayley tree. Nodes in the boundary of the Cayley tree are coloured in black.

Consider a w -Cayley tree: each node in the Cayley tree has w neighbours, except the nodes in the boundary (i.e., leaves of the Cayley tree), which have 1 neighbour. Let us designate a root, which we locate at the top level. We then locate the neighbours of the root at a level below the root, and we then locate the neighbours of the neighbours at the following level. If we repeat this process until there are no more nodes to place at the following level, the number of nodes in the n^{th} level from the top is exactly $w(w - 1)^{n-1}$ nodes, except the levels with nodes in the boundary of the Cayley tree. A Cayley tree has n_{top} levels, and we place the root at the $n_{\text{top}}^{\text{th}}$ level and the boundary nodes at the 1st level.

To calculate the fraction of spin-up nodes in a Bethe lattice, we start with an external field h that is small enough to make all nodes have spin down. We then

increase h to a finite value, and we flip up any spin whose net local field is positive. We start flipping the spins of nodes from level 1, and then we move to levels 2, 3, and so on. If any spin at level n is flipped, we check the descendents again for possible spin flips. This gives *monotone* dynamics [50], because the spin can only flip upwards but not downwards.

Let P_n indicate the conditional probability that a uniformly randomly chosen node at level n flips to spin up, given that the spin of its neighbour at level $n + 1$ remains down. The quantity P_n is then determined by $w - 1$ neighbours at level $n - 1$, where the spins of these nodes are flipped up independently with probability P_{n-1} . Therefore, the probability that x neighbours at $n - 1$ level have flipped up their spins is $(1 - P_{n-1})^x P_{n-1}^{w-1-x}$. Let $p(s, h)$ denote the probability that the local field at a uniformly randomly chosen node is large enough so that the node's spin will flip up if s neighbours at a lower level have spin up. The probability P_n can then be expressed as

$$P_n = \sum_{x=0}^{w-1} \binom{w-1}{x} (P_{n-1})^x (1 - P_{n-1})^{w-1-x} p(x, h). \quad (4.1)$$

Assume that we are given $p(s, h)$ for all values of h . With the initial condition $P_1 = p(1, h)$, we can use (4.1) recursively to determine P_n for all $n \geq 2$. For large n , the probability P_n tends to a fixed point P^* given by the self-consistent equation

$$P^* = \sum_{x=0}^{w-1} \binom{w-1}{x} (P^*)^x (1 - P^*)^{w-1-x} p(x, h). \quad (4.2)$$

Similarly, the root at level n_{top} flips its spin up with probability

$$P_{\text{root}}^* = \sum_{x=0}^w \binom{w}{x} (P^*)^x (1 - P^*)^{w-x} p(x, h), \quad (4.3)$$

which is equivalent to the expected fraction of spin-up nodes in a Bethe lattice.

There are similarities between the RFIM on the Bethe lattice at zero temperature and the WTM on a random network. Both of the RFIM and the WTM are binary-state dynamics in which a node can be in either of two states: spin up and down for the RFIM, and adopted and unadopted for the WTM. They are also monotone dynamics in which a change of a node's state can occur in only one direction: from spin down to spin up for the RFIM, and from unadopted to adopted for the WTM. The local quenched random field can be considered similarly as the node threshold in

Table 4.1: Comparison between the Random field Ising model and Watts threshold model.

	Random field Ising model	Watts threshold model
Binary-state dynamics	Spin up and spin down	Adopted and unadopted
Monotone dynamics	Spin down to spin up	Unadopted to adopted
State-change condition	Net local field must be larger than 0	Adopted fraction of neighbours must be at least as large as threshold
No noise fluctuation	Zero-temperature	No noise factors that affect the adoption condition

the WTM, because the local quenched random field determines how many neighbours of a node should be in the spin-up state to change the state of the node. The zero-temperature setting can be interpreted as no influence on the adoption of a node in the WTM by any external forces, such as broadcast media [51]. We summarize this comparison between the RFIM and the WTM in Table 4.1.

With these similarities between the RFIM on a Bethe lattice and the WTM on a random network, we derive an analytical approximation (specifically, a pair approximation) for the temporal evolution of the adopted fraction for the WTM on a random network, and then we modify the approximation for the WTM with timers on random networks.

4.2 Pair approximation of the WTM

We consider a pair approximation of the WTM for undirected, unweighted networks. We assume that our networks are locally tree-like, so that, asymptotically as $N \rightarrow \infty$, cycles can be ignored (and there should be few short cycles in empirical networks) [97]. Using these assumptions, it has been shown [50, 52] that one can approximate the evolution of the fraction $\rho(t)$ of adopted nodes in a network by calculating the probability that a node chosen uniformly at random has adopted by time t .

To calculate the probability of a node to adopt, we first rearrange the network into a tree, with the chosen node at the top level (i.e., the ∞ level), and its neighbours on the next level (see Fig. 4.2). The position of a node in the tree is thus determined by the distance between it and the top-level node. A node’s “parent” is a neighbour located one level higher, and its “children” are its neighbours located one level lower.

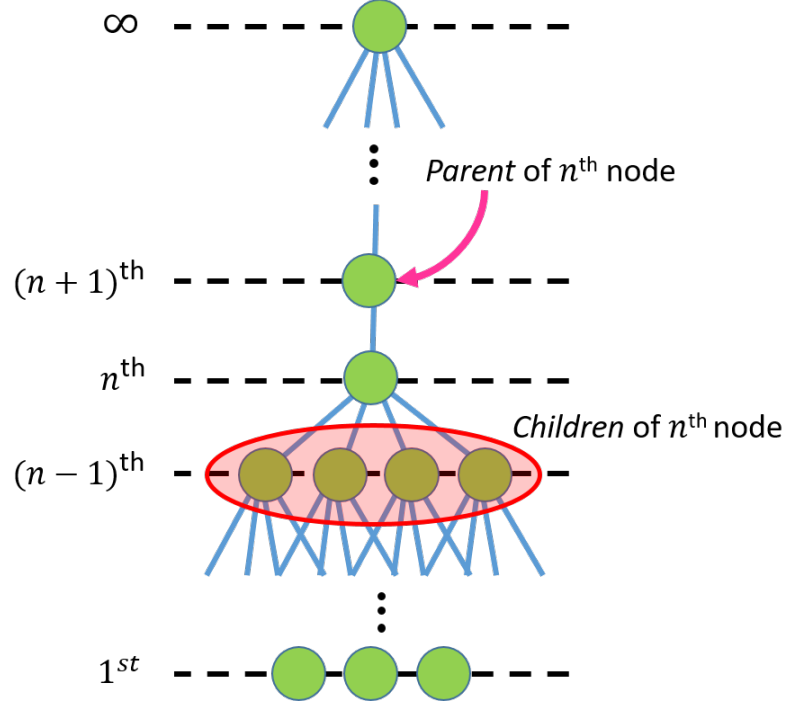


Figure 4.2: An illustration of level-by-level spreading of adoption behaviour in a network.

Because the threshold-adoption condition is that there are at least the threshold fraction of adopted neighbours, the probability of the top-level node to adopt is determined by the fraction of adopted children, whose probability of adoption is in turn determined by their childrens' adoption status, and so on. We thus write

$$\rho(t) = \rho_0 + (1 - \rho_0) \sum_{k=1}^{\infty} P_k \sum_{m=0}^k \binom{k}{m} q_{\infty}(t)^m [1 - q_{\infty}(t)]^{k-m} F(m, k), \quad (4.4)$$

where ρ_0 is the fraction of seed nodes, k is the degree, P_k is the probability that the node degree is k (i.e., $\{P_k\}$ is the degree distribution), m is the number of adopted children, and $F(m, k)$ is the (neighbourhood-influence) response function [66, 90, 139], which corresponds to the probability that a node satisfies the threshold-adoption condition for a given m and k . As we discussed in Section 3.1, the threshold-adoption condition of a node v_i in the WTM is that the fraction m/k of adopted neighbours is at least its threshold ϕ_i . Therefore, for the WTM, the response function $F(m, k)$ is the probability for a node to have a threshold smaller than m/k ; we can calculate this probability from the cumulative distribution function of the thresholds. Finally,

the term $q_n(t)$ is given by

$$q_n(t) = \rho_0 + (1 - \rho_0) \sum_{k=1}^{\infty} \frac{k}{z} P_k \sum_{m=0}^{k-1} \binom{k-1}{m} q_{n-1}(t)^m [1 - q_{n-1}(t)]^{k-1-m} F(m, k),$$

$$q_n(0) = \rho_0, \quad (4.5)$$

where z is the mean degree of a network. Equation (4.5) gives the probability that a node at level n of the tree has adopted at or before time t (i.e., that it is in the adopted state at time t), conditional on its parent node at level $(n+1)$ being unadopted. The rationale behind the construction of $q_n(t)$ is as follows: A node v_i chosen uniformly at random at level n is in the adopted state if either

- the node is a seed, which occurs with probability ρ_0 ;
- or the node is not a seed, which occurs with probability $(1 - \rho_0)$, but it meets the threshold-adoption condition at or before time t . This condition is to have a threshold lower than m/k when the node has $k-1$ child nodes at the $(n-1)^{th}$ level, with m among them having adopted before time t .

The factor $k-1$ comes from the fact that the node has an unadopted parent at level $(n+1)$, so a node with degree k and threshold m/k should have m adopted nodes among $k-1$ child nodes at level $n-1$; this yields the term $\binom{k-1}{m} q_{n-1}(t)^m [1 - q_{n-1}(t)]^{k-1-m}$. Finally, we need to sum over all possible k , because the nodes at level n follow the degree distribution $\{P_k\}$. One reaches the nodes at level $(n-1)$ by following an edge between a child at level $(n-1)$ and the node at level n , so we use the excess degree distribution $\{kP_k/z\}$.

The above pair approximation shows good agreement with numerical simulations of the WTM [50, 52], especially in large networks that are locally tree-like. This pair approximation has been generalised to several situations, including the study of the WTM on networks with community structure (including with heterogeneous communities) [48, 98] and other forms of clustering [49, 66].

4.3 Pair approximation of the WTM with timers

We cannot simply use (4.4) and (4.5) for the WTM with timers, as a timer affects the time that a node adopts. Therefore, we need to understand the effect of timers in nodes' adoption and modify the pair-approximation equations accordingly. In the WTM with timers, a node with timer τ waits τ time steps after its threshold fraction

of neighbours have adopted before it adopts. In other words, the adoption of a node is determined by both (i) its timer τ and (ii) the fraction of neighbours that have adopted τ time steps before the current time step. We need to consider both of these facets to derive a pair approximation for the WTM with timers.

The condition associated with (i) is determined by the response function $G(m, k, \tau)$ of the WTM with timers that decides whether a node has satisfied both the threshold-adoption condition and the timer-adoption condition. The response function $G(m, k, \tau)$ is the probability that a node has threshold less than m/k and has timer τ . It is given by

$$G(m, k, \tau) = F(m, k)[C_\tau(\tau) - C_\tau(\tau - 1)], \quad (4.6)$$

where $F(m, k)$ is the response function of the WTM and C_τ is the cumulative distribution function of the timers.

One can satisfy the condition associated with (ii) if a sufficient fraction m/k of the children at level $(n - 1)$ have adopted τ time steps before the current time step t . The probability that a child at level $n - 1$ has adopted τ time steps ago is $q_n(t - \tau)$. Combining the conditions from (i) and (ii), we can express the adoption condition of a node with degree k , threshold m/k , and timer τ at time t as

$$\binom{k}{m} q_n(t - \tau)^m [1 - q_n(t - \tau)]^{k-m} G(m, k, \tau). \quad (4.7)$$

Similar to equations (4.4) and (4.5), we need to sum over all m , k , and τ . We thereby obtain

$$\rho(t) = \rho_0 + \sum_{\tau=0}^t \left[(1 - \rho_0) \sum_{k=1}^{\infty} P_k \sum_{m=0}^k \binom{k}{m} q_\infty(t - \tau)^m [1 - q_\infty(t - \tau)]^{k-m} G(m, k, \tau) \right], \quad (4.8)$$

$$q_{n+1}(t) = \rho_0 + \sum_{\tau=0}^t \left[(1 - \rho_0) \sum_{k=1}^{\infty} \frac{k}{z} P_k \sum_{m=0}^{k-1} \binom{k-1}{m} q_n(t - \tau)^m [1 - q_n(t - \tau)]^{k-1-m} G(m, k, \tau) \right], \quad (4.9)$$

with $q_n(0) = \rho_0$. As we saw in Fig. 3.4a, a numerical evaluation (solid curve) of the algebraic equations (4.4,4.5) derived from our theory shows good agreement with direct numerical simulations (squares) of the WTM with timers for ER networks.

Figure 4.3 shows the results of numerical simulations of the WTM with timers and the numerical evaluations of the algebraic equations (4.4,4.5) for standard configuration-model networks, generalised configuration-model networks with 3-cliques, and generalised configuration-model networks with 4-cliques. The green curves are for homogeneous timers with $\tau = 4$, the orange curves are for the WTM with heterogeneous

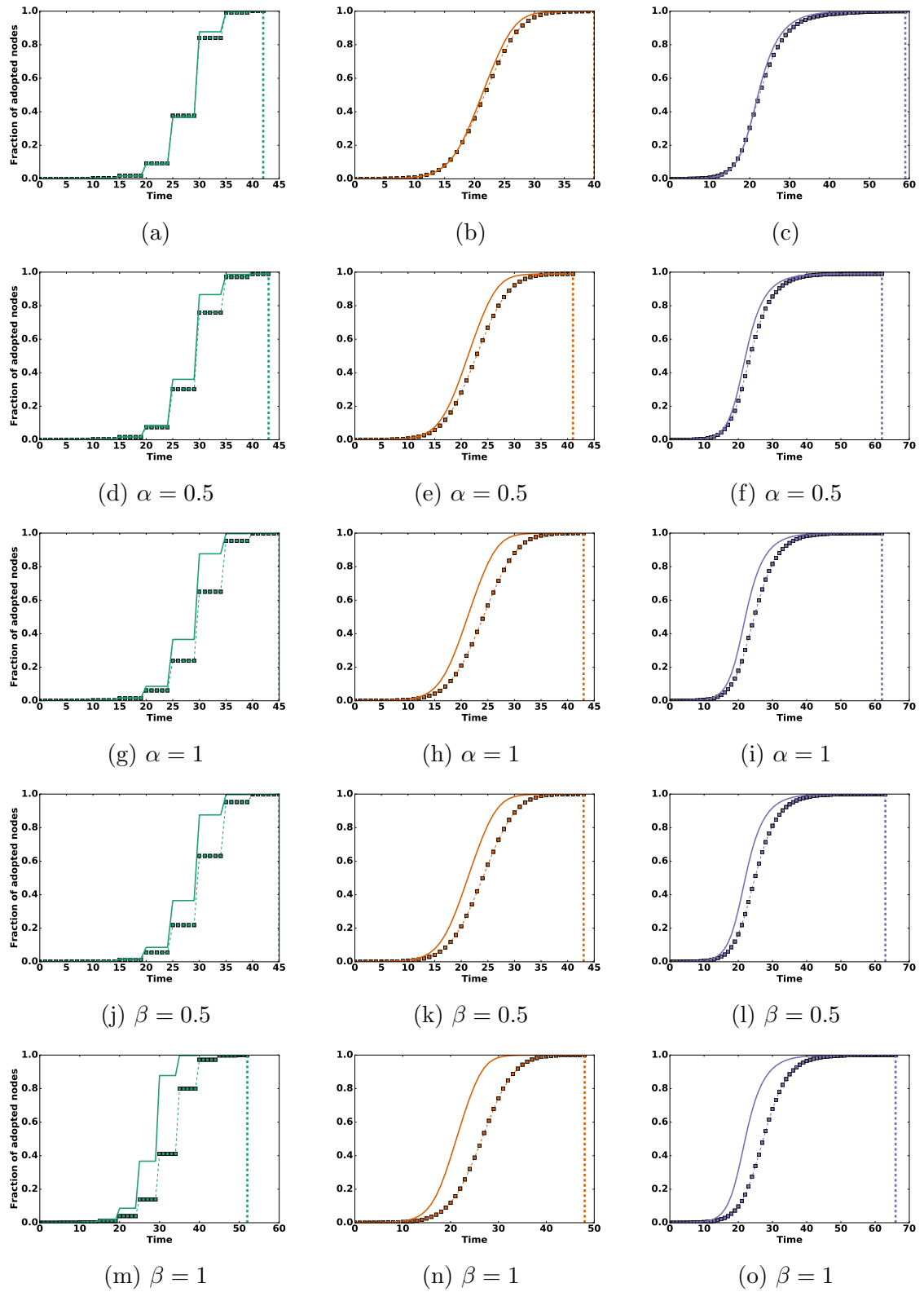


Figure 4.3: (Continue on the following page.)

Figure 4.3: Adoption curves of the WTM with timers on configuration-model networks with $N = 10,000$ nodes and a Poisson degree distribution with mean $z = 6$. We use (a–c) standard configuration-model networks, (d–i) generalised configuration-model networks with 3-cliques and (j–o) generalised configuration-model networks with 4-cliques. We consider different values of the edge–clique ratios α and β , where α determines the edge–clique ratio for a node of degree $k \geq 3$ in Congen-3 networks and β determines the edge–clique ratio for a node of degree $k \geq 4$ in Congen-4 networks. We use the parameter values $\alpha = 0.5$ in panels (d–f), $\alpha = 1$ in panels (g–i), $\beta = 0.5$ in panels (j–l), and $\beta = 1$ in panels (m–o). For all networks, each node has a homogeneous threshold value of $\phi = 0.1$. The green curve is for the WTM with homogeneous timers with $\tau = 4$, the orange curves are for the WTM with heterogeneous timers τ that are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and the blue curves are for heterogeneous timers determined from the Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and then rounding down to an integer. To isolate the effects of different distributions of timers, in each case, we run the WTM using the same networks with the same seed nodes. Dashed lines mark the times at which the adoption process of the corresponding colour reaches a steady state. Our numerical results are means over 1,000 realizations. For each realization, we generate an independent network. We also determine the seed node (uniformly at random) and timer values separately for each realization.

timers τ that are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and the blue curves are for heterogeneous timers determined from the Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and then rounding down to an integer. As we can see from the results, the discrepancies between the numerical simulation and analytical approximation are larger for generalised configuration-model networks with cliques than for standard configuration-model networks, and the discrepancies become larger as networks include a larger number of cliques (which results from having larger α or β). This occurs because equations (4.4), (4.5) are derived based on the assumption that the structure of a network is locally tree-like. Therefore, the analytical approximation is not guaranteed to work well for networks that include cycles. Many real-world networks have high global clustering coefficient and community structure, and some scientists attribute such structural characteristics of real-world networks to homophily [15, 61, 95]. We show the results of numerical simulations of the WTM with timers and the analytical results of (4.4) and (4.5) for real-world networks (e.g., FACEBOOK100 networks [131]) in Chapter 5.

4.3.1 Cascade condition for the WTM with a timer

As we saw in Section 2.3, a global cascade occurs when a small seed fraction ρ_0 of adopters leads to a much larger ρ_∞ . As we can see in the equation (4.8), the quantity ρ_∞ can be estimated by q_∞ , which is the fixed point of (4.9). In this section, we calculate the condition for a small seed fraction $\rho_0 = q_n(0)$ result in a global cascade (i.e., $\rho_\infty \gg \rho_0 = q_n(0)$).

Let us rewrite (4.9) as

$$q_{n+1}(t) = \rho_0 + (1 - \rho_0)K[q_n(t)], \quad (4.10)$$

where the nonlinear function $K[q_n(t)]$ is

$$K[q_n(t)] = \sum_{\tau=0}^t \sum_{k=1}^{\infty} \frac{k}{z} P_k \sum_{m=0}^{k-1} \binom{k-1}{m} q_n(t-\tau)^m [1 - q_n(t-\tau)]^{k-1-m} G(m, k, \tau). \quad (4.11)$$

We consider the $t \rightarrow \infty$ limit because we are interested in steady state behaviour of the WTM with timers. For a constant timer $\tau \ll t$, the probability $q_n(t-\tau) \approx q_n(t)$ as $t \rightarrow \infty$. Additionally, the sum $\sum_{\tau}^{\infty} G(m, k, \tau) = F(m, k)$ because $\sum_{\tau}^{\infty} [C_{\tau}(\tau) - C_{\tau}(\tau-1)] = 1$. For $\tau \approx t$, the response function $G(m, k, \tau) \approx 0$ because $\lim_{\tau \rightarrow t} [C_{\tau}(\tau) - C_{\tau}(\tau-1)] \approx 0$ as $t \rightarrow \infty$. Therefore, as $t \rightarrow \infty$, we can rewrite $K[q_n(t)]$ as

$$K[q_n(t)] \approx \sum_{k=1}^{\infty} \frac{k}{z} P_k \sum_{m=0}^{k-1} \binom{k-1}{m} q_n(t)^m [1 - q_n(t)]^{k-1-m} F(m, k). \quad (4.12)$$

Note that $K(0) = F(0)$.

We can linearize (4.10) near $q = 0$ to obtain

$$q_\infty \approx \rho_0 + (1 - \rho_0) [K(0) + q_\infty K'(0)], \quad (4.13)$$

where $K'(0)$ is the coefficient of the first-order term of the Taylor expansion around $q = 0$:

$$K'(0) = \sum_{k=1}^{\infty} \frac{k}{z} P_k (k-1) \left[F\left(\frac{1}{k}\right) - F(0) \right]. \quad (4.14)$$

We can rearrange (4.13) as follows:

$$q_\infty \approx -\frac{(1 - \rho_0)K(0) + \rho_0}{(1 - \rho_0)K'(0) - 1}. \quad (4.15)$$

If the denominator of (4.15) is negative, then the resulting approximation to q_∞ is positive. Additionally, the order of magnitude of q_∞ is of the same order of magnitude of ρ_0 , so the resulting ρ_∞ cannot be much larger than ρ_0 ; a global cascade cannot occur. Therefore, for a global cascade to occur, it is necessary that

$$(1 - \rho_0)K'(0) - 1 > 0. \quad (4.16)$$

By inserting (4.14) into (4.16), we can see that the cascade condition (2.39) that we saw in Section 2.3 is a special case of (4.16) when $\rho_0 = 0$ and $F(0) = 0$.

The cascade condition (4.16) for the WTM with timers is the same as the cascade condition for the WTM without a timer. In fact, we can intuitively understand that the two conditions are identical; because the WTM dynamics are monotonic, if a node adopts in the limit $t \rightarrow \infty$, the node also adopts in the WTM with a timer in the limit $t \rightarrow \infty$; consequently, in the $t \rightarrow \infty$ limit, $q_n(t)$ of the WTM with timers and $q_n(t)$ of the WTM yields the same value $q_n(\infty)$. Therefore, the cascade condition of the WTM with timers is identical to the cascade condition of the WTM without timers.

4.4 Summary

We began this chapter by introducing the zero-temperature random-field Ising model on Bethe lattices, which motivated the derivation of a pair approximation for the WTM. We modified the pair approximation for the WTM to incorporate a timer, and we observed that our analytical calculation agrees well with the numerical result of the temporal evolution of the adopted fraction of nodes, particularly when networks are locally tree-like. If we use networks with cliques, then the discrepancies between the numerical and analytical results become larger (and even larger if we increase the number of cliques in the networks). This makes sense, as we derived our analytical equation assuming that networks have, locally, tree-like structure. It would be interesting to develop our analytical approach further to incorporate non-tree-like structures in networks, such as local clustering and community structure.

We also used a pair approximation to derive a cascade condition for the WTM with timers. Because the WTM dynamics are monotonic, we see that the cascade condition for both the WTM with timer and the WTM without timers should be the same in the $t \rightarrow \infty$ limit.

Because our main focus has hitherto been to understand the behaviour of the WTM with a timer, we have tested the model on random-network models that are well-studied. In Chapter 5, we study the behaviour of the WTM with timers on networks generated using real-world data. We also investigate ways to use the WTM with timers for understanding real-world data on the spread of adoptions.

Chapter 5

WTM with timers on real networks

In this chapter, we study the WTM with timers on real-world networks. In Section 5.1, we run the WTM with timers on FACEBOOK100 networks [130, 131], and we investigate the resulting adoption curves. We use these simulations to investigate the dissemination trees in Section 5.2, and compare stems and branches. In Section 5.3, we suggest a way to calculate thresholds and timers from real-world data on the dissemination of information. Specifically, in Section 5.3.1, we use three well-known data sets on the spread of innovations. Finally, in Section 5.3.2, we use the WTM with timers to analyse Twitter dynamics related to The Spanish May 15th Movement [20].

5.1 Adoption process for the WTM with timers on real-world networks

We investigate the adoption process for the WTM with timers on nine FACEBOOK100 networks [130, 131]. Each network represents a Facebook friendship network at one university in the United States. An undirected and unweighted edge¹ between two nodes (i.e., users) indicates that the two nodes are friends on Facebook on one particular day in September 2005. At that time, one needed a .edu e-mail address to have an account on Facebook. In each of these networks, a node is an individual, and an unweighted, undirected edge represents a Facebook friendship between a pair of individuals. The networks in Table 5.1 have different mean degrees — some small and some large — but they all have heavy-tailed degree distributions. Therefore, to ensure that a node whose degree is equal to the mean degree adopts if a single

¹In Facebook, a friendship is formed between two individuals if and only if the two agree to form a friendship. In other words, friendship is a bidirectional relationship in Facebook, and we considered all friendships to have equal strength. In this way, we used undirected and unweighted edges to represent friendship on Facebook networks.

Table 5.1: The characteristics of nine FACEBOOK100 networks: from top to bottom, we give details for California Institute of Technology, Reed College, Haverford College, Simmons College, and Swarthmore College, Howard University, Rice University, Cornell University, and University of Michigan. The quantity N is the number of nodes, z is the network’s mean degree, and $\langle l \rangle$ is the mean shortest-path length between pairs of nodes.

	N	z	$z_{\max}$	C	$\langle l \rangle$
Caltech	762	43	248	0.41	1.54
Reed	962	39	313	0.32	1.62
Haverford	1446	82	375	0.32	1.50
Simmons	1510	43	300	0.32	2.57
Swarthmore	1657	73	577	0.30	2.32
Howard	4047	101	1215	0.23	1.59
Rice	4083	90	581	0.29	1.65
Cornell	18621	84	3156	0.22	1.91
Michigan	30106	78	2031	0.21	2.04

one of its neighbours has adopted, we consider a homogeneous adoption threshold of $\phi = 0.01$; subsequently, we choose a seed adopted node uniformly at random at $t = 0$. We now incorporate both homogeneous and heterogeneous timers into a spreading process on the FACEBOOK100 networks, and we examine the WTM with a timer on the LCC of the FACEBOOK100 networks. In our simulations, we report sample means of many realizations.

In Fig. 5.1, we show the adoption curves of the WTM with timers on nine FACEBOOK100 networks. The green curves are for a homogeneous timer $\tau = 4$, the orange curves are for heterogeneous timers distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and the blue curves are for heterogeneous timers given by integers that we round down from a random variable that follows the Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$. Similar to the results for random networks in Fig. 3.4a, the adoption curves for heterogeneous timers are much smoother than the adoption curves for a homogeneous timer. This is because the nodes that adopt simultaneously for a homogeneous timer — which make adoption curves for a homogeneous timer have stair-like shape — adopt at different times for heterogeneous timers, and fewer nodes adopt simultaneously. Again, similar to Fig. 3.4, the time to steady state for a homogeneous timer (green dashed line) is longer than the time to steady state for uniformly randomly distributed timers (orange dashed line), and the time to steady state for Gamma distributed timers (blue dashed line) is longer than the time to steady state for a homogeneous timer.

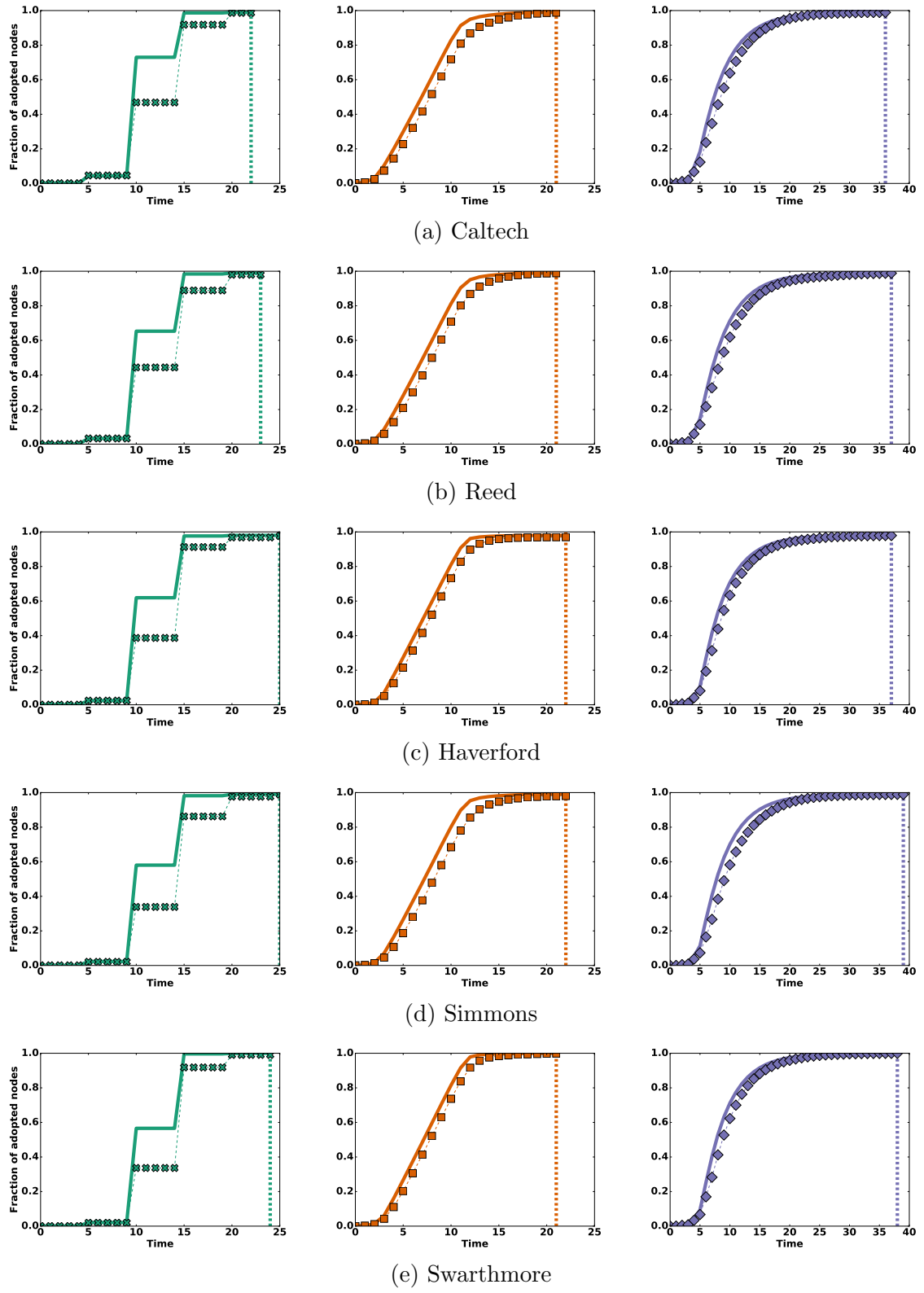


Figure 5.1: (Continue on the following page.)

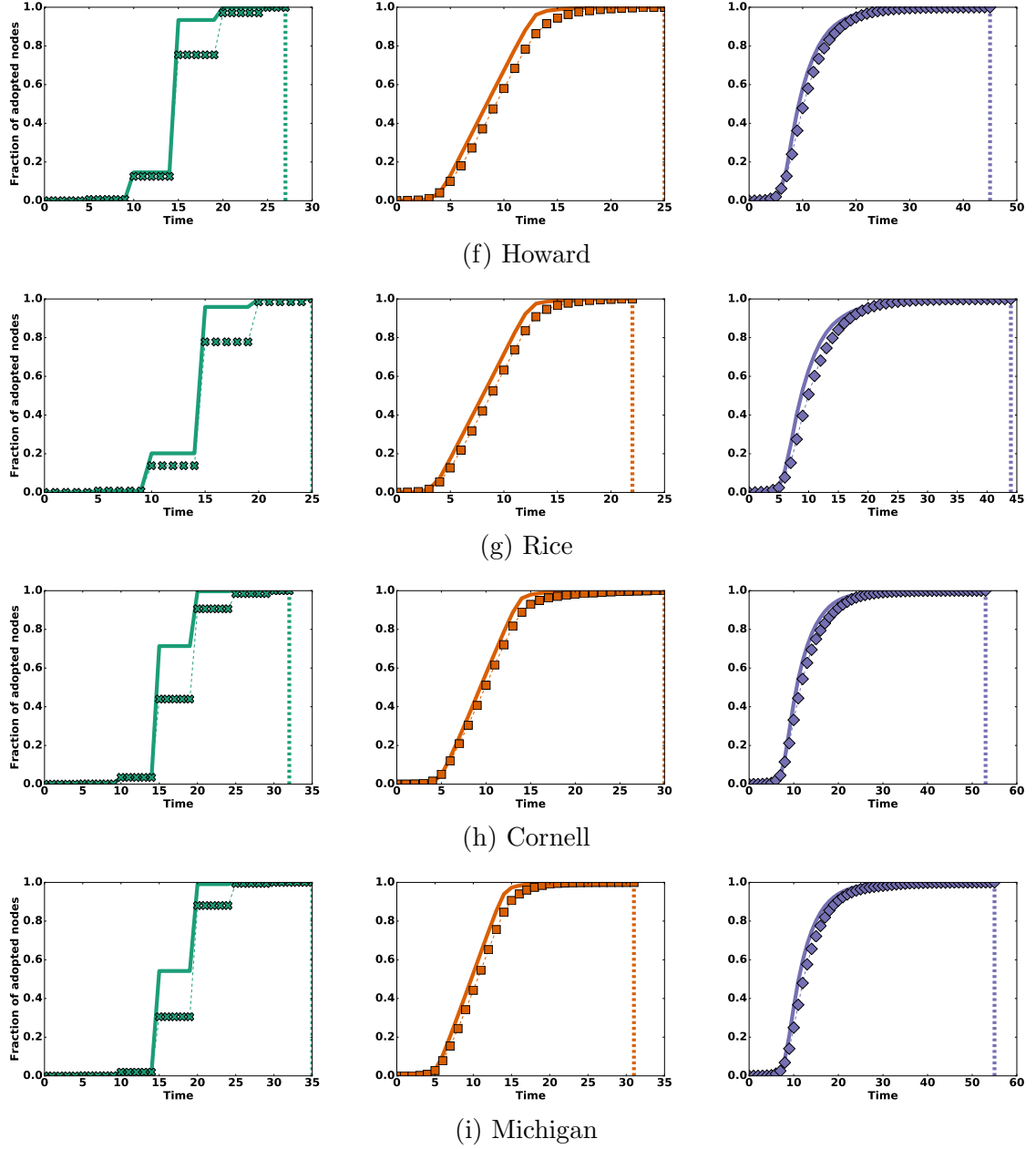


Figure 5.1: Adoption curves of the WTM with timers on the nine FACEBOOK100 networks [130, 131]. The green crosses are for the WTM with homogeneous timers with $\tau = 4$, the orange squares are for the WTM with timers that are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and the blue diamonds are for timers determined from the Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and then rounding down to an integer. The dashed lines mark the times at which the adoption process of the corresponding colour reaches a steady state, and the solid curves are results from an analytical approximation (see Section 4.3). To isolate the effects of different distributions of timers, in each case, we run the WTM using the same networks with the same seed nodes. Our numerical results are means over 1,000 realizations. We also determine the seed node (uniformly at random) and timer values separately for each realization.

There are several notable differences between Fig. 3.4a and Fig. 5.1. An obvious difference between the two figures is that the discrepancies between the numerical simulations (squares) and the analytical approximations (solid curves) are much larger for FACEBOOK100 networks than for random networks. This is because FACEBOOK100 networks do not have locally tree-like structures; instead, FACEBOOK100 networks have high clustering as well as large transivities [130], and community structures [131].

Another difference between Fig. 3.4a and Fig. 5.1 is that the adoption curves — both numerically computed ones and analytically approximated ones — for uniformly randomly distributed timers on FACEBOOK100 networks (i.e., the orange curves in fig. 5.1) grow less rapidly than the adoption curve for uniformly randomly distributed timers on random networks (i.e., the orange curve in fig. 3.4). For analytically approximated adoption curves, we can trace which term or parameter in the approximation made the two analytical curves show different behaviour. As we saw in section 4.3, there are four terms that determine the adopted fraction $\rho(t)$ at time t : the term ρ_0 that is determined by the size of seed fraction, the term $g(m, k, \tau)$ that is determined by the distributions of thresholds and timers, the term p_k that is determined by degree distribution, and the term q_∞ that is the fixed point of recursive formula q_n . because q_n is a recursive formula with terms ρ_0 , $g(m, k, \tau)$ and p_k , it is effectively the terms ρ_0 , $g(m, k, \tau)$ and p_k that determine $\rho(t)$. Among the three terms, seed fraction is kept constant for both random networks and the nine FACEBOOK100 networks. Additionally, for both random networks and FACEBOOK100 networks, we use the same distribution functions (with the same mean and standard deviation) to determine the thresholds, and the same distribution functions with the same mean and standard deviation to determine the timers. The degree distributions p_k , however, are different between the random networks and FACEBOOK100 networks; therefore, the difference between the results of analytical approximation for random networks and the analytical approximation for FACEBOOK100 networks arise from the different degree distributions.

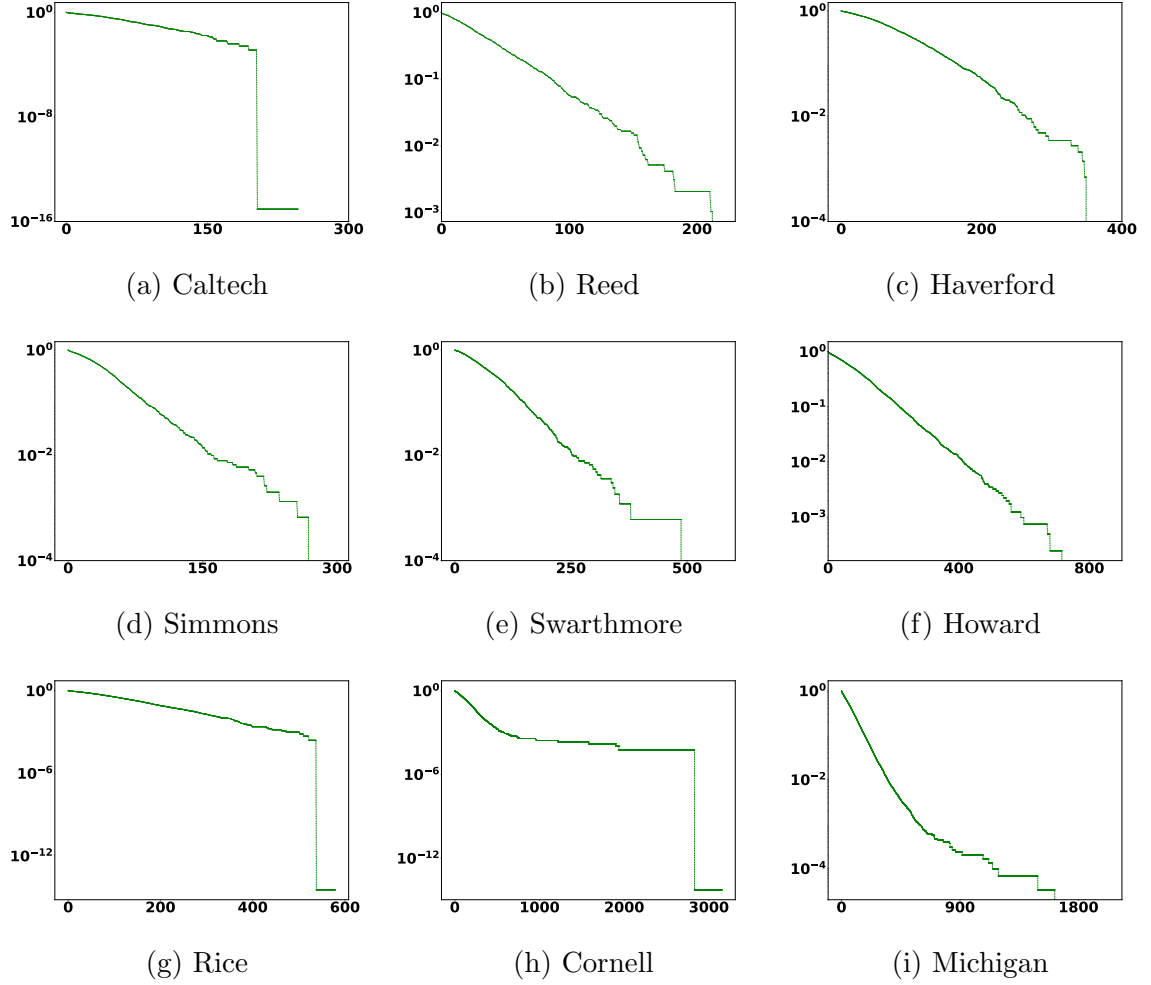


Figure 5.2: Cumulative degree distributions of the nine FACEBOOK100 networks: from left to right, (a) Reed College, (b) Simmons College, (c) California Institute of Technology, (d) Haverford College, and (e) Swarthmore College. The vertical axes are plotted on a logarithmic scale.

We show the cumulative degree distributions of the nine FACEBOOK100 networks in Fig. 5.2. As one can see from Fig. 5.2, the degree distributions of the nine FACEBOOK100 networks have heavier tails compared to random networks we used in Chapter 3, whose degree distributions are Poisson². Therefore, we can speculate that such degree distributions of the nine FACEBOOK100 networks with heavy tail make the adoption curves for uniformly randomly distributed timers different from the adoption curves for uniformly randomly distributed timers for the random networks.

²In our thesis, we used configuration models with Poisson-degree distribution and ER networks. As we saw in Chapter 2, the degrees of nodes in ER networks follow Poisson distribution in $N \rightarrow \infty$ limit, and Poisson distribution does not have heavy tail.

5.2 Dissemination trees for real-world networks

We investigate dissemination trees for the WTM with timers on the aforementioned five FACEBOOK100 networks [130, 131]. To ensure that a node whose degree is equal to the mean degree adopts if a single one of its neighbours has adopted, we consider a homogeneous adoption threshold of $\phi = 0.01$. In Table 5.2, we show the quantities that characterize the stems and branches for the WTM model with timers on the five FACEBOOK100 networks. For each of the five networks, we observe qualitatively similar results as in the random-graph ensembles, although the FACEBOOK100 networks have different structural characteristics (e.g., in terms of local clustering and community structure) than the random-graph models. We find that the percentage f_{a_s} of stems is smaller than the percentage f_{a_b} of branches, and that the numbers n of adoption paths are larger for the WTM with homogeneous timers than with heterogeneous timers. Accordingly, for heterogeneous timers, the lengths l_s of the stems and the lengths l_b of the branches are both larger than for homogeneous timers. Additionally, $\frac{\tau_{s,\text{het}}}{\langle\tau\rangle} < 1$ and $\frac{\tau_{b,\text{het}}}{\langle\tau\rangle} > 1$.

As we can see from the paragraph above, stems tend to grow faster and become longer than branches for the WTM with heterogeneous timers. In other words, the characteristics of the stems and the branches for the five FACEBOOK100 networks seem to be similar to those of the random networks that we studied in Section 3.3.2.2. When timers are heterogeneous, stems grow faster and become longer, initiating adoption spreading in branches earlier than when the timers are homogeneous. Consequently, a majority of nodes adopt earlier for the WTM model with heterogeneous timers than with homogeneous ones. In Table 5.2, we show values for how long it takes for at least 90% of the nodes to adopt.

In Table 5.3, we give the times to reach certain fractions ρ^* of adopted nodes for FACEBOOK100 networks [131] when timers are homogeneous, distributed uniformly at random, and determined using a Gamma distribution and then rounded down to an integer.

Table 5.2: Comparison between stem and branch characteristics for the WTM model with timers on FACEBOOK100 networks. Each node has a homogeneous adoption threshold of $\phi = 0.01$. The notation x_{hom} indicates that we calculate the quantity x for homogeneous timers, the notation x_{unif} indicates that we calculate the quantity x for timers that are distributed uniformly at random, and the notation x_{Gam} indicates that we calculate the quantity x for timers determined using a Gamma distribution (and then round down to an integer). The quantity T is the time it takes to reach a steady state, and t_{ρ^*} is the time that it takes for a fraction ρ^* of nodes to adopt. (We use $\rho^* = 0.9$ in this table; see Table 5.3 for other values of ρ^* .) The quantity a is the mean number of adoption paths, μ_l is the mean adoption path length, σ_l is the standard deviation of the adoption path lengths, and v is the structural virality. The quantity f_{a_s} is the mean percentage of stems among adoption paths, f_{a_b} is the mean percentage of branches among adoption paths, l_s is the mean stem length, l_b is the mean branch length, τ_s is the mean timer value of stem nodes, τ_b is the mean timer of branch nodes, and $\langle \tau \rangle$ is the mean value of all timers. Each quantity is a mean over 100 realizations with different timers and seed nodes (which are determined uniformly at random).

	T_{hom}	$t_{0.9, \text{hom}}$	a_{hom}	$\mu_{l, \text{hom}}$	$\sigma_{l, \text{hom}}$	v_{hom}	Homogeneous timer				
							$f_{a_s}(\%)$	$f_{a_b}(\%)$	$l_{s, \text{hom}}$	$l_{b, \text{hom}}$	$\frac{\tau_{s, \text{hom}}}{\langle \tau \rangle}$
Caltech	21.94	15.55	17083.39	4.25	0.36	3.13	7.15	92.85	5.39	1.54	1.0
Reed	23.38	16.03	18329.97	4.36	0.40	3.29	5.75	94.25	5.68	1.71	1.0
Haverford	25.86	15.60	78135.33	4.31	0.38	2.98	2.63	97.37	6.17	2.46	1.0
Simmons	25.14	16.64	36247.05	4.53	0.39	3.40	3.79	96.21	6.03	1.93	1.0
Swarthmore	24.39	15.97	83141.94	4.40	0.39	3.10	3.12	96.88	5.88	1.98	1.0
Howard	27.32	18.29	396182.38	4.89	0.40	3.32	1.40	98.60	6.46	1.86	1.0
Rice	25.05	18.01	416740.36	4.87	0.40	3.37	5.72	94.28	6.01	1.65	1.0
Cornell	32.40	20.85	3562141.22	5.39	0.45	3.78	0.26	99.74	7.48	2.14	1.0
Michigan	35.76	21.52	3691495.95	5.52	0.40	3.98	0.03	99.97	8.15	3.00	1.0
	T_{unif}	$t_{0.9, \text{unif}}$	a_{unif}	$\mu_{l, \text{unif}}$	$\sigma_{l, \text{unif}}$	v_{unif}	Uniformly random distribution of timers				
							$f_{a_s}(\%)$	$f_{a_b}(\%)$	$l_{s, \text{unif}}$	$l_{b, \text{unif}}$	$\frac{\tau_{s, \text{unif}}}{\langle \tau \rangle}$
Caltech	21.19	11.85	2721.44	4.83	0.52	3.89	1.55	98.45	7.30	2.10	1.03
Reed	21.92	11.96	3223.54	5.01	0.60	4.11	0.95	99.05	8.00	2.43	1.02
Haverford	22.04	11.50	13573.66	5.33	0.57	3.79	0.83	99.17	8.11	2.51	1.02
Simmons	23.34	12.13	6516.15	5.40	0.57	4.31	1.30	98.70	8.19	2.45	1.02
Swarthmore	21.02	11.66	14358.79	5.41	0.56	3.94	0.95	99.05	8.13	2.38	1.02

Table 5.2: (Continued from the previous page.)

		Gamma distribution (and then rounding down to an integer) of timers											
	T_{Gam}	$t_{0.9, \text{Gam}}$	a_{Gam}	$\mu_{l, \text{Gam}}$	$\sigma_{l, \text{Gam}}$	v_{Gam}	$f_{a_s}(\%)$	$f_{a_b}(\%)$	$l_{s, \text{Gam}}$	$l_{b, \text{Gam}}$	$\frac{\tau_{s, \text{Gam}}}{\langle \tau \rangle}$	$\frac{\tau_{b, \text{Gam}}}{\langle \tau \rangle}$	
Howard	25.39	13.18	90301.69	6.36	0.66	4.09	0.31	99.69	9.56	2.93	0.14	1.02	
Rice	22.43	12.61	77854.58	6.17	0.57	4.14	0.90	99.10	8.75	2.51	0.12	1.02	
Cornell	30.11	13.94	608076.74	7.25	0.60	4.79	0.10	99.90	10.97	3.59	0.10	1.01	
Michigan	31.24	14.49	756821.64	7.54	0.53	5.04	0.03	99.97	11.78	4.22	0.11	1.00	
Caltech	36.15	15.39	2586.5	4.29	0.44	4.01	2.32	97.68	6.20	1.96	0.45	1.03	
Reed	37.60	15.57	2902.57	4.40	0.47	4.23	1.89	98.11	6.55	2.12	0.45	1.02	
Haverford	38.24	15.15	8664.29	4.51	0.43	4.11	1.03	98.97	6.88	2.59	0.47	1.01	
Simmons	39.84	15.86	5069.47	4.62	0.46	4.48	1.78	98.22	6.72	2.17	0.43	1.02	
Swarthmore	38.71	15.28	9306.55	4.58	0.44	4.24	1.50	98.50	6.62	2.19	0.38	1.02	
Howard	45.07	16.92	37063.75	5.20	0.49	4.55	0.56	99.44	7.59	0.24	0.36	1.01	
Rice	44.12	16.62	39213.34	5.14	0.40	4.57	1.89	98.11	7.07	2.26	0.34	1.02	
Cornell	53.11	18.51	219198.40	5.91	0.54	5.27	0.15	99.85	8.87	3.07	0.35	1.00	
Michigan	55.76	19.16	249417.49	6.04	0.41	5.55	0.03	99.97	9.40	3.66	0.36	1.00	

Table 5.3: Time to reach certain fractions of adopted nodes in the WTM model with homogeneous and heterogeneous timers on FACEBOOK100 networks. The quantity N is the number of nodes in a network, z is its mean degree, and $\langle l \rangle$ is its mean shortest-path length between pairs of nodes. All nodes have a homogeneous adoption threshold of $\phi = 0.01$. The notation x_{hom} indicates that we calculate the quantity x for homogeneous timers, the notation x_{unif} indicates that we calculate the quantity x for timers that are distributed uniformly at random, and the notation x_{Gam} indicates that we calculate the quantity x for timers determined using a Gamma distribution and then rounded down to an integer. The quantity T is the time that it takes to reach a steady state, t_{ρ^*} is the time that it takes for the fraction ρ^* of adopted nodes to reach ρ^* . Each quantity is a mean over 100 realizations with a different distribution of timers for each realization, and we also choose the seed node uniformly at random for each realization.

				Homogeneous timer					
	N	z	$\langle l \rangle$	T_{hom}	$t_{0.9,\text{hom}}$	$t_{0.8,\text{hom}}$	$t_{0.7,\text{hom}}$	$t_{0.6,\text{hom}}$	$t_{0.5,\text{hom}}$
Caltech	762	43	1.54	21.95	15.60	14.75	14.01	12.91	12.07
Reed	962	39	1.62	23.40	16.19	15.22	14.40	13.44	12.55
Haverford	1446	82	1.50	25.83	15.38	14.96	14.51	13.71	13.17
Simmons	1510	43	2.57	25.34	16.90	15.71	15.27	14.44	13.69
Swarthmore	1657	73	2.32	24.35	15.91	15.17	14.69	14.20	13.37
Howard	4047	101	1.59	27.32	18.29	16.91	16.16	15.71	15.31
Rice	4083	90	1.65	25.05	18.01	16.62	15.79	15.32	15.09
Cornell	18621	84	1.91	32.40	20.85	20.22	19.63	18.45	17.66
Michigan	30106	78	2.04	35.76	21.52	20.58	19.86	19.53	19.04
				Timers distributed uniformly at random					
	N	z	$\langle l \rangle$	T_{unif}	$t_{0.9,\text{unif}}$	$t_{0.8,\text{unif}}$	$t_{0.7,\text{unif}}$	$t_{0.6,\text{unif}}$	$t_{0.5,\text{unif}}$
Caltech	762	43	1.54	21.24	11.84	10.69	9.72	8.80	7.86
Reed	962	39	1.62	22.0	12.08	10.92	9.96	9.03	8.10
Haverford	1446	82	1.50	22.36	11.45	10.43	9.50	8.59	7.68
Simmons	1510	43	2.57	23.47	12.22	11.07	10.11	9.18	8.26
Swarthmore	1657	73	2.32	21.01	11.61	10.58	9.65	8.73	7.82
Howard	4047	101	1.59	25.39	13.18	12.13	11.18	10.26	9.34
Rice	4083	90	1.65	22.43	12.61	11.58	10.64	9.72	8.80

Table 5.3: (Continued from the previous page.)

	N	z	$\langle l \rangle$	Gamma distribution (and then rounding down to an integer) of timers						
				T_{Gam}	$t_{0,9,\text{Gam}}$	$t_{0,8,\text{Gam}}$	$t_{0,7,\text{Gam}}$	$t_{0,6,\text{Gam}}$	$t_{0,5,\text{Gam}}$	
Cornell	18621	84	1.91	30.11	13.94	12.86	11.90	10.98	10.06	
Michigan	30106	78	2.04	31.24	14.49	13.40	12.44	11.52	10.60	
Caltech	762	43	1.54	37.04	15.58	13.20	11.42	10.15	9.15	
Reed	962	39	1.62	37.33	15.58	12.63	10.90	9.66	8.66	
Haverford	1446	82	1.50	38.44	15.12	12.28	10.62	9.44	8.50	
Simmons	1510	43	2.57	39.92	15.95	13.02	11.30	10.06	9.07	
Swarthmore	1657	73	2.32	38.63	15.27	12.45	10.79	9.60	8.68	
Howard	4047	101	1.59	45.07	16.92	14.02	12.33	11.12	10.17	
Rice	4083	90	1.65	44.12	16.62	13.75	12.06	10.85	9.90	
Cornell	18621	84	1.91	53.11	18.51	15.60	13.89	12.67	11.70	
Michigan	30106	78	2.04	55.76	19.16	16.26	14.55	13.32	12.35	

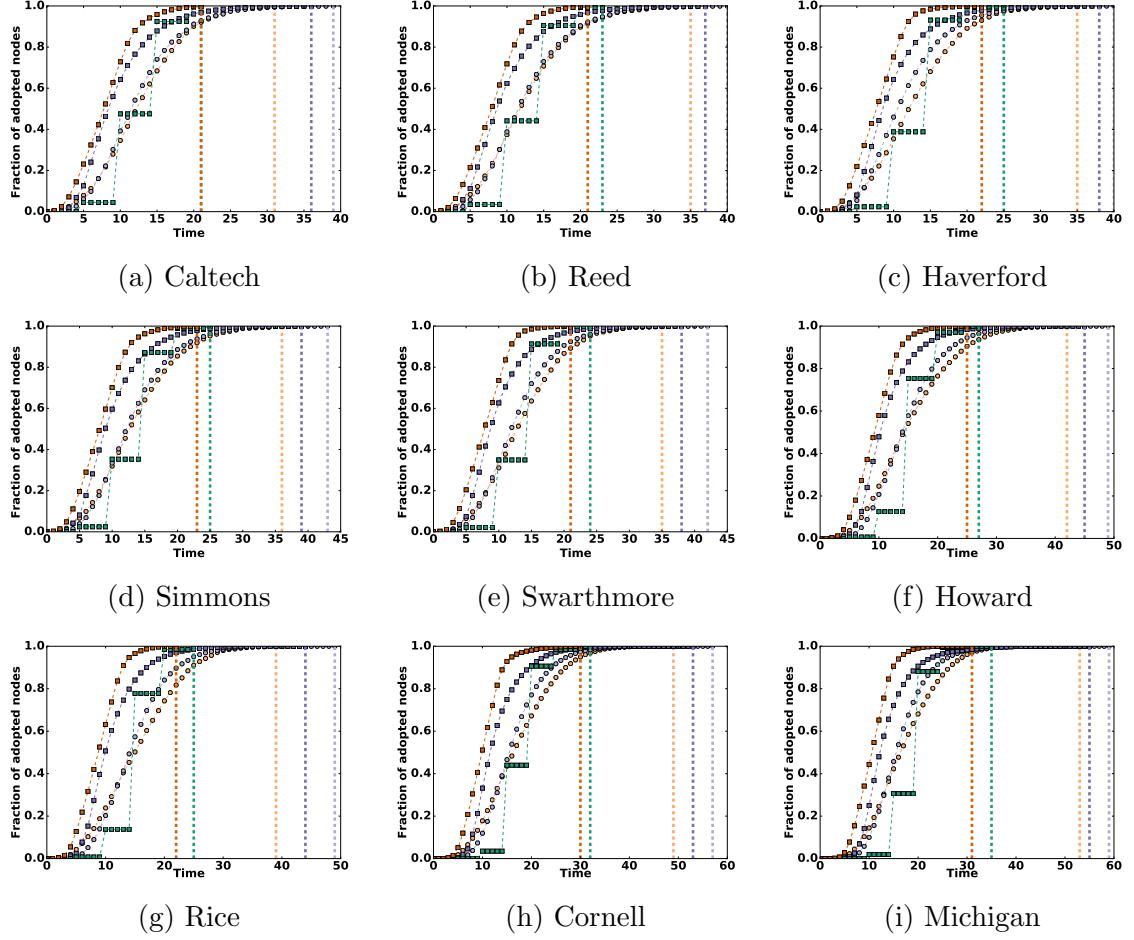


Figure 5.3: Adoption curves of the WTM with timers on FACEBOOK100 networks and adoption curves after we adjust the timers of stem nodes of dissemination trees. The green curve is for the WTM with homogeneous timers with $\tau = 4$, the orange curves are for the WTM with heterogeneous timers τ that are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$, and the blue curves are for heterogeneous timers determined from the Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and then rounding down to an integer. To isolate the effects of different distributions of timers, in each case, we run the WTM using the same networks with the same seed nodes. The dark orange and blue curves are before we change the timers in dissemination trees, and the corresponding light-coloured curves are after we change those timer values (see the main manuscript). The squares (for the dark curves) and disks (for the light curves) are the results of numerical simulations, and the dashed lines mark the times at which the adoption process of the corresponding colour reaches a steady state. Our numerical results are means over 1,000 realizations. For each realization, we generate an independent network. We also determine the seed node (uniformly at random) and timer values separately for each realization.

As in Section 3.3.2.2, we change the timers of stem nodes of dissemination trees to the mean value $\langle \tau \rangle$ of the timers (which is $\mu_\tau = 4$ in our example). The times that

the stem nodes' neighbours in a tree adopt then also change, as the adoption time of a node in an adoption path is determined by the sum of its timer and the adoption time of its predecessor node. (Note that we are not rerunning the WTM dynamics.)

In Fig. 5.3, we investigate how the adoption curves change because of the change of timers of stem nodes. In each figure, the orange and blue curves both represent adoption curves for the WTM with heterogeneous timers. For the orange curve, timers are distributed uniformly at random from the set $\{0, 1, \dots, 8\}$; for the blue curve, timers are determined from a Gamma distribution with mean $\mu_\tau = 4$ and standard deviation $\sigma_\tau = 4$ and are then rounded down to an integer. The curves with lighter colours are the adoption curves after we change the timers of stem nodes of dissemination trees. The light orange curve is for uniformly-randomly-distributed timers, and the light blue curve is for Gamma-distributed timers. The green curve, which we include for comparison, is for a WTM with a homogeneous timer. We observe that the light-coloured curves intersect the green curve earlier than do the dark-coloured curves, similarly with the result for the WTM with timers on random-network models in Chapter 3; therefore, we can generalise our observation of the stems in random networks to the stems in the five FACEBOOK100 networks that the stems play important role for accelerating the spread of adoption to the majority of nodes, although stem nodes constitute fewer than branch nodes in the networks.

5.3 Real-world data analysis

Up until now, our work has been on undirected networks; in Chapter 3 and 4, we used random-network models where a pair of nodes is chosen uniformly at random and then the nodes become adjacent via an undirected edge, and in Section 5.1 and 5.2 we used FACEBOOK100 networks where a friendship is indicated by an undirected and unweighted edge. However, in many real-world networks, the edges are directed to express the unidirectional property of relationships between nodes. In addition, the spread of information on such networks is often constrained by the direction of edges; for example, an adoption spreads following the direction of an edge, not against it³. In this section, we use real-world networks that have directed edges, and we propose a measure to calculate the threshold and the timer of a node in the directed networks.

³In a directed network, a pair of nodes can have a bidirectional relationship if the pair is connected via a reciprocal edge — two directed edges pointing towards each other. Because an adoption can spread in either direction on a reciprocal edge, in a dynamical process on networks a reciprocal edge behaves similar to an undirected edge. Therefore, for a dynamical process on networks, an undirected network can be considered as a directed network where all edges are reciprocated [20, 58].

On directed networks, we estimate the thresholds of nodes in the networks based on the following mechanism: given the data on directed networks and the data on the times of adoptions, we calculate the threshold ϕ_{v_i} of a node v_i as the fraction of adopted out-neighbours of v_i by the time v_i adopts [58, 134]. Mathematically, we can express as follows:

$$\phi_{v_i} = \sum_{v_j \in \Gamma_{\text{out}}(v_i)} \frac{s_{v_j}(t_{v_i})}{|\Gamma_{\text{out}}(v_i)|}, \quad (5.1)$$

where t_{v_i} is the time of adoption of node v_i , the adoption state of node v_j at t_{v_i} is $s_{v_j}(t_{v_i}) \in \{0, 1\}$, the quantity $\Gamma_{\text{out}}(v_i)$ is the set of node v_i 's out-neighbours, and $|\Gamma_{\text{out}}(v_i)|$ is the size of the set $\Gamma_{\text{out}}(v_i)$. Accordingly, we calculate the timer τ_{v_i} as the difference between t_{v_i} and the largest t_{v_j} that is smaller than t_{v_i} , where $v_j \in \Gamma_{\text{out}}(v_i)$, and then we subtract 1 because it takes one synchronous time step to update a node:

$$\tau_{v_i} = t_{v_i} - \max_{\substack{v_j \in \Gamma_{\text{out}}(v_i) \\ t_{v_j} < t_{v_i}}} t_{v_j} - 1. \quad (5.2)$$

It is important to note that this way of calculating a node's threshold is subject to overestimating the actual threshold of the node, which may lead to a systematic upward bias in threshold estimation [12]. See [12] for more details. More in-depth research is required to reduce the error that occurs from threshold measurement; however, we do not investigate this issue any further, because it is not our main interest in this thesis.

5.3.1 Three small real-world networks and their dissemination trees

To investigate the dissemination trees based on the real-world data, we use three small real-world networks, for which we possess data on all nodes' times of adoptions. The three small networks that we use are from (i) the study of the medical innovation dissemination among physicians by Coleman et al. [30], (ii) the study of Brazilian farmers' adoption of hybrid seed corn by Rogers et al. [121], and (iii) the study of the Korean women's adoption of family planning method by Park et al [108]. Coleman et al. studied 125 physicians who could have prescribed tetracycline, which was a powerful antibiotics introduced in the mid-1950s [136]. The network data were collected by asking physicians to name those with whom they most frequently interact either for discussion, friendship, or advice [30, 133]. The data on the times of adoptions were collected by examining when the physicians first prescribed tetracycline. Rogers et

al. studied the dissemination of hybrid corn among 692 Brazilian farmers⁴. The network data were collected by asking farmers to identify either their three best friends, the three most influential people in their community, the three most influential people with respect to farm innovations, or the best person to organise a cooperative project [121, 133]. The data on the times of adoptions were collected by asking farmers when they first planted hybrid corn. Park et al. investigated the dissemination of family planning methods among 1047 married women of child-bearing ages. The network data were collected by asking women to name who they sought advice about either family planning, general information, abortion, health, the purchase of consumer goods, or children’s education [108, 133]. The data on the times of adoptions were collected by asking respondents to state the year that they first started to use a modern family-planning method. All data that we use on the three networks and on the times of adoptions of nodes in the networks are from NETDIFFUSER, an R package developed by Vega Yon et al [136].

In each study of the three networks, a node specifically nominates people (or a person, if there is only one) who may have had influence on the node. Based on that, we can make a nominator be adjacent to its nominee with an outward directed edge. We thereby examine the three networks as directed networks. Note that the number of a node’s outward-adjacent neighbours (henceforth abbreviated as ‘out-neighbours’) and inward-adjacent neighbours (henceforth abbreviated as ‘in-neighbours’) can be different, because nomination need not be reciprocal.

We first investigate the distribution of node thresholds and timers. In Fig. 5.4, we show the cumulative distribution of node thresholds (red) and timers (blue) of the three networks. These distributions have some commonalities across the three networks. For example, all of them have a steep increase at $\phi = 0$. It means that the three networks have large seed fractions, which adopt without the influence of its neighbours. Additionally, in Fig. 5.4c and Fig. 5.4e, the cumulative distributions of thresholds increase steadily with the increase of threshold, except for the points where cumulative distributions have a steep increase (e.g., see $\phi = 0$ and 1 of Fig. 5.4c and $\phi = 0, 0.15, 0.30, 0.50$, and 1 of Fig. 5.4e). The cumulative distribution of thresholds for physicians has too few data points to analyze, but it still exhibits similar behaviour as the other cumulative distributions of thresholds. The adjacent data points are close to each other, except for the points where cumulative threshold distribution has

⁴Rogers et al. also investigated the dissemination of farm practices in Nigeria and India, but it was only in Brazil (and only for hybrid corn) that the dissemination of innovation reached more than a small proportion of the farmers [136].

a steep increase (see $\phi = 0$ and 1 of Fig. 5.4a). In Section 5.3.2, we investigate the distribution of thresholds (and timers) on large real-world networks with data on the time of adoption.

The cumulative distributions of node timers of the three networks also have some common characteristics. An obvious similarity between the cumulative distributions of the timers is that they are concave, with a decreasing slope as the timer increases. Therefore, with our calculation method, the fraction of nodes decreases as a timer increases.

We now examine the stems and branches of the dissemination trees of the three networks. Unlike the dissemination trees of the random networks and FACEBOOK100 networks that we investigated in Section 3.3.2 and 5.2 respectively, the thresholds and the timers of nodes in the three networks are not drawn from certain distributions, but instead are calculated from empirical data on the times of adoptions of nodes [see (5.1) and (5.2)]. Therefore, unlike the random networks and FACEBOOK100 networks, the thresholds are not necessarily homogeneous for the three networks.

We calculate the stems and branches only of the component ζ with the largest number of adopters at steady state, because our purpose of investigating stems and branches is to examine the role of stem nodes and branch nodes in disseminating adoptions to the largest population in a network. Stems and branches of adoption spreading in other components that are disconnected from ζ have nothing to do with the adoption spreading in ζ , and therefore we pick a component ζ with the largest number of adopters and investigate the stems and branches of ζ .

For each of the three networks in Table 5.4, we find some quantities qualitatively similar to the quantities for the random-graph ensembles and FACEBOOK100 networks. We find that the mean percentage f_{a_s} of stems is smaller than the mean percentage f_{a_b} of branches. Additionally, the mean timer τ_s of stem nodes is smaller than the mean timer τ_b of branch nodes, so stems tend to grow faster and become longer than branches ($l_s > l_b$).

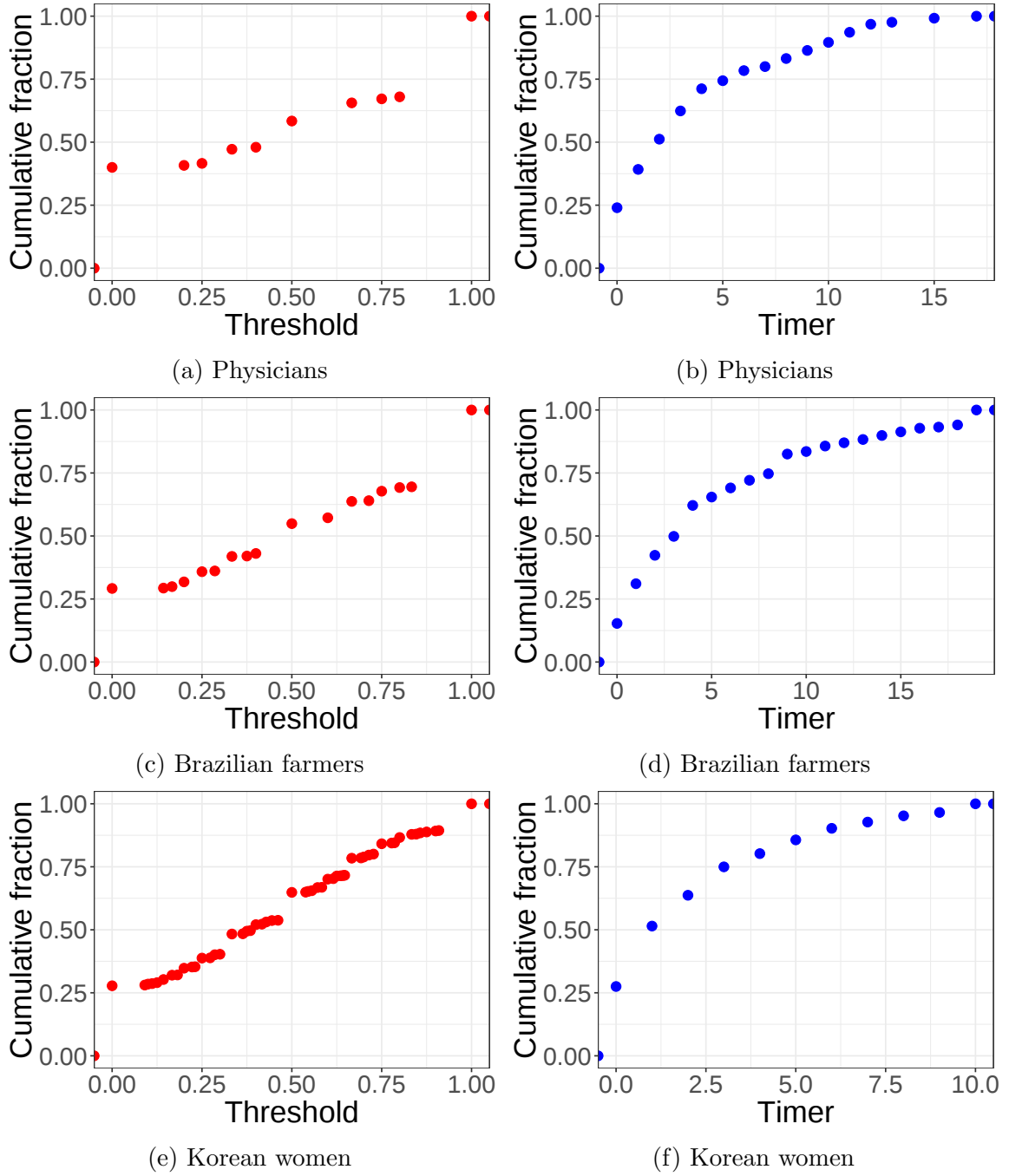


Figure 5.4: Cumulative distributions of threshold (red) and timer (blue) of (a,b) the physicians network, (c,d) the Brazilian farmers network, and (e,f) Korean women network. We calculate thresholds as the adopted fraction of out-neighbours, and we calculate timers as the time difference between a node and its most recently adopted out-neighbours. Data on the networks and the times of adoptions of nodes are from an R package, called `NETDIFFUSER`, developed by Vega Yon et al [136].

Table 5.4: Comparison between the characteristics of stems and branches of the physicians network, the Brazilian farmers network, and the Korean women network. The quantity N_ζ is the number of nodes in the component ζ with the most adopters at steady state, z_ζ is the mean degree of the component ζ , and $\langle l \rangle$ is the mean shortest-path length between pairs of nodes. The quantity f_{a_s} is the mean percentage of stems among adoption paths, f_{a_b} is the mean percentage of branches among adoption paths, l_s is the mean stem length, l_b is the mean branch length, τ_s is the mean timer value of stem nodes, and τ_b is the mean timer of branch nodes. The quantity z_s is the mean degree of stem nodes, z_b is the mean degree of branch nodes, g_s is the mean eigenvector centrality of stem nodes, g_b is the mean eigenvector centrality of branch nodes, b_s is the mean betweenness centrality of stem nodes, and b_b is the mean betweenness centrality of branch nodes.

	N_ζ	z_ζ	$\langle l \rangle$	$f_{a_s}(\%)$	$f_{a_b}(\%)$	$\frac{\tau_s}{\langle \tau \rangle}$	$\frac{\tau_b}{\langle \tau \rangle}$	ϕ_s	ϕ_b	$\langle \phi \rangle$	z_s	z_b	g_s	g_b	b_s	b_b
Physicians	60	2.67	4.00	0.05	0.95	0.47	1.11	0.58	0.55	0.551	3.20	2.56	0.14	0.12	46.77	103.53
Brazilian farmers	82	2.98	3.09	0.02	0.98	0.61	1.04	0.50	0.57	0.564	4.00	2.88	0.46	0.03	244.30	28.84
Korean women	56	4.79	2.93	0.05	0.95	0.24	1.11	0.35	0.40	0.394	8.00	4.33	0.30	0.20	246.92	73.28

Because the thresholds are heterogeneous in these empirical networks, we compare the mean threshold ϕ_s of the stem nodes and the mean threshold ϕ_b of the branch nodes. We find that $\phi_s < \phi_b$ for Brazilian farmers and Korean women networks, whereas $\phi_s > \phi_b$ for the physicians network. Considering that the stem nodes initiate the spread of adoption in the branches to which each of the stem nodes is connected, one can construe stem nodes as locally early adopters. In that respect, some may consider it more natural for early adopters to have a small mean threshold and small mean timer, but our calculation of the stem nodes of the three networks in Table 5.4 indicates that the stem nodes of the three networks can have a larger mean threshold than the branch nodes, although the mean timers of stem nodes is smaller than those of branches.

Let us use subscript $_{\text{Phy}}$ to indicate the Physicians' network, $_{\text{Bra}}$ to indicate the Brazilian farmers' network, and $_{\text{Kor}}$ to indicate the Korean women network. Then, the ratios $\frac{\tau_s}{\langle \tau \rangle}$ of the mean timer of stem node to the mean timer of all nodes of the three networks follow $\frac{\tau_s}{\langle \tau \rangle}_{\text{Kor}} < \frac{\tau_s}{\langle \tau \rangle}_{\text{Phy}} < \frac{\tau_s}{\langle \tau \rangle}_{\text{Bra}}$, and the mean thresholds of the three networks follow $\langle \phi \rangle_{\text{Kor}} < \langle \phi \rangle_{\text{Phy}} < \langle \phi \rangle_{\text{Bra}}$. As we discussed in Section 3.3.3, in networks with a homogeneous threshold and heterogeneous timers, stem nodes tend to have large mean timer if we use large mean threshold. We speculate that we can generalise this relationship between the mean timer of stem nodes and the mean threshold of all nodes to real-world networks, even when nodes have heterogeneous thresholds, as our results show in Table 5.4. In future work, we will use several real-world data sets on the spread of adoption for a more thorough investigation of the relationship between the ratio $\frac{\tau_s}{\langle \tau \rangle}$ and the mean threshold $\langle \phi \rangle$.

We also calculate structural characteristics of stem nodes and branch nodes using centrality measures. Centrality measures are often employed to determine dynamical, as well as structural, importance of nodes in networks [78, 92, 93, 115]. For our work, we measure the mean degree centrality (i.e., degree) z_s of stem nodes, the mean degree centrality z_b of branch nodes, the mean eigenvector centrality g_s of stem nodes, the mean eigenvector centrality g_b of branch nodes, the mean betweenness centrality b_s of stem nodes, and the mean betweenness centrality b_b of branch nodes. As we can see in Table 5.4, all three networks have $z_s > z_b$ and $g_s > g_b$, so stem nodes are located at a structurally central point according to degree centrality and eigenvector centrality. However, the mean betweenness centrality b_s of stems is smaller than b_b in the physicians networks, whereas b_s is larger than b_b for the Brazilian farmers network and the Korean women network. The results imply that eigenvector centrality and degree centrality — both of which are to do with how well a node is connected (see

Section 2.1.6) — capture the importance of stem nodes, which play the central role in fast spread of adoption. However, betweenness centrality does not show consistent results for determining the stem nodes.

As we pointed out at the beginning of this section, the physicians network, Brazilian farmers network, and Korean women network are all small. In Section 5.3.2, we use larger networks to calculate thresholds and timers using (5.1) and (5.2). We examine the distribution of thresholds and timers on large real-world networks with data on the times of adoptions in Section 5.3.2.

5.3.2 The distribution of thresholds and timers in Twitter network

Borge-Holthoefer et al. [20] investigated rich quantitative data on The Spanish May 15th Movement, and much work has been done to study the data from various angles [18, 19, 58, 59]. The Spanish May 15th Movement, which is often referred to as 15-M movement (in Spanish, *movimiento 15-M* [1]), is the anti-austerity movement in which people of Spain challenged the rigid, top-down, party driven political system of Spain since 1978 [9]. The notable characteristic of the movement is that the movement was organised using social media such as Facebook and Twitter. Borge-Holthoefer et al. studied the data that includes 85,851 users who publicly exchanged 581,749 messages (tweets) from April 25th, 2011 (at 00:03:26) to May 26th, 2011 (at 23:59:55) through an online social networking service called *Twitter* (<http://www.twitter.com>). The data records, during one month, which user mentions which user (or users, if more than one) in which tweet at what time. With the data, we can construct a directed network in which an edge goes from a user (source) who mentions to users (targets) who get mentioned in a tweet [20]. It is important to note that this way of constructing a network only gives a dynamical instance of a larger underlying network that also includes (i) users who did not tweet during the given period of time and (ii) users who did not mention their neighbours. Also, we assume that the time when a user tweets for the first time is the user's time of adoption. Based on the data, we calculate the thresholds and timers of the users (nodes) of the network of the 15-M movement (henceforth abbreviated as "15M network").

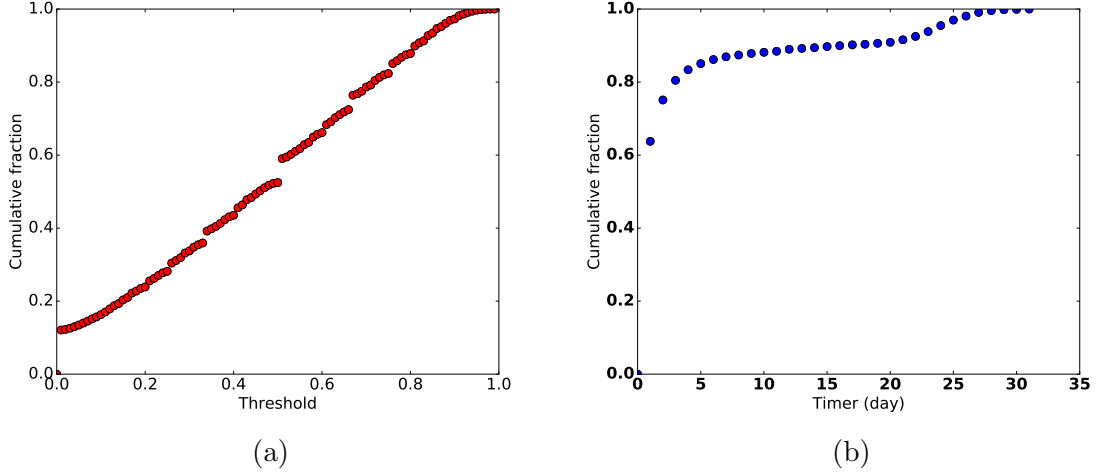


Figure 5.5: Cumulative distribution of threshold (red) and timer distribution (blue) of the 15M network. We use 0.01 as the bin width of the threshold distribution, and we use 1 day (i.e., 86,400 seconds) as the bin width of the timer distribution. We calculate thresholds as the adopted fraction of out-neighbours, and we calculate timers as the time difference between a node and its most recently adopted out-neighbours minus 1.

We first investigate the cumulative distribution of thresholds and timers of nodes in the 15M network. We calculate thresholds using (5.1), and timers using (5.2). Similar to the cumulative distributions of thresholds and timers of the three networks that we calculated in Section 5.3.1, the cumulative threshold distribution has a steep increase at $\phi = 0$, which we interpret to mean the networks has a large seed fractions of nodes that adopt without influence from their neighbours. Additionally, similar to the Brazilian farmers network (see Fig. 5.4c) and Korean women network (see Fig. 5.4e), the cumulative distribution of thresholds increases steadily except for the points at which cumulative distributions have a steep increase (e.g., see $\phi = 0, 0.26, 0.34, 0.55, 0.66$, and 0.77 of Fig. 5.5a).

The cumulative distribution of node timers of 15M network also shares some common characteristics with the cumulative distributions of the three networks in Fig. 5.4. An obvious similarity is that the cumulative distributions are concave curve of which the slope decreases as timer increases. Therefore, with our calculation method, the fraction of nodes decreases as a timer increases.

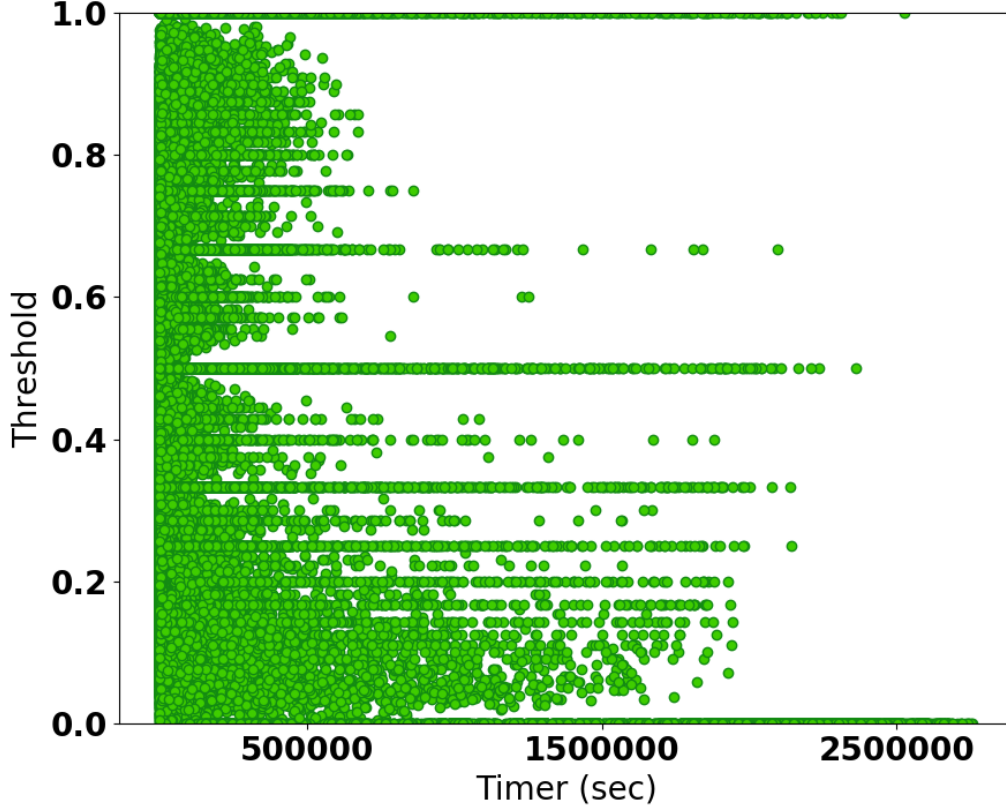


Figure 5.6: Scatter plot of thresholds and timers of nodes in the 15M network. Each green data point corresponds to each node in the network, and each data point is positioned along the vertical axis by its threshold, and along the horizontal axis by its timers. We calculate thresholds as the adopted fraction of out-neighbours, and we calculate timers as the time difference (in the units of second) between a node and its most recently adopted out-neighbours minus 1.

Fig. 5.6 shows the scatter plot of thresholds and timers of nodes in the 15M network. Each data point corresponds to each node in the network, and each data point is positioned by its threshold and timer, which are obtained by (5.1) and (5.2), respectively. One can see there is no correlation between thresholds and timers, however one may find a correlation between the two if one uses different ways of calculating thresholds and timers. Also, note that there is a strong bias towards the threshold $\phi = 0$, because we consider all nodes that adopt without influence from their neighbours as seeds, although some of them may not be seeds if they were influenced by their neighbours via social networks other than Twitter. Employing multiplex network [77,83] may provide a sensible way to model the influence of neighbours from

multiple social networks.

We noted in the beginning of Section 5.3 that our measurement of thresholds and timers is one of many possible ways to measure them. In that respect, the distribution of threshold and timer is subject to the ways of calculating threshold and timer. For example, in equation (5.1), we calculated the threshold of a node as the fraction of out-neighbours that adopted before the node adopts. This is because we assumed that the last neighbour to adopt makes the node adopt definitively. However, the node may have determined to adopt before the last neighbour to adopt adopts; it could be the second from the last neighbour to adopt, or any x^{th} from the last neighbour to adopt where $x \in \{0, 1, 2, \dots\}$. Accordingly, we can generalise the calculation of the threshold and timer of v_i as follows:

$$\phi_{v_i} = \sum_{v_j \in \Gamma_{\text{out}}(v_i)} \frac{s_{v_j}(t_x)}{|\Gamma_{\text{out}}(v_i)|}, \quad (5.3)$$

$$\tau_{v_i} = t_{v_i} - t_x - 1, \quad (5.4)$$

where t_x is the time of adoption of x^{th} from the last neighbour to adopt.

In Fig. 5.7, we show how the cumulative distributions of thresholds and timers change by increasing x from 0 to 10 and then to 100. The cumulative threshold distribution curve moves upward with the increase of x , because the number of a node's neighbours that must adopt before the node adopts decreases with the increase of x . Consequently, the cumulative distribution curve for threshold has a steeper increase at $\phi = 0$. However, the points where the cumulative threshold distribution curve has a steep increase for $x = 0$ (e.g., see $\phi = 0, 0.26, 0.34, 0.55, 0.66$, and 0.77 of the light green curve in Fig. 5.7a) has a shallower increase for $x = 10$ and $x = 100$. In contrast to the cumulative threshold distribution, the cumulative timer distribution curve moves downward with the increase of x , because a node must wait a longer time for a larger x . Consequently, the cumulative timer distribution curve has a steep increase at a larger timer value for a larger x .

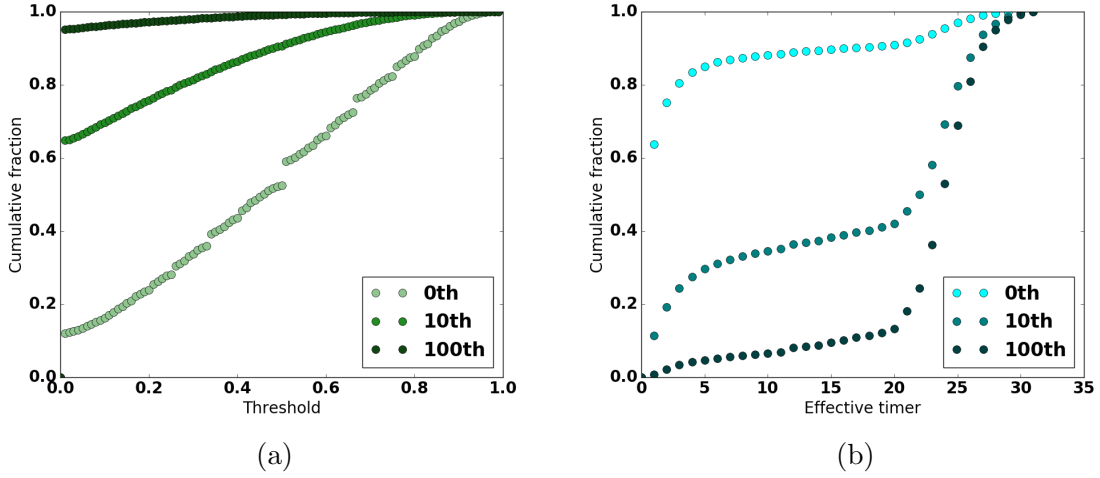


Figure 5.7: Change of the cumulative distributions of (a) thresholds, and (b) timers for the 15M network. We assume that a node decides to adopt when the x^{th} from the last out-neighbour to adopt adopts, where $x \in \{0, 1, 2, \dots\}$, and we calculate thresholds and timers according to (5.3) and (5.4). We increase x from 0 to 10 and then to 100, and we investigate the ensuing changes in the distributions of thresholds and timers. We determine the time that a node as the bin width of threshold distribution is 0.01, and the bin width of timer distribution is 1 day, which corresponds to 86,400 seconds.

5.4 Summary

In this chapter, we used the WTM with timers to analyse real-world data sets. In Section 5.1, we investigated the WTM with timers on five FACEBOOK100 networks, and we found that the different structural characteristics of the networks give adoption curves that have different shapes from the adoption curves for the random networks that we studied in Chapter 3. However, similar to the random networks, the stems of the five FACEBOOK100 networks play an important role for spreading adoptions to the majority of nodes in networks faster for heterogeneous timers than for a homogeneous timer (see Section 5.2). Lastly, in Section 5.3, we proposed a way to calculate thresholds and timers from real-world data on the spread of adoptions, and we investigated the cumulative distributions of thresholds and timers. We found some similarities between the cumulative distributions of thresholds and timers across the different data sets that we used for our study.

Chapter 6

Summary and Conclusion

We started our thesis in Chapter 1 with a brief introduction to the study of networks, and how and why mathematical modelling is useful for studying social networks. We went through mathematical basics that are key to understand the thesis in Chapter 2, including graph theory, Erdős-Rényi random graph model, probability generating function, site-percolation, and Watts cascade model. In Chapter 3, we introduced a timer model, and investigated how the delay of adoption of individuals can affect the spread of adoptions in a network. We explored this idea in Section 3.1.1 by augmenting the Watts threshold model (WTM) with timers on the nodes, and we examined how incorporating homogeneous and heterogeneous timers can affect the dynamics of contagions on small networks (See Section 3.2), and on large networks (See Section 3.3). In particular, we investigated adoption paths — the paths through which adoptions spread — in Section 3.1.2 to study the different behaviours between a homogeneous and heterogeneous timers. We found that for the WTM with a homogeneous timer, neither the adoption order of nodes nor adoption paths change, because a homogeneous timer simply delays the adoption of each individual by the same amount of time. However, heterogeneous timers can change both the adoption order of nodes and adoption paths, and the precise effects depend on both network structure and timer distribution. For example, in Section 3.3.1, we observed that incorporating heterogeneous timers into the WTM can either accelerate or decelerate the time to a steady state in comparison to incorporating a homogeneous timer. However, in our calculations using two different types of heterogeneous timer distributions (specifically, uniformly random timers and timers determined using a Gamma distribution and then rounding down to an integer), we found that the majority of nodes (up to 90% in our study) in networks adopt earlier with heterogeneous timers than with homogeneous ones.

To understand how the majority of nodes in networks adopt earlier with heterogeneous timers, in Section 3.3.2, we examine the adoption paths of the WTM with a homogeneous timer and heterogeneous timers. Specifically, we investigate stems (i.e., the longest adoption paths) and branches (i.e., the rest of adoption paths other than stems) in Section 3.3.2.1, with which we can construct a dissemination tree (i.e., trees that are constructed based on the adoption paths of the WTM with timers on networks) as we show in Section 3.3.2.2. As a result, we speculate that stems play a significant role in spreading adoptions faster to the majority of nodes in networks for the WTM with heterogeneous timers than for the WTM with a homogeneous timer. To support our speculation, we examined the delay of adoption processes when changing the timers of stem nodes to the mean value of timers, and we found that adoption processes are delayed even though these nodes constitute fewer than 1% of the adopted nodes in the examined networks.

We show analytics of the WTM with timers on networks in Section 4. We use a pair approximation that was first developed by Gleeson and Cahalane [51, 52], which was motivated by the analytical model for the single-spin flip dynamics of the random-field Ising model on a Bethe lattice at zero temperature (See Section 4.1). Pair approximation for the WTM shows good agreement with the numerical computations for the temporal evolution of the fraction of adopted nodes, as we show in Section 4.2. We modify the approximation for the WTM with timers in Section 4.3, and we found that it also agrees well with the numerical results for the WTM with timers on ER networks and a configuration model with Poisson-degree distribution. Because the pair approximation assumes that a network is locally tree-like, the discrepancy between the approximation and numerical computation is large for networks with cliques. Using pair approximation, we derive the cascade condition for the WTM with timers in Section 4.3.1, and we found that the cascade condition we obtained using generating function in Section 2.3 is a special case of the cascade condition we obtained using the pair approximation.

In Chapter 5, we study the application of the WTM with timers to real-world data. We ran the WTM with timers on five FACEBOOK100 networks in Section 5.1 to study the adoption curves both numerically and analytically. Because FACEBOOK100 networks do not have locally tree-like structure, the discrepancies between the numerical curves and analytical curves are large. Also, the growth of adoption curves for the five FACEBOOK100 networks is different from the growth of adoption curves for the random-graph ensembles, because of the different degree distributions between the FACEBOOK100 networks and random-graph ensembles. Based on the

adoption process of the WTM with timers on FACEBOOK100 networks, we examined the dissemination trees in Section 5.2. As with the dissemination trees of random-graph ensembles, the dissemination trees of the five FACEBOOK100 networks have small number of stem nodes with smaller timers than the timers of branch nodes, from which we can make the same speculation as the random-graph ensembles that stems play a significant role in spreading adoptions faster to the majority of nodes in FACEBOOK100 networks for the WTM with heterogeneous timers than for the WTM with a homogeneous timer.

In Section 5.3, we suggest a way to calculate threshold and timer from real-world data on the dissemination of information. In particular, in Section 5.3.1, we use three well-known data sets on the spread of innovations, and we use data on Twitter dynamics related to the May 15th movement in Spain in Section 5.3.2. We found that the cumulative distributions of thresholds across different data show similar characteristics; the cumulative distributions of thresholds show steep increases at certain thresholds, and all data show large seed fraction. The cumulative distributions of timers of all data show concave curves, which implies that as timer increases, the fraction of nodes with the timer decreases.

Throughout this thesis, we studied the idea of laziness by augmenting a timer with the WTM, and investigated how it leads to the change of the spreading dynamics of the WTM on networks: particularly, on social networks. There are many social influence models that represent how people spread information on social networks, but by a different mechanism than the WTM. Therefore, we cannot simply generalise our findings in this thesis to any social influence model with a timer, although our study of the WTM with timers may provide insights into a social influence model with a timer.

We also note that there is more than one way to augment a timer with a dynamical process on networks, and we studied only one way of augmenting a timer: a node waits for constant time steps after the node meets an adoption condition, and the change of surroundings — such as change of neighbour state — during the waiting time does not affect the node’s adoption. However, there may be several other ways to implement a timer, which can give rise to a dynamical model with completely different features.

In future work, we seek to investigate real-world data for the spread of information, memes, innovations, misinformation, and other phenomena on networks using models that include latencies (i.e., timers) in addition to influence thresholds. People can be late adopters for a variety of reasons (e.g., hard to convince versus easy to convince but

lazy), and incorporating timers into contagion models should be helpful for exploring different mechanisms of late adoption and their possible consequences for forecasting virality. In these studies, it will be important to augment a variety of different spreading models with timers. The deterministic update rule in the WTM made it particularly suitable for a first foray into incorporating timers in spreading processes, and that is why we used it in the present thesis. It will be interesting to incorporate timers into stochastic spreading processes to examine how this changes spreading behaviour and virality forecasts.

On a broader picture, to better understand spreading dynamics on social networks, it is necessary to study what makes social contagions distinctive to other contagions. There is a growing volume of publications that compares social contagions that are typically complex contagions, and biological contagions that are typically simple contagions [6, 25, 26, 40, 64, 144]. Further advances in this regard will not only enable us to develop more well-refined social contagion models, but also provide us with greater insight into social influence on social networks.

Appendix A

Time to steady state for networks with infinitely large loops

In Section 3.2, we saw that the mean time to steady state for the WTM with timers for a seed v_0 adjacent to a square loop is smaller for heterogeneous timers $\langle T_{\text{het}} \rangle$ than for a homogeneous timer T_{hom} . In this appendix, we show that for a seed adjacent to an infinitely large loop, $\langle T_{\text{het}} \rangle \geq T_{\text{hom}}$ for a loop with an odd number of nodes (henceforth called an “odd-sized” loop) and $\langle T_{\text{het}} \rangle \leq T_{\text{hom}}$ for a loop with an even number of nodes (henceforth called an even-sized loop).

In Figure A.1, we show a seed node v_0 adjacent to (a) an infinitely large odd-sized loop with nodes $v_i \in \{v_1, v_2, \dots, v_{2N+1}\}$ and (b) an infinitely large even-sized loop with nodes $v_j \in \{v_1, v_2, \dots, v_{2N+2}\}$. Assume that node v_1 of each loop connects a seed v_0 to the loop. For both cases, let us put nodes $v_2, v_3, \dots, v_{N+1}$ to the left of v_1 , and the rest of nodes $v_{N+2}, v_{N+3}, \dots, v_{2N+1}$ to the right of v_1 . We use X to denote a path $(v_1, v_2, \dots, v_{N+1})$, and we use Y to denote $(v_1, v_{N+2}, \dots, v_{2N+1})$. The length $|A|$ of path A and the length $|B|$ of B are the same as $N+1$ (i.e., $|A| = |B| = N+1$).

If we run the WTM with a homogeneous timer μ_τ , the time $T_{\text{odd,hom}}$ to steady state for the network with infinitely large odd-sized loop is $\mu_\tau(N+1) + N+1$, where the last $N+1$ comes from the fact that we use synchronous updating for the WTM with timers. If we run the WTM with heterogeneous timers with mean μ_τ , the rates of adoption spreading in A and B are approximately equal, because the time $T_{\text{odd,hom}}^A$ to spread an adoption through A approaches $\lim_{N \rightarrow \infty} \sum_{i=1}^{N+1} \tau_{v_i} + N+1 = \mu_\tau(N+1) + N+1$ in the $N \rightarrow \infty$ limit, and the time $T_{\text{odd,hom}}^B$ to spread an adoption through B also approaches $\mu_\tau(N+1) + N+1$ in the $N \rightarrow \infty$ limit, because $|A| = |B|$. Although both $T_{\text{odd,hom}}^A$ and $T_{\text{odd,hom}}^B$ approach $\mu_\tau(N+1) + N+1$ in the $N \rightarrow \infty$ limit, one of $T_{\text{odd,hom}}^A$ or $T_{\text{odd,hom}}^B$ may be larger than the other. Suppose that $T_{\text{odd,hom}}^A \geq T_{\text{odd,hom}}^B$; then, the spread of an adoption along path A determines the time to steady state for the WTM

with heterogeneous timers on the network that has an infinitely large odd-sized loop. Assuming that the mean μ_τ of the timers is the same between a homogeneous timer and heterogeneous timers, it follows that $jT_{\text{odd,het}}^A \geq T_{\text{odd,hom}} \geq T_{\text{odd,het}}^B$. Therefore, given that the mean of the heterogeneous timers is the same as a homogeneous timer, the time $T_{\text{odd,het}}$ to steady state for heterogeneous timers on the networks that has an infinitely large odd-sized loop is larger than or equal to the time $T_{\text{odd,hom}}$ to steady state for a homogeneous timer.

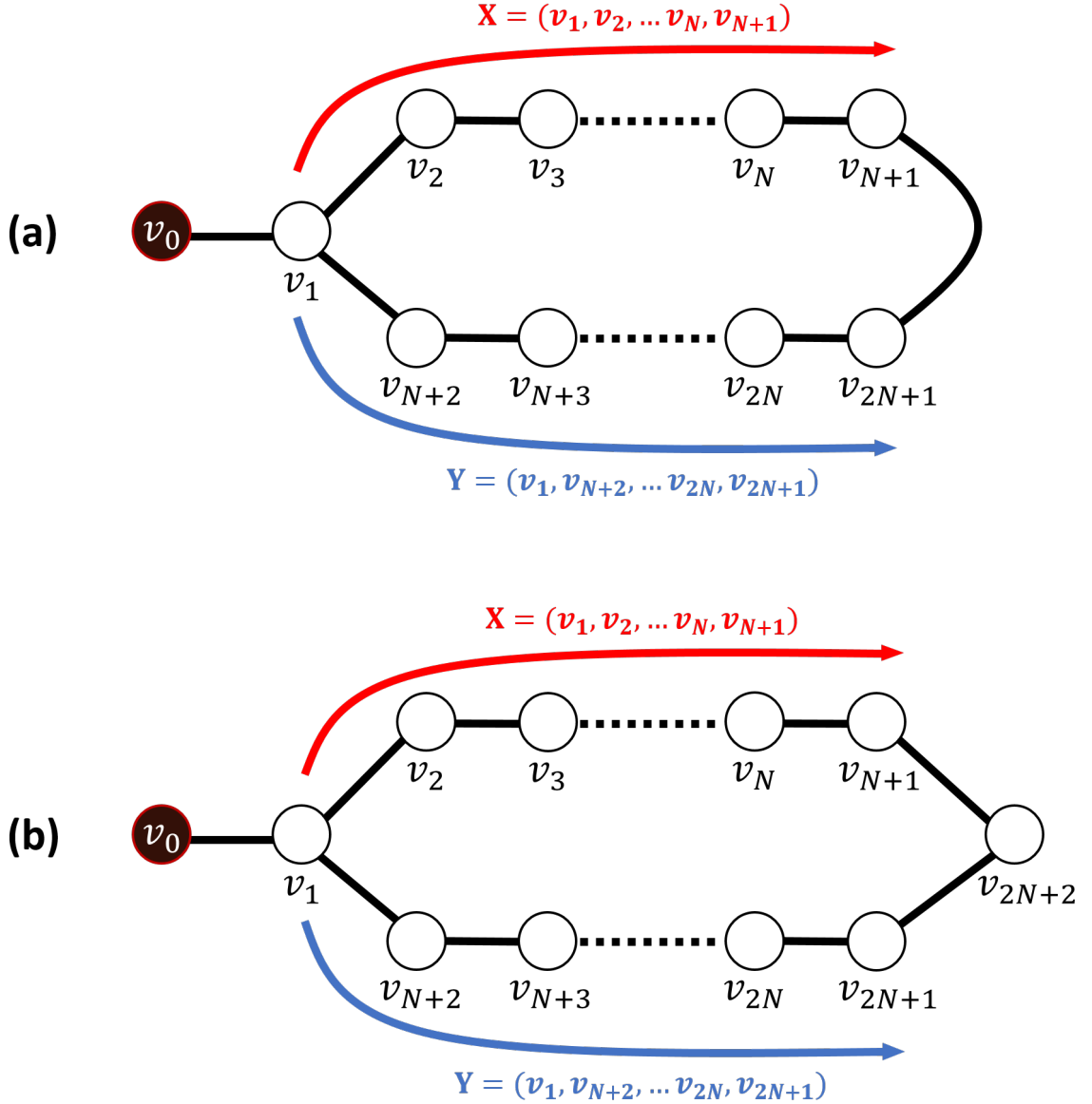


Figure A.1: Examples of networks in which a seed node v_0 is adjacent to (a) an infinitely large odd-sized loop with nodes $v_i \in \{v_1, v_2, \dots, v_{2N+1}\}$ and (b) an infinitely large even-sized loop with nodes $v_j \in \{v_1, v_2, \dots, v_{2N+2}\}$. We use X to denote a path $(v_1, v_2, \dots, v_{N+1})$, and we use Y to denote $(v_1, v_{N+2}, \dots, v_{2N+1})$.

For a network that has an infinitely large even-sized loop of size $2N + 2$, if we run the WTM with a homogeneous timer μ_τ , the time $T_{\text{even,hom}}$ to steady state is $\mu_\tau(N + 2) + N + 2$, where the last $N + 2$ comes from synchronously updating the nodes in paths A , B , and node v_{2N+2} . If we run the WTM with heterogeneous timers with mean μ_τ , the times $T_{\text{even,het}}^A$ and $T_{\text{even,het}}^B$ to spread an adoption through A and B approach $\mu_\tau(N + 1) + N + 1$ as $N \rightarrow \infty$, although one of $T_{\text{even,het}}^A$ or $T_{\text{even,het}}^B$ may be larger than the other. Suppose that $T_{\text{even,het}}^A \geq T_{\text{even,het}}^B$; then, the adoption path B triggers the timer of node v_{2N+2} , and it forms a longer adoption path $B + v_{2N+2}$. Between paths A and $B + v_{2N+2}$, the path that terminates later than (or equal to) the other determines the time $T_{\text{even,het}}$ to steady state. The mean $\langle \tau_{v_{2N+2}} \rangle$ of the timer of node v_{2N+2} is μ_τ , and therefore, the mean time $\langle T_{\text{even,het}}^{B+v_{2N+2}} \rangle$ to spread an adoption through $B + v_{2N+2}$ is $\mu_\tau(N + 2) + N + 2$, which is larger than $\langle T_{\text{even,het}}^A \rangle = \mu_\tau(N + 1) + N + 1$. Therefore, the spread of an adoption through $B + v_{2N+2}$ determines the time $T_{\text{even,het}}$ to steady state for the network with the infinitely large even-sized loop. Because we assumed that $T_{\text{even,het}}^B \leq T_{\text{even,het}}^A$, the mean time $\langle T_{\text{even,het}}^{B+v_{2N+2}} \rangle$ to spread an adoption through $B + v_{2N+2}$ is smaller than or equal to $T_{\text{even,hom}}$. Therefore, for a network with an infinitely large even-sized loop, the mean time to steady state is smaller or equal for heterogeneous timers than it is for a homogeneous timer.

Appendix B

Stems with timers larger than the mean

In this appendix, we discuss situations in which a stem in a network includes nodes with timers larger than the mean, in contrast to those that we investigated in Section 3.3.2. As we have seen, the characteristics of stems and branches can depend on both network structure and the timers. In Section 3.3.2, we focused on examining homogeneous adoption thresholds, such that nodes with degree equal to a network's mean degree meet the threshold adoption condition if even a single neighbour has adopted. In that situation, we observed that stem nodes tend to have mean timers that are smaller than the mean branch-node timers. However, depending on network structure (see Fig. B.1a) and the assignments of timers (see Fig. B.1b), it is possible for the mean stem-node timers to be larger than the mean branch-node timers. In Fig. B.1a, the adoption paths do not change, regardless of the assignments of thresholds and timers, because the seed node is not adjacent to any node in a cycle of length 4 or more (see Section 3.2). Therefore, in Fig. B.1a, the stems (v_3, v_4, v_5, v_6) and (v_7, v_8, v_9, v_{10}) have larger mean timers than the branches. (Note that the number of stems is larger than the number of branches, which is also different from our findings on large random networks in Tables 3.3 and 5.2.)

Even if a network includes a cycle with 4 or more nodes, the stem nodes can have a larger mean timer than the branch nodes, depending on network structure and timer assignments. With the given timer assignment, the network in Fig. B.1b has the stem $(v_1, v_6, v_5, v_7, v_8)$, which has a larger mean timer than the branch (v_2, v_3, v_4) .

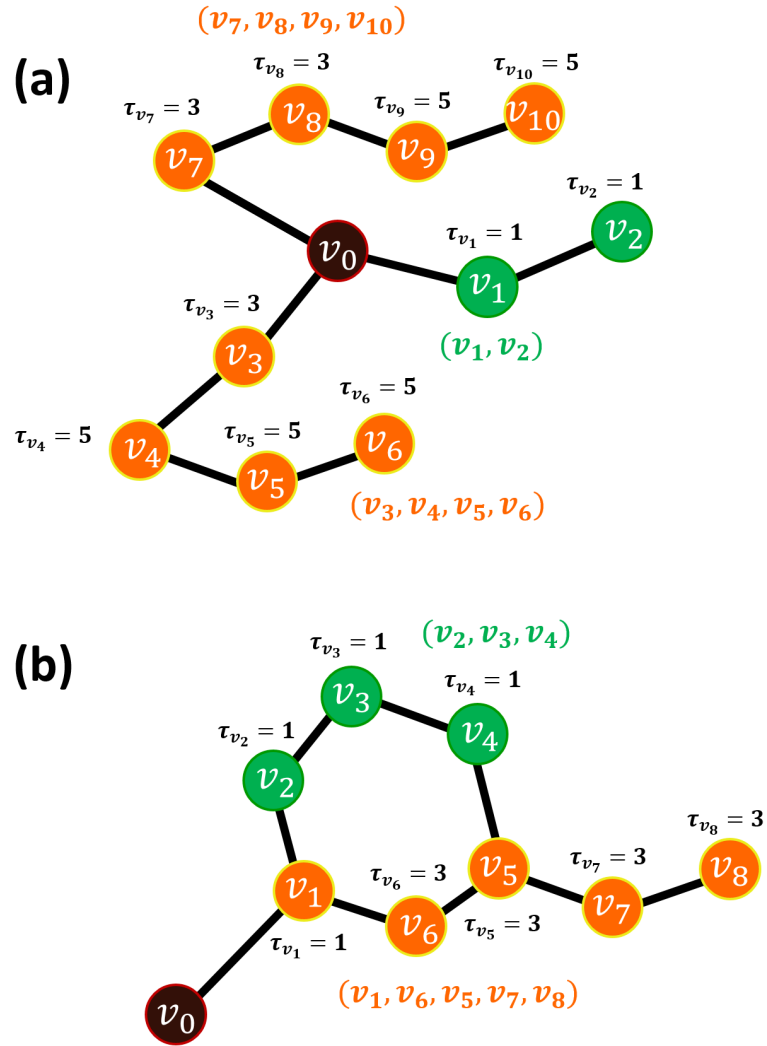


Figure B.1: Examples of networks with larger mean stem-node timers than mean branch-node timers.

Bibliography

- [1] Movimiento15m: Todo sobre el movimiento 15m. <http://www.movimiento15m.org/>. Accessed: 2017-09-30.
- [2] Eytan Bakshy, Itamar Rosenn, Cameron Marlow, and Lada Adamic. The role of social networks in information diffusion. In *Proceedings of the 21st International Conference on World Wide Web*, pages 519–528. ACM, 2012.
- [3] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.
- [4] Albert-László Barabási, Hawoong Jeong, Zoltan Néda, Erzsebet Ravasz, Andras Schubert, and Tamas Vicsek. Evolution of the social network of scientific collaborations. *Physica A: Statistical mechanics and its applications*, 311(3):590–614, 2002.
- [5] Albert-László Barabási and Zoltán N Oltvai. Network biology: understanding the cell’s functional organization. *Nature Reviews Genetics*, 5(2):101–113, 2004.
- [6] Vladimir Barash and John Kelly. Salience vs. commitment: Dynamics of political hashtags in russian twitter. (9), 2012.
- [7] Frank M. Bass. A new product growth for model consumer durables. *Management Science*, 15(5):215–227, 1969.
- [8] Rodney J Baxter. *Exactly solved models in statistical mechanics*. Elsevier, 2016.
- [9] Diego Beas. How spain’s 15-m movement is redefining politics. *The Guardian*, 15, 2011.
- [10] Richard A Becker, Ramon Caceres, Karrie Hanson, Ji Meng Loh, Simon Urbanek, Alexander Varshavsky, and Chris Volinsky. A tale of one city: Using cellular network data for urban planning. *IEEE Pervasive Computing*, 10(4):18–26, 2011.

- [11] Edward A Bender and E Rodney Canfield. The asymptotic number of labeled graphs with given degree sequences. *Journal of Combinatorial Theory, Series A*, 24(3):296–307, 1978.
- [12] George Berry and Christopher J Cameron. A new method to reduce over-estimation of thresholds with observational network data. *arXiv preprint arXiv:1702.02700*, 2017.
- [13] Norman Biggs, E Keith Lloyd, and Robin J Wilson. *Graph Theory 1736–1936*. Oxford University Press, 1976.
- [14] Sushil Bikhchandani, David Hirshleifer, and Ivo Welch. A theory of fads, fashion, custom, and cultural change as informational cascades. *Journal of Political Economy*, 100:992–1026, 1992.
- [15] Marián Boguñá, Romualdo Pastor-Satorras, Albert Díaz-Guilera, and Alex Arenas. Models of social networks based on social distance attachment. *Physical Review E*, 70(5):056122, 2004.
- [16] Béla Bollobás. Random graphs. *Academic, London*, 1985.
- [17] Stephen P Borgatti, Ajay Mehra, Daniel J Brass, and Giuseppe Labianca. Network analysis in the social sciences. *Science*, 323(5916):892–895, 2009.
- [18] J. Borge-Holthoefer, R. A. Baños, S. González-Bailón, and Y. Moreno. Cascading behavior in complex socio-technical networks. *Journal of Complex Networks*, 1:3–24, 2013.
- [19] Javier Borge-Holthoefer and Yamir Moreno. Absence of influential spreaders in rumor dynamics. *Physical Review E*, 85(2):026116, 2012.
- [20] Javier Borge-Holthoefer, Alejandro Rivero, Iñigo García, Elisa Cauhé, Alfredo Ferrer, Darío Ferrer, David Francos, David Iñiguez, María Pilar Pérez, Gonzalo Ruiz, et al. Structural and dynamical patterns on online social networks: The spanish may 15th movement as a case study. *PLOS ONE*, 6(8):e23883, 2011.
- [21] Guido Caldarelli, Andrea Capocci, Paolo De Los Rios, and Miguel A Muñoz. Scale-free networks from varying vertex intrinsic fitness. *Physical Review Letters*, 89(25):258702, 2002.

- [22] Duncan S Callaway, Mark E J Newman, Steven H Strogatz, and Duncan J Watts. Network robustness and fragility: Percolation on random graphs. *Physical Review Letters*, 85(25):5468–5471, 2000.
- [23] Arnaud Casteigts, Paola Flocchini, Walter Quattrociocchi, and Nicola Santoro. Time-varying graphs and dynamic networks. *International Journal of Parallel, Emergent and Distributed Systems*, 27(5):387–408, 2012.
- [24] Claudio Castellano, Daniele Vilone, and Alessandro Vespignani. Incomplete ordering of the voter model on small-world networks. *Europhysics Letters*, 63(1):153–158, 2003.
- [25] Damon Centola. The spread of behavior in an online social network experiment. *Science*, 329(5996):1194–1197, 2010.
- [26] Damon Centola and Michael Macy. Complex contagions and the weakness of long ties. *American Journal of Sociology*, 113(3):702–734, 2007.
- [27] KG Chakraborty. Statistical mechanics of a bethe lattice. *Zeitschrift für Physik B Condensed Matter*, 55(3):231–234, 1984.
- [28] John Chalupa, Paul L Leath, and Gary R Reich. Bootstrap percolation on a Bethe lattice. *Journal of Physics C: Solid State Physics*, 12(1):L31–L35, 1979.
- [29] Nicholas A Christakis and James H Fowler. Social contagion theory: Examining dynamic social networks and human behavior. *Statistics in Medicine*, 32(4):556–577, 2013.
- [30] Elihu Katz Coleman, James S. and Herbert Menzel. *Medical Innovation: A Diffusion Study*. The Bobbs Merrill Company, 1966.
- [31] Deepak Dhar, Prabodh Shukla, and James P Sethna. Zero-temperature hysteresis in the random-field Ising model on a Bethe lattice. *Journal of Physics A: Mathematical and General*, 30(15):5259–5267, 1997.
- [32] Oxford English Dictionary. *Oxford English Dictionary Online*. Oxford University Press, Oxford, UK <http://www.oed.com>, 2008.
- [33] Peter S Dodds, Kameron Decker Harris, and Christopher M Danforth. Limited imitation contagion on random networks: Chaos, universality, and unpredictability. *Physical Review Letters*, 110(15):158701, 2013.

- [34] Peter S Dodds and Joshua L Payne. Analysis of a threshold model of social contagion on degree-correlated networks. *Physical Review E*, 79(6):066115, 2009.
- [35] Peter S Dodds and Duncan J Watts. Universal behavior in a generalized model of contagion. *Physical Review Letters*, 92(21):218701, 2004.
- [36] Cyril Domb. On the theory of cooperative phenomena in crystals. *Advances in Physics*, 9(35):245–361, 1960.
- [37] Sergei N Dorogovtsev and José FF Mendes. *Evolution of Networks: From Biological Nets to the Internet and WWW*. Oxford University Press, 2013.
- [38] Paul Erdős and Alfréd Rényi. On random graphs i. *Publicationes Mathematicae Debrecen*, 6:290–297, 1959.
- [39] Peter G Fennell, Sergey Melnik, and James P Gleeson. Limitations of discrete-time approaches to continuous-time contagion dynamics. *Physical Review E*, 94:052125, 2016.
- [40] Clay Fink, Aurora Schmidt, Vladimir Barash, Christopher Cameron, and Michael Macy. Complex contagions and the diffusion of popular twitter hashtags in nigeria. *Social Network Analysis and Mining*, 6(1):1, 2016.
- [41] Bailey K Fosdick, Daniel B Larremore, Joel Nishimura, and Johan Ugander. Configuring random graph models with fixed degree sequences. *arXiv preprint arXiv:1608.00607*, 2016.
- [42] Noah E. Friedkin and Eugene C. Johnsen. *Social Influence Network Theory*. Cambridge University Press, Cambridge, UK, 2011.
- [43] Sebastian Funk, Marcel Salathé, and Vincent AA Jansen. Modelling the influence of human behaviour on the spread of infectious diseases: A review. *Journal of the Royal Society Interface*, 7(50):1247–1256, 2010.
- [44] Aram Galstyan and Paul Cohen. Cascading dynamics in modular networks. *Physical Review E*, 75(3):036109, 2007.
- [45] Mahdieh Ghasemi, Hossein Seidkhani, Faezeh Tamimi, Maseud Rahgozar, and Ali Masoudi-Nejad. Centrality measures in biological networks. *Current Bioinformatics*, 9(4):426–441, 2014.

- [46] Edgar N Gilbert. Random graphs. *The Annals of Mathematical Statistics*, 30(4):1141–1144, 1959.
- [47] Malcolm Gladwell. *The Tipping Point: How Little Things Can Make A Big Difference*. Hachette Digital, Inc., 2006.
- [48] James P Gleeson. Cascades on correlated and modular random networks. *Physical Review E*, 77(4):046117, 2008.
- [49] James P Gleeson. Bond percolation on a class of clustered random networks. *Physical Review E*, 80(3):036107, 2009.
- [50] James P Gleeson. Binary-state dynamics on complex networks: pair approximation and beyond. *Physical Review X*, 3(2):021004, 2013.
- [51] James P Gleeson and Diarmuid J Cahalane. An analytical approach to cascades on random networks. *SPIE Fourth International Symposium on Fluctuations and Noise*, page 66010W, 2007.
- [52] James P Gleeson and Diarmuid J Cahalane. Seed size strongly affects cascades on random networks. *Physical Review E*, 75(5):056103, 2007.
- [53] James P Gleeson, Davide Cellai, Jukka-Pekka Onnela, Mason A Porter, and Felix Reed-Tsochas. A simple generative model of collective online behavior. *Proceedings of the National Academy of Sciences*, 111(29):10411–10415, 2014.
- [54] James P Gleeson, Sergey Melnik, Jonathan A Ward, Mason A Porter, and Peter J Mucha. Accuracy of mean-field theory for dynamics on real-world networks. *Physical Review E*, 85(2):026106, 2012.
- [55] James P Gleeson, Kevin P OSullivan, Raquel A Baños, and Yamir Moreno. Effects of network structure, competition and memory time on social spreading phenomena. *Physical Review X*, 6(2):021019, 2016.
- [56] Sharad Goel, Ashton Anderson, Jake Hofman, and Duncan J Watts. The structural virality of online diffusion. *Management Science*, 62(1):180–196, 2015.
- [57] Manuel Gomez Rodriguez, Jure Leskovec, and Andreas Krause. Inferring networks of diffusion and influence. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1019–1028. ACM, 2010.

- [58] S. González-Bailón, J. Borge-Holthoefer, A. Rivero, and Y. Moreno. The dynamics of protest recruitment through an online network. *Scientific Reports*, 1:197, 2011.
- [59] Sandra González-Bailón, Javier Borge-Holthoefer, and Yamir Moreno. Broadcasters and hidden influentials in online protest diffusion. *American Behavioral Scientist*, 57(7):943–965, 2013.
- [60] Mark Granovetter. Threshold models of collective behavior. *American Journal of Sociology*, 83(6):1420–1443, 1978.
- [61] Steve Gregory. Ordered community structure in networks. *Physica A: Statistical Mechanics and its Applications*, 391(8):2752–2763, 2012.
- [62] Catherine Grevet, Loren G Terveen, and Eric Gilbert. Managing political differences in social media. In *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing*, pages 1400–1408. ACM, 2014.
- [63] Thilo Gross and Hiroki Sayama. *Adaptive Networks: Theory, Models and Applications*. Springer, 2009.
- [64] Douglas Guilbeault, Joshua Becker, and Damon Centola. Complex contagions: A decade in review. In Sune Lehmann and Yong-Yeol Ahn, editors, *Spreading Dynamics in Social Systems*. Springer-Verlag, forthcoming (arXiv:1710.07606).
- [65] Roger Guimera, Stefano Mossa, Adrian Turtshi, and LA Nunes Amaral. The worldwide air transportation network: Anomalous centrality, community structure, and cities’ global roles. *Proceedings of the National Academy of Sciences of the USA*, 102(22):7794–7799, 2005.
- [66] Adam Hackett and James P Gleeson. Cascades on clique-based graphs. *Physical Review E*, 87(6):062801, 2013.
- [67] Adam Hackett, Sergey Melnik, and James P Gleeson. Cascades on a class of clustered random networks. *Physical Review E*, 83(5):056107, 2011.
- [68] Adam W Hackett. Cascade dynamics on complex networks. 2011.
- [69] Joan M Herbers. Time resources and laziness in animals. *Oecologia*, 49(2):252–262, 1981.

- [70] Jake M. Hofman, Amit Sharma, and Duncan J. Watts. Prediction and explanation in social systems. *Science*, 355(6324):486–488, 2017.
- [71] Petter Holme and Mark E J Newman. Nonequilibrium phase transition in the coevolution of networks and opinions. *Physical Review E*, 74(5):056108, 2006.
- [72] Matthew O. Jackson and Yves Zenou. Games on networks. In P. Young and S. Zamir, editors, *Handbook of Game Theory Vol. 4*, pages 95–163. Elsevier, 2014.
- [73] Jonas S Juul and Mason A Porter. Synergistic effects in threshold models on networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 28(1):013115, 2018.
- [74] Fariba Karimi and Petter Holme. Threshold model of cascades in empirical temporal networks. *Physica A*, 392(16):3476–3483, 2013.
- [75] Brian Karrer and MEJ Newman. Random graphs containing arbitrary distributions of subgraphs. *Physical Review E*, 82(6):066118, 2010.
- [76] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146. ACM, 2003.
- [77] Jung Yeol Kim and K-I Goh. Coevolution and correlated multiplexity in multiplex networks. *Physical Review Letters*, 111(5):058702, 2013.
- [78] Konstantin Klemm, M Ángeles Serrano, Víctor M Eguíluz, and Maxi San Miguel. A measure of individual role in collective dynamics. *Scientific reports*, 2, 2012.
- [79] Dirk Koschützki and Falk Schreiber. Centrality analysis methods for biological networks and their application to gene regulatory networks. *Gene regulation and systems biology*, 2:193, 2008.
- [80] Michio Kurata, Ryoichi Kikuchi, and Tatsuro Watari. A theory of cooperative phenomena. iii. detailed discussions of the cluster variation method. *The Journal of Chemical Physics*, 21(3):434–448, 1953.

- [81] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is Twitter, a social network or a news media? In *Proceedings of the 19th International Conference on World Wide Web*, pages 591–600. ACM, 2010.
- [82] David Lazer, Alex Sandy Pentland, Lada Adamic, Sinan Aral, Albert-László Barabási, Devon Brewer, Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann, et al. Life in the network: the coming age of computational social science. *Science (New York, NY)*, 323(5915):721, 2009.
- [83] Kyu-Min Lee, Charles D Brummitt, and K-I Goh. Threshold cascades with response heterogeneity in multiplex networks. *Physical Review E*, 90(6):062816, 2014.
- [84] Roger Leenders. Longitudinal behavior of network structure and actor attributes: modeling interdependence of contagion and selection. *Evolution of Social Networks*, 1:165–184, 1997.
- [85] Kristina Lerman. Information is not a virus, and other consequences of human cognitive limits. *arXiv preprint arXiv:1605.02660*, 2016.
- [86] Ted G Lewis. *Network Science: Theory and Applications*. John Wiley & Sons, 2011.
- [87] Thomas M Liggett. *Stochastic Interacting Systems: Contact, Voter and Exclusion Processes*, volume 324. Springer, 1999.
- [88] Yongwhan Lim, Asuman Ozdaglar, and Alexander Teytelboym. A simple model of cascades in networks. Technical report, mimeo, 2015.
- [89] Susanne Lohmann. The dynamics of informational cascades. *World politics*, 47(1):42–101, 1994.
- [90] Dunia Lopez-Pintado and Duncan J Watts. Social influence, binary decisions and collective dynamics. *Rationality and Society*, 20(4):399–443, 2008.
- [91] Helen Margetts, Peter John, Scott Hale, and Taha Yasseri. *Political Turbulence: How Social Media Shape Collective Action*. Princeton University Press, 2015.
- [92] Naoki Masuda, Yoji Kawamura, and Hiroshi Kori. Impact of hierarchical modular structure on ranking of individual nodes in directed networks. *New Journal of Physics*, 11(11):113002, 2009.

- [93] Naoki Masuda and Hiroshi Kori. Dynamics-based centrality for directed networks. *Physical Review E*, 82(5):056107, 2010.
- [94] Nicholas J McCullen, Alastair M Rucklidge, Catherine SE Bale, Tim J Foxon, and William F Gale. Multiparameter models of innovation diffusion on complex networks. *SIAM Journal on Applied Dynamical Systems*, 12(1):515–532, 2013.
- [95] Miller McPherson, Lynn Smith-Lovin, and James M Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27:415–444, 2001.
- [96] John K McSweeney. Single-seed cascades on clustered *arXiv preprint arXiv:1510.00056*, 2015.
- [97] Sergey Melnik, Adam Hackett, Mason A Porter, Peter J Mucha, and James P Gleeson. The unreasonable effectiveness of tree-based theory for networks with clustering. *Physical Review E*, 83:036112, Mar 2011.
- [98] Sergey Melnik, Mason A Porter, Peter J Mucha, and James P Gleeson. Dynamics on modular networks with heterogeneous correlations. *Chaos*, 24(2):023106, 2014.
- [99] Sergey Melnik, Jonathan A Ward, James P Gleeson, and Mason A Porter. Multi-stage complex contagions. *Chaos*, 23(1):013124, 2013.
- [100] Bjarke Mønsted, Piotr Sapieżyński, Emilio Ferrara, and Sune Lehmann. Evidence of complex contagion of information in social media: An experiment using twitter bots. *arXiv preprint arXiv: 1703.06027*, 2017.
- [101] Geoffrey A Moore. *Crossing the chasm*. New York: HarperBusiness, 2006.
- [102] Adilson E Motter and Yang Yang. The unfolding and control of network cascades. *Physics Today*, 70(1):32–39, 2017.
- [103] Seth Myers and Jure Leskovec. On the convexity of latent social network inference. In *Advances in Neural Information Processing Systems*, pages 1741–1749, 2010.
- [104] Mark E J Newman. Random graphs with clustering. *Physical Review Letters*, 103(5):058701, 2009.
- [105] Mark E J Newman. *Networks: An Introduction*. Oxford University Press, 2010.

- [106] Mark E J Newman, Steven H Strogatz, and Duncan J Watts. Random graphs with arbitrary degree distributions and their applications. *Physical Review E*, 64(2):026118, 2001.
- [107] Ashkan Aleali Paulo Shakarian Nikhil Kumar, Ruocheng Guo. An empirical evaluation of social influence metrics. *arXiv:1607.00720*, 2016.
- [108] Hyung-Jong Park, Lawrence Kincaid, Kyung-Kyoon Chung, Dal-Sun Han, and Sea-Baick Lee. Mothers clubs and family planning in korea. *Seoul Korea Seoul National University School of Public Health May 1974. 312 p*, 1974.
- [109] Francisco J Pérez-Reche, Jonathan J Ludlam, Sergei N Taraskin, and Christopher A Gilligan. Synergy in spreading processes: from exploitative to explorative foraging strategies. *Physical Review Letters*, 106(21):218701, 2011.
- [110] Pablo Piedrahíta, Javier Borge-Holthoefer, Yamir Moreno, and Alex Arenas. Modeling self-sustained activity cascades in socio-technical networks. *EPL (Europhysics Letters)*, 104(4):48004, 2013.
- [111] Martin Pielot, Rodrigo de Oliveira, Haewoon Kwak, and Nuria Oliver. Didnt you see my message? predicting attentiveness to mobile instant messages. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, 2014.
- [112] Norman Polansky, Ronald Lippitt, and Fritz Redl. An investigation of behavioral contagion in groups. *Human Relations*, 3:319–348, 1950.
- [113] Mason A Porter and James P Gleeson. Dynamical systems on networks: A tutorial. In *Frontiers in Applied Dynamical Systems: Reviews and Tutorials*, volume 4. Springer, 2016.
- [114] Mason A Porter and Sam D Howison. The role of network analysis in industrial and applied mathematics. arxiv.org/abs/1703.06843.
- [115] Giuliano Punzo, George F Young, Malcolm Macdonald, and Naomi E Leonard. Using network dynamical influence to drive consensus. *Scientific reports*, 6, 2016.
- [116] Anatol Rapoport. Cycle distributions in random nets. *The Bulletin of Mathematical Biophysics*, 10(3):145–157, 1948.

- [117] Oliver Riordan. *Graph theory course lecture notes*. University of Oxford, 2014. Course Material for B8.5: Graph Theory.
- [118] Garry Robins, Philippa Pattison, and Peter Elliott. Network models for social influence processes. *Psychometrika*, 66(2):161–189, 2001.
- [119] Everett M Rogers. Categorizing the adopters of agricultural practices. *Rural Sociology*, 23(4):345–354, 1958.
- [120] Everett M Rogers. *Diffusion of Innovations*. Simon and Schuster, 2010.
- [121] Everett M Rogers, JR Ascroft, and N Rolling. Diffusion of innovations in brazil, nigeria, and india. *Unpublished report, Michigan State University*, 1970.
- [122] Henry A Rowley, Shumeet Baluja, and Takeo Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):23–38, 1998.
- [123] L. K. Runnels. Phase transition of a bethe lattice gas of hard molecules. *Journal of Mathematical Physics*, 8(10):2081–2087, 1967.
- [124] Salvatore Scellato, Anastasios Noulas, Renaud Lambiotte, and Cecilia Mascolo. Socio-spatial properties of online location-based social networks. *The International AAAI Conference on Weblogs and Social Media*, 11:329–336, 2011.
- [125] Thomas C Schelling. Dynamic models of segregation. *Journal of Mathematical Sociology*, 1(2):143–186, 1971.
- [126] Thomas C Schelling. Hockey helmets, concealed weapons, and daylight saving: A study of binary choices with externalities. *Journal of Conflict Resolution*, pages 381–428, 1973.
- [127] Munik Shrestha and Cristopher Moore. Message-passing approach for threshold models of behavior in networks. *Physical Review E*, 89(2):022805, 2014.
- [128] Tristan Mark Snowsill, Nick Fyson, Tijn De Bie, and Nello Cristianini. Refining causality: who copied from whom? In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 466–474. ACM, 2011.
- [129] Ray Solomonoff and Anatol Rapoport. Connectivity of random nets. *The Bulletin of Mathematical Biophysics*, 13(2):107–117, 1951.

- [130] Amanda L Traud, Eric D Kelsic, Peter J Mucha, and Mason A Porter. Comparing community structure to characteristics in online collegiate social networks. *SIAM Review*, 53:526–543, 2011.
- [131] Amanda L Traud, Peter J Mucha, and Mason A Porter. Social structure of Facebook networks. *Physica A*, 391(16):4165–4180, 2012.
- [132] Johan Ugander, Lars Backstrom, Cameron Marlow, and Jon Kleinberg. Structural diversity in social contagion. *Proceedings of the National Academy of Sciences of the United States of America*, 109(16):5962–5966, 2012.
- [133] Thomas W Valente. *Network Models of the Diffusion of Innovations*. Hampton Press, 1995.
- [134] Thomas W Valente. Social network thresholds in the diffusion of innovations. *Social Networks*, 18(1):69–89, 1996.
- [135] Thomas W Valente, Stephanie R Dyal, Kar-Hai Chu, Heather Wipfli, and Kayo Fujimoto. Diffusion of innovations theory applied to global tobacco control treaty ratification. *Social Science & Medicine*, 145:89–97, 2015.
- [136] George G. Vega Yon, Stephanie Dyal, Timothy Hayes, and Thomas Valente. *netdiffuseR: Analysis of Diffusion and Contagion Processes on Networks*, 2017. R package version 1.18.1.
- [137] Duncan J Watts. A simple model of global cascades on random networks. *Proceedings of the National Academy of Sciences of the USA*, 99(9):5766–5771, 2002.
- [138] Duncan J. Watts. Should social science be more solution-oriented? *Nature Human Behaviour*, 1:0015, 2017.
- [139] Duncan J Watts and Peter S Dodds. Influentials, networks, and public opinion formation. *Journal of Consumer Research*, 34(4):441–458, 2007.
- [140] Duncan J Watts, Peter S Dodds, and Mark E J Newman. Identity and search in social networks. *Science*, 296(5571):1302–1305, 2002.
- [141] Duncan J Watts and Steven H Strogatz. Collective dynamics of ‘small-world’ networks. *Nature*, 393(6684):440–442, 1998.

- [142] Eric W Weisstein. Gamma distribution. From Mathworld — A Wolfram Web Resource, Available at <http://mathworld.wolfram.com/DeltaFunction.html>.
- [143] Jianshu Weng, Ee-Peng Lim, Jing Jiang, and Qi He. Twiterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 261–270. ACM, 2010.
- [144] Lilian Weng, Filippo Menczer, and Yong-Yeol Ahn. Virality prediction and community structure in social networks. *Scientific Reports*, 3, 2013.
- [145] Herbert S Wilf. *Generatingfunctionology*. Academic Press, Inc., 1990.
- [146] Ling Heng Wong, Philippa Pattison, and Garry Robins. A spatial model for social networks. *Physica A*, 360(1):99–120, 2006.
- [147] Nicholas C Wormald. Some problems in the enumeration of labelled graphs. *Bulletin of the Australian Mathematical Society*, 21(1):159–160, 1980.
- [148] H Peyton Young. Innovation diffusion in heterogeneous populations: Contagion, social influence, and social learning. *American Economic Review*, 99(5):1899–1924, December 2009.