

Analysis and Control of the Linear Threshold Model of Cascades in Large-Scale Networks

A Local Mean-Field Approach
- PhD Thesis -



Wilbert Samuel Rossi

Politecnico di Torino
Doctoral School

PhD program in Mathematics in Engineering - XXVII cycle

Supervisors: F. Fagnani¹, G. Como²
PhD course coordinator: L. Rondoni³

March 2015

¹Dept. of Mathematical Sciences "G. L. Lagrange", Politecnico di Torino (Italy)

²Dept. of Automatic Control, Lund University (Sweden)

³Dept. of Mathematical Sciences "G. L. Lagrange", Politecnico di Torino (Italy)

to Laura

Abstract

The spread of new ideas, behaviors and technology may exhibit cascading effects in social, economic and technological networks. These phenomena generally depend on the topology of the network as well as the nature of the local agents' dynamics.

In this thesis we consider the Linear Threshold Model, deployed on random graphs. The model describes a binary activation process in a network of agents. At every iteration, each agent compares the number of active neighbors with a personal activation threshold, which determines the subsequent active or inactive state of the agent. The threshold condition can also be interpreted as a graphical game with coordination structure. In representing processes of technology adoption, it is more suitable a Permanent Activation variant of the Linear Threshold Model where active agents can never deactivate. We proved a sufficient condition under which the two version of the model coincide.

We analyzed the linear threshold model on a large random network, specifically the directed configuration model with heterogeneous agents. The tree-like local structure of the random networks allows to approximate the evolution of the expected fractional activation with a recursive equation. This equation, called Local Mean-Field dynamic, describes the evolution of the expected activation on an infinite tree with the same statistical properties of the original network. We proved a concentration theorem: for a generic instance of the network, the probability that the activation process and the Local Mean Field dynamic are close converges to one exponentially fast in the network size.

If the activation thresholds are constant, the analysis reduces to the study of the fixed point of a scalar autonomous system and the corresponding trajectories. This analysis gives the asymptotic extension of the activation: we observed that in networks with sufficiently heterogeneous thresholds selective activation may occur. With constant thresholds the approach can be extended to study the Permanent Activation dynamic.

Remarkably, the Local Mean Field dynamic equation and the concentration theorem continue to hold when the thresholds are dynamically adjusted, making the approach amendable to the design of control strategies. We formulated an optimal control problem and we considered a simplified version on a regular network. We compared the optimal solution with two sub-optimal strategies, developed with the aim to identify an heuristics for the problem's solution.

Several aspects of the research discussed in the this Dissertation can be further investigated and generalized. To mention one, the comparison of the analysis presented here with other network topologies and possibly real network data.

Riassunto

La diffusione di nuove idee, comportamenti e tecnologie può presentare effetti a cascata in reti sociali, economiche e tecnologiche. In genere questi fenomeni dipendono sia dalla struttura topologica della rete che dal comportamento dinamico dei singoli agenti.

Nella tesi consideriamo il “modello di soglia lineare” (*Linear Threshold Model*), analizzandolo su un grafo casuale. Il modello descrive un processo di attivazione in una rete di agenti. Ad ogni iterazione, ciascun agente considera il numero di vicini già attivi e lo confronta con la sua soglia di attivazione, aggiornando il proprio stato di attivazione. Tramite la teoria dei giochi, la condizione di soglia può essere interpretata come un gioco di coordinazione su una rete. Per rappresentare i processi di adozione di nuove tecnologie, si può considerare una variante del modello di soglia lineare, in cui lo stato attivo sia permanente. Sotto alcune condizioni sufficienti, le due versioni del modello coincidono.

Nella tesi analizziamo il modello di soglia lineare in un grafo casuale contenente un grande numero di agenti, nello specifico il cosiddetto *Configuration Model* diretto. Tale grafo casuale ha una struttura che localmente risulta analoga a quella di un albero. Questo aspetto permette di approssimare il progredire della frazione di agenti attivi attesi con una equazione ricorsiva. L'equazione, che chiamiamo “dinamica locale di campo medio” (*Local Mean-Field dynamic*), corrisponde appunto all'evoluzione dell'attivazione attesa su un grafo ad albero, infinito, avente le stesse proprietà statistiche della rete originale. Abbiamo quindi dimostrato un teorema di concentrazione: per una generica rete casuale, la probabilità che il processo di attivazione e la dinamica locale di campo medio siano vicine converge ad uno, in modo esponenziale rispetto alla taglia della rete.

Se le singole soglie di attivazione sono costanti, l'analisi dell'equazione ricorsiva si riduce allo studio delle traiettorie e dei punti fissi di un sistema dinamico autonomo e scalare. Dall'analisi si può ricostruire l'estensione asintotica dell'attivazione: abbiamo osservato che, se le soglie di attivazione sono sufficientemente eterogenee, si può verificare una attivazione selettiva. Inoltre, con le soglie costanti, l'approccio presentato è adatto allo studio del modello con attivazione permanente.

La dinamica locale di campo medio ed il teorema di concentrazione continuano a valere anche quando le soglie di attivazione vengono variate durante la dinamica. Questo permette di formulare un problema di controllo ottimo, con lo scopo di individuare delle strategie di controllo del processo di attivazione. Di tale problema consideriamo una versione semplificata e confrontiamo la soluzione con due strategie sub-ottime, nel tentativo di individuare un

approccio euristico all'ottimizzazione.

Molti aspetti del lavoro discusso in questa tesi possono essere estesi e generalizzati. Una possibile estensione di questo lavoro potrebbe essere il confronto dei risultati ottenuti sul *Configuration Model* diretto con i fenomeni che avvengono su reti aventi diversa struttura topologica.

Acknowledgements

First and foremost, I would like to thank my advisors, Fabio Fagnani and Giacomo Como. I am grateful to them for their patient guidance and encouragement throughout these three years.

My acknowledgements go to Julien Hendrickx for his careful reading of this Dissertation; his precise comments have been extremely useful.

I would also like to thank all the researchers who have provided me with feedback on my work, and the administrative staff for their help. I just mention Paolo Frasca, Martina Maggio and Eva Westin, but my gratitude extends to many people in Torino and Lund.

A huge thank goes to Massimiliano, for proofreading this manuscript with great care and helping me to decrease the number of typos significantly. For all the remaining ones I take sole responsibility.

I would like to thank the PhD students and “sympathizers” with whom I shared offices, lunches, gossips and Champions League evenings, like Domenica, Lucia, Lorenzo, Yang, Christian, Leonid and Gustav.

During this PhD I also had a “spiritual” advisor, Sergio, a wise guide and a sincere friend. I would also like to thank Ilir, Gianmarco, Biagio, Ajosh, Davide, Simone, Danilo and all my friends in Cavour, Torino and Lund. Funny and relaxing moments (as well as more “mature” conversations) would not have been possible without them!

A special thank goes to Laura: she motivated me when nothing was working and was enthusiastic when the results finally came.

Last, but certainly not least, I have to express my gratitude to my family. They have always been supportive, patient and caring even when I was too stressed to listen to them.

Surely I am missing someone, but it is really time to print the thesis! 😊

Ringraziamenti

Per prima cosa desidero ringraziare i miei tutori, Fabio Fagnani e Giacomo Como. Sono grato per la loro paziente guida ed incoraggiamento durante questi tre anni.

Ringrazio Julien Hendrickx per la sua attenta lettura della Dissertazione; i suoi commenti precisi sono stati molto utili.

Vorrei anche ringraziare tutti quei ricercatori con cui ho discusso del mio lavoro, ed il personale di segreteria per il loro aiuto con le pratiche amministrative. Mi limito a menzionare Paolo Frasca, Martina Maggio ed Eva Westin, ma la mia gratitudine si estende a molte persone a Torino e Lund.

Un caloroso ringraziamento va a Massimiliano, per aver controllato l'inglese della Dissertazione ed avermi aiutato a ridurre gli errori. Riguardo a quelli rimanenti mi assumo tutte le responsabilità.

Voglio anche ringraziare i dottorandi e “simpatizzanti” con cui ho condiviso uffici, pranzi, chiacchierate e serate a guardare la Champions League: Domenica, Lucia, Lorenzo, Yang, Christian, Leonid e Gustav in particolare.

Durante il dottorato ho anche avuto un tutor “spirituale”, Sergio, un amico sincero ed una guida saggia. Ringrazio anche Ilir, Gianmarco, Biagio, Ajosh, Davide, Simone e Danilo, e tutti i miei amici di Cavour, Torino e Lund. Senza di voi, i momenti divertenti e rilassanti (come pure alcune discussioni “mature”) non sarebbero stati possibili.

Un ringraziamento speciale va a Laura, per avermi motivato quando le cose non funzionavano e per il suo entusiasmo quando i risultati sono finalmente arrivati.

Infine, esprimo tutta la mia gratitudine alla mia famiglia. Mi ha sempre incoraggiato, con pazienza ed attenzione, anche quando ero troppo stressato per ascoltarla.

Sicuramente sto dimenticando qualcuno, ma è proprio giunto il momento di stampare la tesi! 😊

Contents

Abstract	2
Riassunto	3
Acknowledgements	5
Ringraziamenti	6
Introduction	9
1 Linear Threshold Models	13
1.1 Linear threshold models of cascades	13
1.2 The equivalence condition	15
1.3 Game theoretic models	18
1.4 Financial networks and contagion	22
2 The Local Mean-Field approximation	28
2.1 Introduction	28
2.1.1 The network model	29
2.2 A heuristic argument	31
2.3 The formal derivation	33
2.4 The concentration theorem	35
2.5 Examples and comments	40
3 Analysis of Local Mean-Field dynamics	46
3.1 Basic properties of the LMF dynamical system	47
3.2 The extension of the LMF approach to the PALTM	49
3.3 Regular networks	53
3.3.1 Networks with one threshold value	54
3.3.2 Networks with two threshold values	57
3.3.3 Networks with several threshold values	67
3.4 Extensions to non-regular networks	68
3.4.1 Asymptotic activation in the PALTM	69
3.4.2 Vanishing initial activation in the PALTM	71

4	Control of Local Mean-Field dynamics	74
4.1	Problem formulation	75
4.2	A regular network case	76
4.2.1	Model Predictive Control	81
4.2.2	Linearization method	83
4.3	Numerical simulations	93
4.3.1	Example 1	95
4.3.2	Example 2	100
4.3.3	Example 3	103
4.3.4	Example 4	106
	Conclusion and future work	113
A	Graph Theory	115
A.1	Fundamentals	115
A.2	Multigraphs, neighborhoods and trees	119
A.3	Random Graphs	121
B	Functions in the LMF equations	124
B.1	Properties of $f_{d,r}(x)$	124
B.2	Properties of $g_{d,r}(x)$	127

Introduction

The spread of new ideas and behaviors, the adoption of innovative technologies and the default contagion of financial institutions are dynamic processes which may exhibit cascading effects in social, economic and technological networks. These phenomena generally depend on the topology of the network as well as on the nature of the local agents' dynamics.

A popular model proposed to describe such diffusion phenomena, for example the diffusion of innovation in social networks, is the *Linear Threshold Model*. Let us suppose that every agent in the network uses an older technology when some early adopter suddenly switch to a newer one. The remaining agents decide the adoption of the new technology with a simple threshold rule. Each agent counts the number of acquaintances with the new technology and decide to adopt if at least a certain (personally defined) number of them, has it. The adoption process can continue in rounds, until either all the network has adopted the new technology or no further adoption is possible. We call *cascade* a large, unexpected increment in the number of final adopters produced by a tiny increment in the number of early adopters. In the example, the manufacturer of the new technology would be clearly interested in understanding where the tipping point is, and which factors can trigger an adoption cascade.

The Linear Threshold Model exhibits such cascading phenomena and can be used to represent such problems. Cascades are one core problem addressed in literature, for example the dependence of the final number of adopters on the number of early adopter and on the network topology. Other important questions regard the identification of influential agents and the dynamic progression of the adoption. From a control perspective, the possibility to change the decision of some agents or to modify its adoption threshold, would be very interesting.

Diffusion processes are common in large networks, say with order of millions agents. Often only statistical informations about the network's topology are available (such as the *degree distribution*) and in any case more detailed knowledge would be unmanageable [34]. For our analysis, we consider a large, *random network* model, able to incorporate the degree distribution of the agents, i.e.

a directed version of the so called *configuration model*. The aim remains to predict the behavior of networked systems, but in a statistical sense. In the adoption example that would mean to relate the fraction of adopters, its dynamic and asymptotic value, to the fraction of early adopters, to the statistic of the adoption thresholds and of the number of acquaintances.

Finally, suppose to have the possibility to influence a limited fraction of agents, from outside the network, to bias the diffusion process (e.g. toward adoption or refusal of the innovation). An interesting question regards the optimal allocation of this “influence” capability, i.e. characterize the category of nodes that, if influenced, produces the larger bias on the full network.

The simplest and most popular opinion dynamics model is the so called *voter model*. At each round one agent of the network is randomly selected, and is given the possibility to revise its opinion. The rule is simple: the agent looks at the opinion of a randomly chosen acquaintance and, with a given probability, copies that opinion, otherwise it maintains its own. The voter model has several difference with respect to the linear threshold model. In the voter model one agent at a time is chosen, and there is a non trivial probability of copying, aspects that makes the timing of the model much “slower”. But the fundamental difference regards the dependance of the node choice to the acquaintance. In the voter model the agent looks at one random neighbor, while in the linear threshold model all the average opinion of the acquaintances counts.

If there are factors of coordination or persuasion involved, local relative considerations become important in understanding whether new beliefs or behaviors are adopted [41]. These neighborhood effects mean that individual decisions among two alternatives may depend on the relative proportion of neighbors who have taken the same decision.

Several variants of the basic linear threshold model are possible: in the technology adoption example, the early adopters may be stubborn or may revise their choice, the adoption may or may not be permanent and different acquaintance can have different influence in the choice. Literature has investigated the question of whether a behavior initially adopted by a small number of agents will spread to a large fraction of the network, considering mainly dynamics where agents can make a single switch to the new behavior. Moreover, not always the adoption is complete: multiple intermediate equilibria can arise and it becomes interesting to predict the outreach of the adoption process. The diffusion of innovation is just one of the binary decision frameworks that are relevant for application in complex problems [42]. Furthermore, the threshold model admits a “game theoretic” motivation, if we think at the agent choice as an attempt to maximize some target function.

In economic networks a notable example of cascade effects regards the insolvencies of financial institution during the last economic crisis: a type of financial contagion that can be understood as a domino effects. The financial contagion model presented by Amini and colleagues in [4] can be reduced to the linear threshold model, with a directed weighted random graph to represent the exposures between nodes, i.e. the interbank borrowed money. The threshold condition represents the amount of losses that a node can handle before the default, with the bank's capital being analogous to the adoption threshold. Some banks may fail for external reasons, like the early adopters of the innovation example. If we consider the money lent to failed institutions as lost, further contagious failures may happen and spread to a large portion of the financial network.

Relative considerations can be introduced by modeling the social interactions as a game. Morris [33] described the class of local interaction games where players interact strategically with their neighbors in the network. The players' payoff of the available behavior depends on the number of neighbors who have chosen the same alternative. Players trying to maximize their payoffs apply a dynamic best response rule and such rule can be reduced to the threshold condition. Morris [33] studied a deterministic local coordination game on infinite homogeneous networks with simple regular topologies. He introduced the concept of cohesion to describe the clusters of adopters and non-adopters, to answer the question of the existence of a finite set of early adopters able to trigger the cascading adoption in the entire network. He observed that the two different behaviors can coexist even in absence of influential stubborn players that do not participate in the game. If cohesion is sufficient, clusters "self-sustain" their choices.

The *bootstrap percolation*, a process first introduced by Chalupa et al. in [14] to study the magnetization on a infinite regular tree graph, is one of the variant of the threshold model. In the bootstrap percolation, all the nodes share the same degree and adoption threshold, and with one of the options being permanent. On a random regular graph, the cascade conditions for the bootstrap percolation, has been studied [6].

A central problem in literature is the relation between topology, number of early adopters and final spread of the diffusion, with the identification of those conditions that allow the diffusion to the full network or predict the cascade resilience. This knowledge allows the identification of both bottlenecks to the spread of an innovation, with possible marketing application [30] trying to maximize the influence of early adopters [27].

On the other hand understanding the conditions for a cascade, shall help quantify contagion risks in financial networks [4]. If the initial shock occurs

unexpectedly, and the network is not resilient to cascades one could intervene dynamically during the early stages of the contagion process in order to mitigate it.

Therefore we are interested in a description of the transient dynamic of the diffusion process. We perform this analysis on a large directed random network, with prescribed degree and threshold statistics. If thresholds can be dynamically adjusted, we would like to identify control policies and heuristics in order to manage the network and bias the agent population to the desired outcome.

In Chapter 1 we describe the linear threshold model, as a model for a binary activation process on a general graph. We present the variant where one of the two states is permanent and prove a sufficient condition under which the two models are equivalent. The rest of the Chapter is devoted to the “game theoretical” motivation and the financial contagion application of the model.

In Chapter 2 we analyze the linear threshold model on a large directed random network with prescribed degree and threshold distributions. With a local mean-field approach we write a recursive equation that approximates the dynamic of the fraction of active agents. The main contribution is a concentration theorem which guarantees that the linear threshold model, when deployed on a configuration model random graph, behaves arbitrarily similarly to its mean-field approximation, with a probability converging exponentially fast in the network size. Remarkably, the equations and concentration theorem continue to hold when thresholds are time-varying.

In Chapter 3 we analyze the local mean-field dynamical system, with non-varying thresholds. After a general part, we focus on regular networks with a limited number of different thresholds. As the system is autonomous, we study the fixed points and the equilibrium conditions. We discuss the presence of phase transitions and observe a phenomena that we called *selective activation*: under some circumstances, the activation process segregates between the nodes with smaller thresholds.

In Chapter 4 we consider the local mean-field dynamical system with time-varying threshold and we formulate an optimal control problem. To limit the complexity we simplify the problem to the case of a regular network, where each agent threshold can be increased at most of one unit. Our aim is to identify some heuristics of the optimal control policy. To do so, we discuss two sub-optimal control strategy that approximate the solution of the optimization problem. We illustrate this methods with simulations.

Finally we conclude with a summary of the results and future research directions.

Chapter 1

Linear Threshold Models

In this Chapter we introduce two linear threshold activation models, prove an equivalence condition and provide two example to motivate these models. Agents on a network can be active or inactive. Whenever the number of active neighbors of an agent exceeds its personal threshold, it gets active. In the *Linear Threshold Model* (LTM), each active agents can become inactive again if the number of active neighbors decreases below the threshold. In the *Permanent Activation Linear Threshold Model* (PALTM) the activation is permanent, i.e. active agents remain active forever.

For some reasonable initial conditions and restrictions, the trajectories produced by these two models are equivalent: a property used in the following Chapters.

The purpose of these models is to analyze cascade propagations in networks, where the initial activation (or infection) of a few agents eventually leads to the activation of a large proportion of the population. Such phenomena are relevant in marketing, epidemics diffusions with neighborhood effects, the spread of innovation or of financial troubles. Indeed, we provide a game theoretical interpretation of these models and we recover the threshold condition from a simplified version of a financial contagion model.

1.1 Linear threshold models of cascades

Consider a graph $G = (V, E)$ with node set V and edge set $E \subseteq V \times V$. Suppose G is static, directed and connected, and has a finite size $n = |V|$. The letters v, w, u will be used to denote the nodes in G , while e will be used for the edges. The terms node, vertex, agent and player will be used interchangeably throughout the Dissertation. We consider directed graphs, where each edge $(v, w) \in E$ represents the dependance of v on w . The models and properties

of this Chapter extend naturally to undirected graphs, which, for the sake of the Chapter, are directed graphs with the special properties that $(v, w) \in E$ if and only if $(w, v) \in E$ and $(v, v) \notin E$. Given a node $v \in V$, the set of its *out*-neighbors is $N_v = \{w \in V : (v, w) \in E\}$, its *out*-degree is $d_v = |N_v|$ and its *in*-degree is $k_v = |\{w \in V : (w, v) \in E\}|$. An introduction to the basic graph theory can be found in Appendix A, together with some concepts used in the following Chapter.

Let each vertexes of the graph be endowed with a binary state $x_v[t] \in \{0, 1\}$, dependent on a discrete time variable t . The two states 0 and 1, *inactive* and *active* respectively, may assume specific meanings depending on the application. The states of all nodes, stacked together in a vector $\mathbf{x}[t] \in \{0, 1\}^V$, provide the configuration of the network.

The evolution of each node state is specified by a local deterministic rule, which depends on a personal, variable endogenous parameter $r_v[t]$, called *activation thresholds*. Consider the set

$$R = \{\mathbf{r} \in \mathbb{N}^V : 0 \leq r_v \leq d_v + 1\},$$

where \mathbb{N} is the integer set. The vector $\mathbf{r}[t] \in R$ contains the individual activation thresholds of all the agents of the graph, at time t . We may refer to $\mathbf{r}[t]$ as a thresholds sequence, when we mean the collection of the time sequence of thresholds $r_v[t]$, one for each agent $v \in V$. If $(v, w) \in E$, node v can access the state of node w , which means node v is influenced by the behaviors of its d_v *out*-neighbors in N_v . At each time step the whole configuration of the network is updated applying the local rules synchronously. In the following we describe two local threshold rules.

In the *Linear Threshold Model* (LTM in short), each agent activates if the number of active *out* neighbors is bigger or equal than its current activation threshold. Otherwise the agent deactivates.

$$x_v[t + 1] = \begin{cases} 1 & \text{if } \sum_{w \in N_v} x_w[t] \geq r_v[t] \\ 0 & \text{otherwise} \end{cases} \quad (1.1)$$

The rule is applied synchronously by each agent of the network. Given an initial configuration $\mathbf{x}[0] \in \{0, 1\}^V$ and the sequence $\mathbf{r}[t] \in R$ of endogenous activation thresholds, the configuration of the network evolves deterministically. In the linear threshold model agents do not have memory about their previous states and need to compute their new states at each iteration: depending on the local condition, they can switch state from 0 to 1 and viceversa.

A node with $r_v[t] = 0$ will activate regardless of the local condition ($x_v[t + 1] = 1$), whereas a node with $r_v[t] = d_v + 1$ will surely deactivate ($x_v[t + 1] = 0$). These two special activation threshold values can be chosen to enforce

stubbornness. Thresholds smaller than 0 or larger than $d_v + 1$ are not necessary and have been excluded on purpose in the definition of the set R .

We call *Permanent Activation Linear Threshold Model* (PALTM in short) a variant of the linear threshold model where the active state is permanent: once a node is activate it remains active forever. Given an initial configuration $\mathbf{x}[0] \in \{0, 1\}^V$ and a thresholds' sequence $\mathbf{r}[t] \in R$, each agent applies the local rule:

$$x_v[t + 1] = \begin{cases} 1 & \text{if } x_v[t] = 1 \quad \text{or} \quad \sum_{w \in N_v} x_w[t] \geq r_v[t] \\ 0 & \text{otherwise.} \end{cases} \quad (1.2)$$

The PALTM, considered on a undirected graph and with activation threshold $r_v[t] = r \geq 2$ for every agent, is also called *Bootstrap Percolation*. The bootstrap percolation is a dynamic version of the percolation models [6], defined as the spread of activation with an excitation threshold $r \geq 2$ [25]. The case with $r = 1$ is usually excluded since, in a undirected connected graph, an active vertex is trivially sufficient to trigger the full activation of the network.

In the PALTM each agent has a memory of its previous state: this aspect marks a substantial difference with the LTM and, since $x_v[t + 1] \geq x_v[t]$ for every agent, the permanent activation model (1.2) is *progressive* [28]. There are however conditions that make the two model produce the same trajectories. Such conditions are the object of the next Section.

1.2 The equivalence condition

In the next Chapter we will develop a technique to analyze the linear threshold model on a family of directed random graphs. We can apply that technique to the permanent activation linear threshold model, if the two processes coincide. In this Section we establish the conditions under which the PALTM and the more flexible LTM are equivalent.

Before going into the details, we provide the sufficient condition that guarantees the equivalence of the two models.

Proposition 1.1 (Equivalence Condition). *Consider a collection of threshold sequences $\mathbf{r}[t] \in R$ such that for every agent v its threshold sequence $r_v[t]$ is non-increasing, i.e. $\forall v \in V, \forall t \geq 0, r_v[t+1] \leq r_v[t]$. Consider an initial configuration $\mathbf{x}[0] \in \{0, 1\}^V$ such that if $x_v[0] = 1$ then $r_v[0] = 0$. Then, the dynamic $\mathbf{x}[t]$ under the PALTM (1.2) and under the LTM (1.1) coincide.*

The Proposition is the direct consequence of Lemma 1.2 and Lemma 1.3, later in this Section.

The equivalence condition 1.1 provides the recipe by which a PALTM process with non-increasing thresholds can be considered as a realization of the LTM. Once the initial condition $\mathbf{x}[0]$ is provided, it suffices to set to zero the sequence of activation thresholds of every initially active node, i.e. for those v with $x_v[0] = 1$ set $r_v[t] = 0, \forall t \geq 0$.

An example of this idea is shown in Figure 1.1. On the left side, there is a directed cycle network with four nodes. For each node, the number inside the circle is the constant activation threshold sequence, while the color represents the initial condition (white = 0, black = 1). On the network in the left side, with these thresholds and initial condition, the PALTM gives full activation at $t = 3$. Under the LTM there will be a single active node at any time, i.e. “the activation cycles through the four agents”, and just having constant threshold is not enough to get a permanent activation. The same network is drawn on the right side of the figure, but the initially active node has now a threshold sequence which is set at constant zero. For the network on the right, both LTM and PALTM give full activation at $t = 3$.

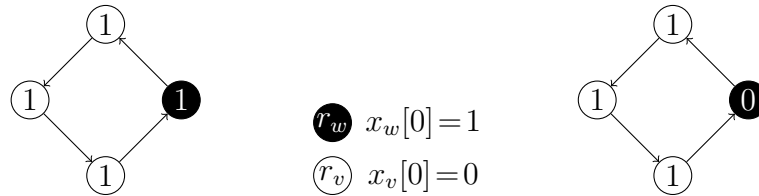


Figure 1.1: An example of the “use” of the equivalence condition 1.1: LTM and PALTM coincide on the right network, but not on the left one.

The equivalence condition is a sufficient condition. In general we can expect equivalence any time the process $\mathbf{x}[t]$ is progressive (i.e. without any deactivation) even under the non permanent LTM. It means that the memory, present in the PALTM, should never be necessary. This is indeed the case, as explained by the following lemma.

Lemma 1.2. *Consider an initial configuration $\mathbf{x}[0] \in \{0, 1\}^V$ and a collection of threshold sequences $\mathbf{r}[t] \in R$. The dynamic $\mathbf{x}[t]$ under the LTM and under the PALTM coincide if and only if $\forall t \geq 0, \nexists v \in V : x_v[t] = 1$ and $\sum_{N_v} x_w[t] < r_v[t]$.*

Proof. Recall the update rule (1.1) of the LTM:

$$x_v[t + 1] = \begin{cases} 1 & \text{if } \sum_{w \in N_v} x_w[t] \geq r_v[t] \\ 0 & \text{otherwise} \end{cases}$$

and that of the PALTM (1.2):

$$x_v[t+1] = \begin{cases} 1 & \text{if } x_v[t] = 1 \text{ or } \sum_{w \in N_v} x_w[t] \geq r_v[t] \\ 0 & \text{otherwise} \end{cases}$$

The updates they describe coincide if and only if $\forall t \geq 0, \forall v \in V$:

$$x_v[t] = 1 \text{ implies } \sum_{w \in N_v} x_w[t] \geq r_v[t]$$

which means that $\forall t \geq 0, \nexists v \in V : x_v[t] = 1$ and $\sum_{N_v} x_w[t] < r_v[t]$. \square

If the initial condition $\mathbf{x}[0]$ and the collection of thresholds sequences $\mathbf{r}[t]$ produce a progressive process $\mathbf{x}[t]$ in a specific graph G under the non-permanent LTM, they would produce exactly the same process under the PALTM.

Now, let suppose that every agent v has a non-increasing threshold sequence. In this case it is enough that the activation of the initial configuration $\mathbf{x}[0]$ does not recede (no deactivation of any initially active node), as explained by the following lemma.

Lemma 1.3. *Consider an initial configuration $\mathbf{x}[0] \in \{0, 1\}^V$ and a collection of threshold sequences $\mathbf{r}[t] \in R$. If $\nexists v \in V : x_v[0] = 1$ and $\sum_{N_v} x_w[0] < r_v[0]$, and $\forall v \in V$ the sequences $r_v[t]$ are non increasing, then the dynamic $\mathbf{x}[t]$ under the PALTM and under the LTM coincide.*

Proof. By induction we show that the hypotheses of lemma 1.2 hold. Suppose that at t^* , $\forall v, x_v[t^*] = 1$ implies $\sum_{w \in N_v} x_w[t^*] \geq r_v[t^*]$. Then $\forall v \in V$ on one side $x_v[t^* + 1] \geq x_v[t^*]$ while on the other $r_v[t^* + 1] \leq r_v[t^*]$. Moreover, by the induction hypothesis, in both LTM and PALTM, $x_v[t^* + 1] = 1$ becomes equivalent to $\sum_{w \in N_v} x_w[t^*] \geq r_v[t^*]$, and since

$$\sum_{w \in N_v} x_w[t^* + 1] \geq \sum_{w \in N_v} x_w[t^*] \geq r_v[t^*] \geq r_v[t^* + 1]$$

we have that $x_v[t^* + 1] = 1$ is equivalent to $\sum_{w \in N_v} x_w[t^* + 1] \geq r_v[t^* + 1]$. Therefore, $\forall t \geq 0, \nexists v \in V : x_v[t] = 1$ and $\sum_{N_v} x_w[t] < r_v[t]$: we can apply lemma 1.2 and conclude. \square

In Figure 1.2 we draw a portion of a larger network, with the agents forming a structure with non receding initial activation. The black circles are initially active nodes, while the number inside is their constant activation threshold. None of the three nodes will ever deactivate.

Proposition 1.1 is a direct consequence of Lemma 1.3.



Figure 1.2: If all active at $t = 0$, these three nodes never deactivate.

1.3 Game theoretic models

Decision making in networks can involve strategic choices, with the actors of the network having to answer to the decisions of the neighbors. These situations are suitably represented in the framework of game theory, adapted to a networked setting. As shown in Remarkably, the linear threshold model has a game theoretic motivation that we describe in this Section. We follow the approach of Jackson [23, ch. 9] to discuss the *networked coordination game* of interest, introducing only the required concepts. The book [20] provides a reference to game theory. The concepts necessary here are concisely introduced in [17, ch. 6 and ch. 19] and [28, 36].

To define a game we shall introduce a *set of players*, the *actions* available and their corresponding *payoffs*. We should also specify the order of play. The players are the elements of a finite set V , with $n = |V|$, that will be the node set of the graph G introduced later. Each player can choose an action (or *strategy*) x_v among those available in the a X_v . A *strategy profile* is an element \mathbf{x} of the set $X = \prod_{v \in V} X_v$ which describes univocally the strategy played by each player in V . Here we consider a binary decision framework, with $X_v = \{0, 1\}$ for all player and with $X = \{0, 1\}^V$.

The payoffs are the reward that each player v receives upon the choice of his strategy, given the other players strategies: they are in general described with functions $u_v : X \rightarrow \mathbb{R}$, which specify the game structure. If the strategy profile is \mathbf{x} , the value $u_v(\mathbf{x})$ is the reward of player v . In standard game theory each player “cares” about the behavior of everyone else. Here we consider the class of *graphical games*, where each player cares about the strategies of a subset of the remaining players. The interactions are represented by a directed graph $G = (V, E)$, where edges like $(v, w) \in E$ mean that the player v “cares” about the strategy chosen by w . Often graphs theory considers symmetric graphs. To avoid trivialities, we suppose that the graph is at least weakly connected with *out*-degrees bigger than zero.

Recall that N_v denotes the set of *out*-neighbors of v . Given a strategy profile $\mathbf{x} \in X$ and a player v , the strategies played by the neighbors of v is $x_{N_v} \in X_{N_v} = \prod_{w \in N_v} X_w$. In graphical games, the payoff to each player are

given by functions $u_v : X_v \times X_{N_v} \rightarrow \mathbb{R}$

$$u_v(x_v, x_{N_v}).$$

These functions may differ from player to player, but depend explicitly only on the actions of the player itself and those taken by its neighbors.

In a *networked coordination game* each player v wishes to coordinate its action with the *out*-neighbors and payoffs depend on the number of neighbors playing each strategy. The dependence can be linear: a parameter $q_v \in [0, 1]$ is enough to describe how a player values each strategy, without loss of generality. According to [28,33], when v plays 0 or 1, we respectively grant him the payoffs:

$$\begin{aligned} u_v(0, x_{N_v}) &= q_v \sum_{w \in N_v} (1 - x_w) = q_v (d_v - \sum_{w \in N_v} x_w) \\ u_v(1, x_{N_v}) &= (1 - q_v) \sum_{w \in N_v} x_w. \end{aligned}$$

Overall, the payoff of v is $u_v(x_v, x_{N_v}) = (1 - x_v) u_v(0, x_{N_v}) + x_v u_v(1, x_{N_v})$. This function depends on the specific player v only by the degree d_v and parameter q_v , while the dependence on the neighbors configuration is through the number m_v of neighbors playing 1: $m_v = \sum_{w \in N_v} x_w$. We can simplify the payoff functions to $u_{d_v, q_v} : \{0, 1\} \times \{0, 1, \dots, d_v\} \rightarrow \mathbb{R}$ with $u_{d_v, q_v}(x_v, m_v)$.

The networked coordination game just described has the *strategic complements* property, which is the property of increasing differences: for $m \geq m'$ and any d, q it holds

$$u_{d,q}(1, m) - u_{d,q}(0, m) \geq u_{d,q}(1, m') - u_{d,q}(0, m').$$

Consequently we have $u_{d,q}(1, m) \geq u_{d,q}(0, m)$ according to the threshold condition $m \geq q_v d_v$, with equality if $m = q_v d_v$.

The specific case with $q_v = q$ for all player has been called *semi-anonymous graphical game* [23]. The payoff functions are just $u_{d_v}(x_v, m_v)$ and the threshold conditions only depend on the *out*-degree, with threshold value $q d_v$. Semi-anonymity is given by two aspects: each player just considers the number of neighbors playing 1 or 0 and the only difference between players arise from the graph topology (the *out*-degree).

We have specified the set of players V and the actions profile X with the corresponding payoffs u_v , but we have not yet discussed the timing of the game. In *strategic form games* $\langle V, X, u_v \rangle$ actions are played simultaneously. Each agent has to choose the strategy that maximizes its payoff, given a belief about the actions of the other players but without knowing them. Given a player v and the configuration x_{N_v} of his neighbors, we call *best response* of the player v every strategy in the set

$$B_v = \operatorname{argmax}_{y \in X_v} u_v(y, x_{N_v}).$$

A fundamental concept for the analysis of games is that of *Nash equilibrium*. A (pure strategy) Nash equilibrium of a strategic form graphical game $\langle V, X, u_v \rangle$ is a strategy profile $\bar{\mathbf{x}} \in X$ such that, for all $v \in V$:

$$u_v(\bar{x}_v, \bar{x}_{N_v}) \geq u_v(y, \bar{x}_{N_v}) \quad \forall y \in X_v$$

The equilibrium condition requires that each player chooses among the actions which offer the highest payoff, in response to the actions of his neighbors. In selecting the best possible option, all players are satisfied with their choices and have no interest in changing the adopted strategy.

The networked coordination game with payoff functions $u_{d_v, q_v}(x_v, m_v)$ is truly a coordination game if all the payoff parameters q_v lie strictly in $(0, 1)$. In this case the game has at least two equilibria: one where all players choose 0, $\mathbf{x} = \mathbf{0}$, and one with everyone playing 1, $\mathbf{x} = \mathbf{1}$. Other intermediate equilibria are possible, depending on the interplay between graph topology and payoff parameters.

Strategic form games are static, but it is easy to introduce dynamic features. In fact, it is sufficient to allow players revise their strategies after they have observed those played by the neighbors. If the configuration is not a Nash Equilibrium, the players that are not satisfied with their strategies will change it. These updates are organized in rounds, until some stable configuration is reached.

We define a discrete time dynamic process $\mathbf{x}[t] \in X$, which we call (*deterministic*) *best response dynamic*. Consider a networked coordination game with payoff parameters q_v and initial strategy profile $\mathbf{x}[0] \in X$. The payoff parameters can change (for external reasons) along the process and we stack the time-varying parameters sequences $q_v[t]$ in the vector sequence $\mathbf{q}[t]$. The graph may change as well; here we limit our attention to static graphs.

The deterministic, discrete time process is:

$$\mathbf{x}[t+1] = F_G(\mathbf{x}[t], \mathbf{q}[t])$$

where each component of $F_G : X \times [0, 1]^V \rightarrow X$ is defined by:

$$x_v[t+1] = \operatorname{argmax}_{y \in X_v} u_{d_v, q_v[t]}(y, \sum_{w \in N_v} x_w[t]) \quad (1.3)$$

with 1 preferred to 0 in case of draw (to make each update deterministic). The update rule says that every player performs a *best response* revision of its strategy, with ties resolved in favor of 1¹.

¹Ties may also be solved by a random choice among the best responses. Ties can occur only if $q_v d_v$ is integer. See also [33, footnote 8].

The deterministic best response dynamic with constant payoff parameters \mathbf{q} is an “autonomous” dynamical system

$$\mathbf{x}[t + 1] = F_{G, \mathbf{q}}(\mathbf{x}[t])$$

that has been studied in literature. The dynamic process converges to a Nash equilibrium or to a limit cycle [1]. In fact, with a finite number n of player there are exactly 2^n strategy profiles in X : the deterministic process either converges to an element in X (which has to be a Nash equilibrium since no one is going to change strategy) or repeats a limit cycle in X . The dynamic process may be regarded as a navigation of a larger graph $H = (X, \mathcal{F})$ where the directed edges in \mathcal{F} are induced by $F_{G, \mathbf{q}}$.

With the due adaptations, the deterministic best response dynamic and the linear threshold model (LTM) describe the same dynamic process $\mathbf{x}[t]$, as established in the following lemma. In both models every agent verifies the threshold condition at each step and can switch strategy at any time. The LTM formulation captures the fundamental aspects of the coordination game and highlights the discrete nature of the threshold condition.

Lemma 1.4. *Consider an initial configuration $\mathbf{x}[0] \in X = \{0, 1\}^V$, the collection of threshold sequences $\mathbf{r}[t] \in R$ and the collection of payoff parameter sequences $\mathbf{q}[t] \in [0, 1]^V$. The linear threshold model (1.1) and the deterministic best response dynamic (1.3) are equivalent (describe the same dynamic $\mathbf{x}[t] \in X$) if for every v the sequences $r_v[t]$ and $q_v[t]$ are such that*

$$r_v[t] = \lceil q_v[t] d_v \rceil \quad \forall t \geq 0.$$

Proof. Let us introduce the sums $m_v[t] = \sum_{w \in N_v} x_w[t]$ for every player v . Suppose the process $\mathbf{x}[t]$ coincides until time t for the two models. In the deterministic best response dynamic $x_v[t + 1] = 1$ if and only if the argmax in (1.3) gives either $\{1\}$ or $\{0, 1\}$, which happens if

$$\begin{aligned} u_{d_v, q_v[t]}(1, m_v[t]) &\geq u_{d_v, q_v[t]}(0, m_v[t]) \\ (1 - q_v[t])m_v[t] &\geq q_v[t](d_v - m_v[t]) \\ m_v[t] &\geq q_v[t] d_v \\ m_v[t] &\geq \lceil q_v[t] d_v \rceil \end{aligned}$$

The last line is an equivalence since $m_v[t]$ is integer. Hence, $x_v[t + 1]$ of the deterministic best response coincides with that of the LTM if and only if $r_v[t] = \lceil q_v[t] d_v \rceil$. If this holds for every v , the two set of rules give a coincident update $\mathbf{x}[t + 1]$ and by induction, given the common initial condition, the two process coincide. \square

The linear threshold model described in Section 1.1 is slightly more flexible than the deterministic best response with parameters $q_v[t] \in [0, 1]$. Each thresholds sequence $r_v[t]$ in the LTM belongs to $\{0, 1, \dots, d_v + 1\}$: thresholds are allowed to be explicitly 0 or $d_v + 1$, corresponding to stubborn behaviors. The “ties resolution” of the deterministic best response, maps $q_v[t] = 0$ to $r_v[t] = 0$, but there is no counterpart for $d_v + 1$.

A complete overlap between the two rules, not dependent on the “ties resolution”, is obtained with the introduction of two special values 1^+ and 0^- for $q_v[t]$, to indicate $q_v > 1$ and $q_v < 0$ respectively. We can associate $r_v[t] = d_v + 1$ to $q_v[t] = 1^+$ and $r_v[t] = 0$ to $q_v[t] = 0^-$. Nodes with $q_v[t] \in \{0^-, 1^+\}$ behave as stubborn, and this approach works regardless of the degree d_v (also for trivial players without *out*-neighbors, $d_v = 0$).

To introduce the networked coordination games is it possible to start from *two-player* games [17, 28, 33]. This approach is natural on symmetric graphs: each pair of edges (v, w) and $(w, v) \in E$ means that the end-vertices v, w are involved in a coordination game with two player only. Each agent v in the network is required to use a common strategy in each game with its *out*-neighbors in N_v . The overall payoff of v is the sum of the payoffs from each edge game, and all the agents shall play simultaneously. A possible generalization adopts a weighted graphs and and express the payoff functions as the weighted sum of the edge payoffs.

Some authors motivate the permanent activation linear threshold model (PALTM) with the coordination games described above, used to describe processes like the permanent adoption of a new technology. To be rigorous, the deterministic best response update allows the revision of either actions 0 or 1. However, as explained by the equivalence condition, in case the payoff parameters are constant and the early adopters are stubborns, no “subsequent” adopters would ever switch back. This is indeed the kind of diffusion processes described in [17, 28, 30], e.g. the diffusion of ideas or the adoption of a new technology, where once a node “activates” it cannot return inactive. In [33] the permanent version of the coordination game is used to get insights on the non permanent one.

1.4 Financial networks and contagion

Systemic risk in financial markets has emerged as a recent, major research topic. Financial institutions are nowadays highly interconnected by intricate financial instruments and structures, and their operation with inadequate capital levels introduce systemic risks and instabilities. In the recent financial crisis, for example, the initial losses in one type of asset could quickly spread the the

whole system, threatening its stability. Networks can be used to represent the mutual exposures between financial institutions, to model the propagation of insolvencies and investigate the cascading effects. Researcher are trying to relate the systemic risks of contagion to measurable features of financial institutions and of their interconnections [4]. Better risk estimates shall produce better financial regulations.

In this section we briefly outline the model of financial default contagion proposed by [4] and discuss how, under some simplification, this model can be reduced to the linear threshold model. We will see how *out* and *in*-degree of a node represent the money that an institution has respectively lent to, or borrowed from, other institutions. The activation thresholds are instead related to the capital levels of each institution.

First we shall describe a financial entity, with its *balance sheet*, and define the *insolvency* condition. Later we introduce the *exposure matrix*, which represents the network of money lent and borrowed, and we describe the propagation of insolvencies across such network.

Let start with a single financial entity, say a bank, and stylize its *balance sheet*. The balance sheet or *statement of financial position* is the formal record of the financial activities of the bank. In a structured way, it summarizes the informations about the company's assets and liabilities at a given point in time. Table 1.1 sketches a simplified balance sheet: *Assets* are accounted on the left, *Liabilities* on the right. In every moment, Assets and Liabilities are balanced exactly (i.e., the net sum of Asset minus Liabilities must be zero at any time).

Asset	Liabilities
Loans, Investments	Debts, Deposits
Reserves	Equity

Table 1.1: A simplified balance sheet.

Liabilities include the ownership equity and any borrowed money. The equity, also called *capital*, is the actual money of the bank (its “own skin in the [business] game” [21]). Accepting deposits and issuing bonds, the company borrows further money, to finance its activities. All this money is a liability because the bank shall pay a rate on it. Assets, on the other side, are the funds managed by the bank, together with other possible kind of assets. This money can be invested or lent to make profit, or saved as reserves.

Let consider the following example, inspired by [40, Problem Set 10].

Example. An investment bank is managing \$1 billion, which it invests in various financial instruments (“assets”) related to the housing market (e.g., the infamous “mortgage backed securities”). Because the bank is investing with

borrowed money, its actual assets are only \$50 million (5%). Accordingly, if the bank loses more than 5%, it becomes *insolvent*. (Which means that it will have to be bailed out, and the bankers may need to forgo any huge bonuses for a few months.)

Among the liabilities, there is a fundamental difference between equity and borrowed money. The equity (or *capital*, or *net worth*) is the “actual” money of the bank, while debts and deposits are someone else money. The bank is, in first instance, responsible with its own money for the borrowed money. It can loose its capital, but if it looses more (i.e. the borrowed money), it becomes *insolvent*. In the example, the bank has a capital of \$50 million and manages \$1 billion of assets and investments. As the two side of the balance sheet are always balanced, the borrowed money amounts to \$950 million. Losses on the assets make the net worth decrease, and insolvency happens when the net worth of the institution is negative or zero, i.e. losses have exceeded the capital. Therefore, the capital is the buffer to absorb possible losses on (all) the managed assets. The capital of the bank is \$ 50 million: the smaller the buffer, the smaller the losses the bank can withstand without becoming insolvent.

Technically, insolvency does not lead straight to *default*, which occurs when the institution fails to fulfill scheduled debt payments and short term liabilities. Default is due to *illiquidity*, i.e. lack of cash among the reserves. However, currently the financial sector is primarily funded by short-term debt: an insolvent institution needs to obtain financing to meet the due payments, and this becomes extremely difficult in case of insolvency. In practice, insolvency leads to illiquidity which in turn leads to default unless, of course, a lender of last resort such as the central bank intervenes [15]. All in all, default can be considered as generated by insolvency.

We proceed with the description of the default cascade model: we introduce the counterparty network and define the default contagion process, as in [4]. Consider a set V of banks and a non negative *exposure matrix* $\mathbf{e} \in \mathbb{R}^{V \times V}$, where each entries $\mathbf{e}_{v,w}$ gives the exposure of bank v to bank w in monetary units, i.e. represents the money that v has lent to w . The weighted graph $G = (V, E, \mathbf{e})$ (with the exposure matrix used as weight function) represents the interbank lent and borrowed money.

For a bank $v \in V$, the sum $\sum_w \mathbf{e}_{v,w}$ gives the amount of interbank assets, while the total interbank liabilities are $\sum_w \mathbf{e}_{w,v}$. If D_v and x_v are the deposit and other asset of bank v , bank v 's net worth (i.e. its capital) must be

$$c_v = x_v + \sum_w \mathbf{e}_{v,w} - \sum_w \mathbf{e}_{w,v} - D_v,$$

because Assets and Liabilities need to balance at any time. The balance sheet of bank v is in Table 1.2.

Asset	Liabilities
Interbank assets $\sum_w \mathbf{e}_{v,w}$	Interbank liabilities $\sum_w \mathbf{e}_{w,v}$ Deposits D_v
Other assets x_v	Net worth c_v

Table 1.2: Stylized balance sheet of bank v [4, page 5].

According to [4], we assign to each institution the ratio of capital to interbank assets and (improperly) call it *capital ratio*, although it is not the ratio to total assets:

$$\gamma_v = \frac{c_v}{\sum_w \mathbf{e}_{v,w}}.$$

Due to issues in real economy businesses, external to the financial network, we assume some of the banks in V suffer enough “other assets” losses to become insolvent, i.e. with negative net worth: to these banks we set $\gamma_v = 0$. The obtained pair of exposure matrix and capital ratios defines the *financial network* (\mathbf{e}, γ) on the set V of financial institution and specifies the quantities involved in the financial contagion process.

The set of initially insolvent institutions is given by

$$\mathbb{D}_0(\mathbf{e}, \gamma) = \{v \in V : \gamma_v = 0\}$$

and represents the *fundamental defaults*, due to losses in non-interbank assets. The fundamental defaults can trigger a cascade of *contagion defaults*, those caused by balance sheet insolvency just because other banks in the financial network are not able to keep their promises [4, 18].

Suppose bank v is exposed to bank w of $\mathbf{e}_{v,w}$ when the bank w declares default, i.e. $w \in \mathbb{D}_0(\mathbf{e}, \gamma)$. Let R_w be the *recovery rate* on the assets of w at defaults, i.e. the proportion of $\mathbf{e}_{v,w}$ that the counterparty v can actually redeem after the default of w . From the default of w , the bank v suffers a loss of $(1 - R_w)\mathbf{e}_{v,w}$. If the losses exceed the capital, i.e. $c_v \leq (1 - R_w)\mathbf{e}_{v,w}$, the net worth of bank v becomes zero or negative, and v becomes insolvent. As argued earlier, in this simplified model this led to default, a *default by contagion*. After the fundamental defaults, the nodes that become insolvent are those in the set

$$\mathbb{D}_1(\mathbf{e}, \gamma) = \left\{ v \in V : \gamma_v \sum_w \mathbf{e}_{v,w} \leq \sum_{w \in \mathbb{D}_0(\mathbf{e}, \gamma)} (1 - R_w) \mathbf{e}_{v,w} \right\},$$

where we used $c_v = \gamma_v \sum_w \mathbf{e}_{v,w}$. These defaults may in turn generate further rounds of insolvencies and defaults.

Given the financial network (\mathbf{e}, γ) of $n = |V|$ banks, the *default cascade* is the sequence of sets $\mathbb{D}_k(\mathbf{e}, \gamma)$ defined by

$$\mathbb{D}_k(\mathbf{e}, \gamma) = \left\{ v \in V : \gamma_v \sum_w \mathbf{e}_{v,w} \leq \sum_{w \in \mathbb{D}_{k-1}(\mathbf{e}, \gamma)} (1 - R_w) \mathbf{e}_{v,w} \right\}$$

and triggered by the *fundamental defaults* $\mathbb{D}_0(\mathbf{e}, \gamma) = \{v \in V : \gamma_v = 0\}$. By construction, this sequence is non-decreasing, bounded by V and admits a limit, reached in at most n steps: $\mathbb{D}_0(\mathbf{e}, \gamma) \subseteq \mathbb{D}_1(\mathbf{e}, \gamma) \subseteq \dots \subseteq \mathbb{D}_{n-1}(\mathbf{e}, \gamma) \subseteq V$.

This is the substance of the contagion model proposed in [4], where it is further assumed $R_w = R$ equal for every institution. Other works argue that, to model the default's short term effects, it can be assumed $R = 0$ as the liquidation process takes time [15, 18].

The fundamental question is to understand how much the contagion process will spread in the financial network, i.e. if the cascade will remain confined or involve a large portion of the institutions. It is accepted that inadequate capital levels together with large interconnectedness may entail systemic instability [4], however it remains to be clarified which measurable features of the financial network need to be monitored and properly regulated to limit the risks of contagion.

The default cascade described above is equivalent to a weighted version of the linear threshold model (1.1). Let consider the financial network (\mathbf{e}, γ) on the set of institution V and assume the recovery rate R_w is zero for every $v \in V$. The weighted graph $G = (V, E, \mathbf{e})$ represents the counterparty network and given a bank v let $N_v = \{w \in V : \mathbf{e}_{v,w} > 0\}$ be the set of v 's debtors. Each bank has a state $x_v[t] \in \{0, 1\}$ representing the bank "doing well" / "default" condition. The fundamentally defaulted banks have $\gamma_v = 0$ and we assign $x_v[0] = 1$.

The default cascade model can be expressed as

$$x_v[t+1] = \begin{cases} 1 & \text{if } \sum_{w \in N_v} \mathbf{e}_{v,w} x_w[t] \geq \gamma_v \sum_w \mathbf{e}_{v,w} \\ 0 & \text{otherwise} \end{cases}$$

and has the same structure of (1.1). The capital of the bank, $c_v = \gamma_v \sum_w \mathbf{e}_{v,w}$, serves as activation threshold.

If we further simplify the contagion model, choosing a binary exposure matrix² $\mathbf{e} \in \{0, 1\}^{V \times V}$, we obtain the "not weighted" Linear Threshold Model (1.1). In this last case, the interbank assets are the *out-degree* of the bank $d_v = \sum_w \mathbf{e}_{v,w}$, the interbank liabilities are the *in-degree* $k_v = \sum_w \mathbf{e}_{w,v}$ and the

²Equivalent to the *Adjacency matrix* of the graph, see Appendix A.

activation threshold is again the capital $r_v = \lceil c_v \rceil = \lceil \gamma_v d_v \rceil$. A binary exposure matrix is however a strong assumption in a financial network, because it means uniform exposition. The analysis presented in the following Chapters is valid for this setting, extensions to weighted networks are left to future work.

Chapter 2

The Local Mean-Field approximation

In this Chapter we analyze the activation processes obtained with the Linear Threshold Model with variable thresholds, introduced in Section 1.1 of the previous Chapter. The analysis will be carried out on a random graph ensemble, specifically the *directed configuration model*. On the ensemble, the average activation can be approximated with a scalar recursive equation which we call *Local Mean Field dynamical system*. The result of this Chapter, under the equivalence conditions of Proposition 1.1, can be applied to the Permanent Activation Linear Threshold Model introduced in Section 1.1.

We briefly recall the LTM and introduce the random network ensemble, in Section 2.1. In Section 2.2 we provide a heuristic argument to give some clues on the evolution of the mean-field dynamical system. Section 2.3 formalizes the recursive formulas to compute the Local Mean Field dynamics, using the *tree-like* local structure of the configuration model. The main contribution of Chapter 2 is the concentration theorem of Section 2.4: when deployed on a directed configuration model random graph, the Linear Threshold Model behaves arbitrarily similarly to its mean-field approximation, provided that the number of node is sufficiently large. Finally, we present a few simulations to illustrate the derivations of this Chapter.

2.1 Introduction

We shortly recall the Linear Threshold Model described in Section 1.1, introduce the edge states and define the quantities analyzed in the following of this Chapter. In Subsection 2.1.1 we describe the network model we adopt.

Consider a connected graph $G = (V, E)$ of size $n = |V|$ and with directed

edges. Given a node $v \in V$, its set of *out*-neighbors is $N_v = \{w \in V : (v, w) \in E\}$, its *out*-degree is $d_v = |N_v|$ while the *in*-degree is $k_v = |\{w \in V : (w, v) \in E\}|$. Each node is endowed with a binary state, and $X = \{0, 1\}^V$ is the configuration space. As before, we have a set $R = \{\mathbf{r} \in \mathbb{N}^V : 0 \leq r_v \leq d_v + 1\}$, in which each element is a possible assignment of the activation thresholds to the nodes. Let $\mathbf{x}[0] \in X$ be the initial configuration and consider the threshold sequences $\mathbf{r}[t] \in R$, for $t \in \mathbb{N}$. The Linear Threshold Model prescribes the deterministic update rule for the state vector $\mathbf{x}[t]$. Each agent v updates synchronously its state, according to

$$x_v[t+1] = \begin{cases} 1 & \text{if } \sum_{w \in N_v} x_w[t] \geq r_v[t] \\ 0 & \text{otherwise.} \end{cases} \quad (2.1)$$

The analysis of the LTM requires the introduction of fictitious binary edge states, $x_e[t] \in \{0, 1\}$ to each directed edge $e \in E$. The technical reasons will be clear later. Every edge state agrees with the state of the terminating node of corresponding edge:

$$x_{(w,v)}[t] = x_v[t] \quad \forall t \geq 0, \quad \forall (w, v) \in E,$$

such that the edge state $x_{(w,v)}[t]$ may be interpreted as a message that v sends to w , about his state $x_v[t]$. Whenever necessary, we will refer to the collection of all the edge states as $\mathbf{x}_e[t] \in \{0, 1\}^E$.

We will analyze the activation dynamic on a large network, and we are interested in the aggregate behaviors. The fraction of active agents is:

$$a[t] = \frac{1}{n} \sum_{v \in V} x_v[t]$$

whereas the corresponding fraction of “active edges” is:

$$m[t] = \frac{1}{|E|} \sum_{e \in E} x_e[t] = \frac{1}{|E|} \sum_{v \in V} k_v x_v[t]$$

These two quantities will be approximated by the mean-field recursive equation.

2.1.1 The network model

We shortly describe our random network model, based on the directed version of the configuration model. The procedure to prepare the network is based on two steps: first we identify “create” n agents with their attributes, then we extract a random wiring to get the network.

1. Consider a set V of n agents. To each one, we assign two number the *out*-degree d_v and the *in*-degree k_v . The *out* and *in*-degree sequences (the collection of all the *out*-degree and *in*-degree) need to be *compatible*, i.e. $\sum_v d_v = \sum_v k_v$. We can already assign to each node v its activation threshold sequence $r_v[t]$ (suppose you already know it) and its initial state $x_v[0]$. Recall that $\mathbf{r}[t]$ need to belong to the set R , while $\mathbf{x}[0] \in \{0, 1\}^V$.

At this point we just miss the wiring, the set E which describes how the node are connected in the network. In the choice of E lies all the randomness, the configuration model multigraph ensemble coincides with the collection of all possible E s. We can imagine every node v having d_v “plugs” and k_v “sockets”: we shall assign to each plug a socket, selected uniformly at random. We already know that the set E has $|E| = \sum_v d_v = \sum_v k_v$ elements, hence we can label all the the *out*-plugs and all the *in*-sockets with the integers $\{1, 2, \dots, |E|\}$.

2. To obtain a specific instance of the configuration model, we choose a random permutation π , on the set $\{1, 2, \dots, |E|\}$. We wire plugs and sockets according to $(i, \pi(i))$: the set of all the corresponding edges in the form (v, w) forms E . The procedure produces the multigraph $G = (V, E)$, which is fine for our analysis.

Notice that, once the degree sequences are established, the set Π of all permutations on $\{1, 2, 3, \dots, |E|\}$ is in one-to one correspondence with the ensemble \mathcal{G} of all multigraphs with such degree sequences. The specific element $\pi \in \Pi$, chosen with uniform probability, gives an an instance of \mathcal{G} as it completely specifies the edge set E of $G = (V, E)$. We call this random graph model *directed configuration model*.

The configuration model is used to generalized the so called *Poisson random graphs*, to incorporate non-Poisson degree distributions [34]. Usually, the *out* and *in*-degree sequences are not defined but rather extracted from a known joint degree distributions, which shall satisfy some properties. To simplify the picture, we instead compute the statistics on the degree sequences and on the threshold sequences, as use that as distributions for the following work.

The number of nodes of with *out*, *in*-degrees d, k is $n_{d,k} = |\{v \in V : d_v = d, k_v = k\}|$, while that of nodes which also have activation threshold r at time t is $n_{d,k,r}[t] = |\{v \in V : d_v = d, k_v = k, r_v[t] = r\}|$. From these numbers, we define the joint statistic on the degrees

$$p_{d,k} = \frac{n_{d,k}}{n} = \mathbb{P}(d_v = d, k_v = k)$$

and the conditional statistic on the thresholds, at time t

$$p_{r|d,k}[t] = \frac{n_{d,k,r}[t]}{n_{d,k}} = \mathbb{P}(r_v[t] = r | d_v = d, k_v = k)$$

We will refer to $p_{d,k}$ and $p_{r|d,k}[t]$ as the joint distribution of the degrees and the conditional distribution of the thresholds, of a generic of G .

Remark 2.1. *The statistics $p_{d,k}$ and $p_{r|d,k}[t]$ are computed after degree and threshold sequences have been decided. It is possible to use prior distribution, to sample the degree and thresholds, with the hypothesis that the statistics converges to the prior distribution as $n \rightarrow \infty$. A further concentration result would then be required, which adds little novelty to the picture.*

2.2 A heuristic argument

The local mean field dynamic equations will be formally proved in the following Section 2.3. Here, we propose a heuristic argument to give an example about the meaning of the mean-field dynamic equation on the directed configuration model ensemble.

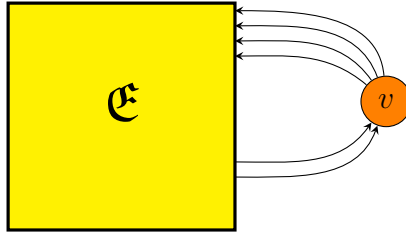
We need to consider the configuration model in an ensemble sense. To do this, we specify all the features of the n agents of V (the compatible *in* and *out*-degree sequences \mathbf{d} and \mathbf{k} , the collection of threshold sequences $\mathbf{r}[t] \in R$ and the initial activation profile $\mathbf{x}[0] \in X$) but we do not specify any wiring of the network. In short, we only do the “step 1” described in Subsection 2.1.1.

The statistics $p_{d,k}$ and $p_{r|d,k}[t]$ can be computed without knowing the specific wiring of the graph, that would be encoded in the set E . In fact we only know the number of edges contained in any possible E , $|E|$, that can be expressed as $|E| = n \sum_{d,k} d p_{d,k} = n \sum_{d,k} k p_{d,k}$.

Before the heuristics, we describe a trivial fact. Imagine a black box called \mathfrak{E} (see figure 2.1) that contains $|E|$ objects (which are the unspecified edges) and suppose we are at time t . In the same moment, every node $v \in V$ gets without replacement d_v objects from the box \mathfrak{E} , and, before time $t + 1$, returns k_v objects to the box. Let $|E|_t$ denote the number of objects in the black box just before the sampling at time t . As the degree sequences are compatible, the number of objects in the box remains trivially constant:

$$|E|_{t+1} = |E|_t + \sum_{d,k} n p_{d,k} (-d + k) = |E|_t = |E|.$$

Recall that each object in \mathfrak{E} has a state $x_e[t] \in \{0, 1\}$, even if the edges e have not been assigned. In this Section we denote the average of the black box

Figure 2.1: The black-box \mathfrak{E} , which is actually yellow!

objects states with $\mu[t]$, i.e. $\mu[t] = \sum_{e \in E} x_e[t]$. In the next Section $\mu[t]$ will be defined more formally and used with that meaning from there on. From the initial configuration $\mathbf{x}[0] \in \{0, 1\}^V$ and the *in*-degree sequence, $\mu[0]$ is determined: $\mu[0] = (|E|)^{-1} \sum_{v \in V} k_v x_v[0]$.

To imagine the average evolution of the activation process, imagine each node v sampling d_v random objects from the black box and subsequently returning k_v object to the box. Imagine that every node to this operation at the same instant: on average v would pick $d_v \mu[t]$ objects with state 1 and return $k_v f_{d_v, r_v[t]}(\mu[t])$ objects again with state 1, where $f_{d,r}(x) = \sum_{i=r}^d \binom{d}{i} x^i (1-x)^{d-i}$.

Hence, with all the nodes taken together, the number of 1 states, $|E| \mu[t]$, evolves as

$$|E| \mu[t+1] = |E| \mu[t] + \sum_{d,k,r} n p_{d,k} p_{r|d,k}[t] (-d\mu[t] + k f_{d,r}(\mu[t]))$$

$$|E| \mu[t+1] = \sum_{d,k,r} n k p_{d,k} p_{r|d,k}[t] f_{d,r}(\mu[t])$$

Then, on average, the fraction of active objects inside the black box \mathfrak{E} is:

$$\mu[t+1] = \sum_{d,k,r} \frac{k}{\bar{k}} p_{d,k} p_{r|d,k}[t] f_{d,r}(\mu[t])$$

where \bar{k} is the average *in*-degree.

This heuristic approach makes explicit that all the randomness of the process is based on the graph wiring. Using the “ensemble” graph already gives the expectations. Since all the objects in the black box \mathfrak{E} are reshuffled every time, and sampled without replacement all together, is it ok to imagine that each of the d_v samples obtained by node v are independent and use the Binomial distribution. Moreover, with n large, the average gets a statistical significance.

Finally, notice that the black box \mathfrak{E} actually mimics the activation process on a complete graph, with n large¹, where at each step every node v watches

¹to avoid multiple samplings and self-sampling.

the state of d_v nodes chose at random and updates accordingly its state, with the restriction that its state was watched by exactly k_v nodes.

2.3 The formal derivation

In this Section we introduce more formally the quantities $\mu[t]$ and $\nu[t]$, and prove the recursive equation described heuristically in the previous Section. We will precisely define the *depth ℓ directed neighborhood* of node v , which, on a large configuration model, has a tree-structure with high probability. Then, we define $\mu[t]$ and $\nu[t]$ as the expected activation of each node with an *out-neighborhood* which is a tree for depth at least t . Finally, we prove the recursion for $\mu[t]$ and consequently get $\nu[t]$.

Let call the length of the shortest path from u to v with $\text{dist}_G(u, v)$. If such a path does not exist, we put $\text{dist}_G(u, v) = \infty$, while when $u = v$, $\text{dist}_G(u, u) = 0$. Basically, $\text{dist}_G(u, v)$ the minimal number of directed edges in E necessary to go from u to v ². For our purposes, the *depth ℓ out-neighborhood* $\mathcal{N}_{u,+}^\ell$ of node u is the subgraph $\mathcal{N}_{u,+}^\ell = (U, \tilde{E})$ of G , with:

$$U = \{v \in V : \text{dist}_G(u, v) \leq \ell\}$$

$$\tilde{E} = \{(w, v) \in E : \text{dist}_G(u, w) \leq \ell - 1\}.$$

Notice that $\mathcal{N}_{u,+}^0 = (u, \emptyset)$ and that it is necessary to specify also \tilde{E} , as $\mathcal{N}_{u,+}^\ell$ in general is not the subgraph induced by U on G . Refer to the appendix A for further details and clarification.

We are now ready to define the quantities involved in the local mean field dynamic LMF-Dyn, the *edge dynamic* $\mu[t]$ and the *vertex dynamic* $\nu[t]$,

$$\mu[t] = \mathbb{P}(x_{(v,u)}[t] = 1 \mid \mathcal{N}_{u,+}^{t-1} \text{ is a tree}) \quad (2.2)$$

$$\nu[t] = \mathbb{P}(x_u[t] = 1 \mid \mathcal{N}_{u,+}^{t-1} \text{ is a tree}) \quad (2.3)$$

which will approximate $m[t]$ and $a[t]$ respectively.

The key point about the LMF-Dyn is that the states of the outgoing edges of a node (or, from the other point of view, the state of the incoming messages) will be modeled as independent and identically distributed random variables. Independence comes from the *out-neighborhood* of a node being a tree, since on the different branches we have independent processes. The directed configuration model exhibits a tree-like local structure: as the size n becomes large, the depth ℓ neighborhood of the nodes are trees with high probability, for depths

²technically, in a general directed graph this is not a metric, because the symmetry property does not hold.

of order $\log n$. The identical distribution is related to the random matching as we expect that the network looks the same around every node.

We provide, with the following lemma, the recursive law for $\nu[t]$ and $\mu[t]$. Preliminarily we observe that, once we have specified $\mathbf{x}[0] \in \{0, 1\}^V$, even without a specific wiring, we already know that the fraction of initially active edges is $m[0] = \sum_v k_v x_v[0]$. Moreover, regardless of the wiring, $\mu[0] = m[0]$.

Lemma 2.2. *Consider a directed configuration model network $G = (V, E)$, prepared according to the procedure described before. Suppose its degree distribution is $p_{d,k}$ and its threshold conditional distribution is $p_{r|d,k}[t]$. Let $\mu[t]$ and $\nu[t]$ be the quantities defined in (2.2) and (2.3), with $\mu[0]$ and $\nu[0]$ given. Then, for every $t \geq 0$*

$$\mu[t+1] = \sum_{d,k,r} \frac{k p_{d,k}}{\bar{k}} p_{r|d,k}[t] f_{d,r}(\mu[t]) \quad (2.4)$$

$$\nu[t+1] = \sum_{d,k,r} p_{d,k} p_{r|d,k}[t] f_{d,r}(\mu[t]) \quad (2.5)$$

where $f_{d,r}(x) = \sum_{i=r}^d \binom{d}{i} x^i (1-x)^{d-i}$.

Proof. Remark: In the present form, this proof holds only for a regular graph. The problem³ is that the quantities $\mu[t]$ and $\nu[t]$, defined in equation (2.2) and equation (2.3) respectively, can be interpreted as conditional probabilities only in such a case. Most of the material in the next Chapters regards regular networks, for which the equations are valid.

We are currently working on a reformulation of those definitions and on a new proof suitable for non regular graphs. We envision that a possible solution could be to use a ‘‘Branching Process’’ network [16] suitably defined. The branching process network will approximate the neighborhood of a generic node in the directed configuration model random network. On one hand, we shall study the dynamic as seen from the root node of the branching process network; on the other hand, we need to bound the differences between the branching process network and the original random network.

Due to the random wiring of the graph, at the beginning each edge is independently active with probability $\mu[0]$. Notice also that $\mathcal{N}_{u,+}^0 = (u, \emptyset)$ is trivially a tree. We prove (2.4) by induction.

Given the definition of $\mu[t+1] = \mathbb{P}(x_{(v,u)}[t+1] = 1 \mid \mathcal{N}_{u,+}^t \text{ is tree-like})$, suppose the edge (v, u) terminates in a node u with activation threshold $r_u[t]$ and d_u out-neighbors w_i . Since the subgraph $\mathcal{N}_{u,+}^t$ is a tree by hypothesis, each

³This problem has been pointed out during the reviewing process of this Dissertation by the external reviewer Julien Hendrickx.

subgraph $\mathcal{N}_{w_i,+}^{t-1}$ is a tree and, moreover, for each pair of *out*-neighbors $w_j \neq w_i$ the *out*-neighborhoods do not intersect $\mathcal{N}_{w_i,+}^{t-1} \cap \mathcal{N}_{w_j,+}^{t-1} = (\emptyset, \emptyset)$. Therefore, the states $x_{(u,w_i)}[t]$ are all i.i.d. Bernoulli random variable with success probability $\mu[t]$. As $\mathcal{N}_{u,+}^t$ is a tree by hypothesis, these random variables are also independent from the specific pattern $r_u[s]$, $s \in \{0, 1, t-1\}$. Then, the probability that u is active at $t+1$ is $f_{d_u, r_u[t]}(\mu[t])$.

It remains to average among the probability that the edge (v, u) terminates in a node u with *out*-degree d , *in*-degree k and threshold $r[t]$. This probability is $\frac{k p_{d,k}}{\bar{k}} p_{r|d,k}$, where the factor k/\bar{k} comes from the proportionally higher probability to be wired with an higher *in*-degree node. Hence,

$$\mu[t+1] = \sum_{d,k,r} \frac{k p_{d,k}}{\bar{k}} p_{r|d,k}[t] f_{d,r}(\mu[t])$$

The proof of $\nu[t]$ is similar, with the exception that the final averaging is on vertices themselves and the right distribution is $p_{d,k} p_{r|d,k}$. \square

We remark that the lemma is based on the definitions of $\mu[t]$ and $\nu[t]$ which guarantee independence and identical distribution among the *out*-neighbors of each involved node. Moreover, the tree structure makes the recursion valid even for time varying thresholds. In general, $\mu[t]$ and $\nu[t]$ are two distinct quantities. They become coincident if the distributions of the *in*-degree is independent from *out*-degree and thresholds (say $p_{r|d,k}[t] p_{d,k} = p_{r|d}[t] p_d p_k$), as in a regular network for example.

In the LMF-Dyn, agents with the same *out*-degree and threshold have the same probability of being active, as they share the same $f_{d,r}(\mu[t])$. This non linear recursive equations hold for the directed configuration model.

Within the tree-like approximation, we could write a recursive law to approximate the evolution of the infected fraction of nodes, which is accurate for time horizons of order $\log n$. We exploit this feature to study the infection process, since for time horizons that scale with $\log n$ cycles play little role.

The most important fact is that the recursive equations describe the correct dynamic of the infection when the network is infinite. On the other hand, when the random network is finite but large, the process concentrates around that dynamic, as proved in the following Section.

2.4 The concentration theorem

We have a concentration theorem for the evolution of the active fraction of edges $m[t]$ around the dynamic described by the recursive equation of $\mu[t]$, which holds for large directed configuration model networks. The proof is

inspired by the result of [37], developed in the context of low-density parity-check codes. First, with the Azuma inequality we show the concentration of $m[t]$ around its expected value $\mathbb{E}[m[t]]$ taken over all the possible wiring of G . Some difficulties lie in finding the right bounding constants while exposing the edges of G . This bounding constants gives a worst case scenario, independent of the actual thresholds, hence valid in the general variable threshold case. Second, when n is large, since the graph is tree-like with high probability for short depths, $\mathbb{E}[m[t]]$ is not far from $\mu[t]$.

In the core of the proof we will use not only the depth ℓ *out*-neighborhood, but also the depth ℓ *in*-neighborhood counterpart, which is defined accordingly.

Theorem 2.3. *Consider a directed configuration model network $G = (V, E)$, with $n = |V|$ vertex and $|E|$ edges. The average degree is $\bar{k} = |E|/n$, while the maximum out-degree and in-degree are d_{max} and k_{max} .*

The set \mathcal{G} is the ensemble of all multigraphs which are proper matching of the n vertexes of G , such that G may be thought as drawn from \mathcal{G} with uniform probability.

Consider the linear threshold model (1.1), with the state of the vertex v for time $t \geq 0$ being $x_v[t] \in \{0, 1\}$. In the graph G , assign to each edge $(w, v) \in E$ an binary state that copies the one of the terminating node, i.e. $x_{(w,v)}[t] = x_v[t]$, for all t . The fraction of edges with active state at time t is $m[t] = (|E|)^{-1} \sum_e x_e[t]$.

The expectation $\mathbb{E}[m[t]]$ is the expected value of $m[t]$ on the ensemble \mathcal{G} . Let $\mu[t] = \mathbb{P}(x_{(v,u)}[t] = 1 \mid \mathcal{N}_{u,+}^{t-1} \text{ is a tree})$ be the probability that and edge $e = (v, u)$ has infected status at time t , given that $\mathcal{N}_{u,+}^{t-1}$ is a tree.

It holds:

[Concentration of $m[t]$ around $\mathbb{E}[m[t]]$] For any $\epsilon > 0$ we have

$$\mathbb{P}(|m[t] - \mathbb{E}[m[t]]| > \epsilon/2) \leq 2e^{-\epsilon^2 n \delta} \quad (2.6)$$

[Convergence of $\mathbb{E}[m[t]]$ to $\mu[t]$] For any $\epsilon > 0$ and $n > \frac{2\gamma}{\epsilon}$ we have

$$|\mathbb{E}[m[t]] - \mu[t]| < \epsilon/2 \quad (2.7)$$

[Concentration of $m[t]$ around $\mu[t]$] For any $\epsilon > 0$ and $n > \frac{2\gamma}{\epsilon}$ we have

$$\mathbb{P}(|m[t] - \mu[t]| > \epsilon) \leq 2e^{-\epsilon^2 n \delta} \quad (2.8)$$

with $\delta = \bar{k}(16k_{max}^{2t})^{-1}$ and $\gamma = k_{max}d_{max}^{2t}$

Proof. We introduce some more notation, used in the proof, with additional comments.

Let $M[t] = \sum_e x_e[t]$ be the number of edges with infected status in G , such that $m[t] = M[t]/|E|$.

Let Π be the set of all permutations of $\{1, 2, \dots, |E|\}$, and σ, ρ some generic elements. The ensemble of networks \mathcal{G} coincides with the set Π , since each of the permutation $\sigma \in \Pi$ defines a specific pairing of the *out*-links and *in*-links of the network. There are $|E|!$ distinct permutation in Π , each described by $|E| - 1$ information.

Let define a partial equivalence relation $=_i$ on Π , ordered by refinement. Given $\sigma, \rho \in \Pi$, σ and ρ are equivalent up to i iff they coincide at least on the first i positions. In short: $\sigma =_i \rho \Leftrightarrow \sigma_j = \rho_j \forall j \leq i$.

Let $\pi \in \Pi$ be the specific permutation describing G . Consider the following two families of subsets of Π (corresponding to subsets of \mathcal{G}):

$$\begin{aligned}\Pi_{\pi,i} &= \{\sigma \in \Pi : \sigma =_i \pi\} \\ \Pi_{\pi,i}^j &= \{\sigma \in \Pi_{\pi,i} : \sigma_{i+1} = j\}\end{aligned}$$

with cardinality $|\Pi_{\pi,i}| = (|E| - i)!$ and $|\Pi_{\pi,i}^j| = (|E| - i - 1)!$. $\Pi_{\pi,i}^j$ are disjoint subsets, with $\Pi_{\pi,i} = \bigcup_j \Pi_{\pi,i}^j$. Notice finally that $\Pi_{\pi,0} = \Pi$ and that $\Pi_{\pi,|E|} = \{\pi\} = \Pi_{\pi,|E|-1}$.

Take $\sigma \in \Pi$, and let $H = (V, E_\sigma)$ be the graph associated to σ , $H \in \mathcal{G}$. Consider an instance of the threshold model process on the graph H , and define:

$$\begin{aligned}M_\sigma[t] &= |\{e \in E_\sigma : x_e[t] = 1\}| = \sum_{e \in E_\sigma} x_e[t] \\ M_{\sigma,i}[t] &= \mathbb{E}[M_\rho[t] | \rho \in \Pi_{\sigma,i}] = \frac{1}{(|E| - i)!} \sum_{\rho \in \Pi_{\sigma,i}} M_\rho[t]\end{aligned}$$

We can write $M_{\pi,i}[t]$ in the following equivalent way:

$$M_{\pi,i}[t] = \mathbb{E}[M_\sigma[t] | \sigma \in \Pi_{\pi,i}] = \sum_{j=1}^{|E|} \mathbb{E}[M_\sigma[t] | \sigma \in \Pi_{\pi,i}^j] \mathbb{P}(\sigma \in \Pi_{\pi,i}^j | \sigma \in \Pi_{\pi,i}) \quad (2.9)$$

For some $j \in \{1, 2, \dots, |E|\}$, $\Pi_{\pi,i}^j = \emptyset$, therefore:

$$\mathbb{P}(\sigma \in \Pi_{\pi,i}^j | \sigma \in \Pi_{\pi,i}) = \begin{cases} 0 & \text{if } \exists k \leq i : \pi_k = j \\ \frac{(|E|-i-1)!}{(|E|-i)!} = \frac{1}{|E|-i} & \text{otherwise} \end{cases}$$

Observe finally that $M[t] = M_\pi[t] = M_{\pi,|E|}[t]$, while $\mathbb{E}[M[t]] = M_{\pi,0}[t] = \frac{1}{|E|!} \sum_{\sigma \in \Pi} M_\sigma[t]$.

To prove the bound (2.6) of $\mathbb{P}(|m[t] - \mathbb{E}[m[t]]| > \epsilon/2)$ we can equivalently work on $\mathbb{P}(|M[t] - \mathbb{E}[M[t]]| > |E|\epsilon/2)$. The proof is based on the Azuma inequality⁴. Notice that $M_{\pi,i}[t]$, with $i \in \{0, 1, \dots, |E|\}$ is by construction a martingale sequence (called Doob's Martingale Process). Therefore we need to compute the bounds

$$|M_{\pi,i+1}[t] - M_{\pi,i}[t]| \leq \alpha_i \quad \forall i \in \{0, 1, \dots, |E| - 1\} \quad (2.10)$$

where we already know that $\alpha_{|E|-1} = 0$. The core idea is to expose the edges of G one at a time. Let expand the first member of (2.10) with (2.9), and bound it with the worst cases:

$$\begin{aligned} |M_{\pi,i+1}[t] - M_{\pi,i}[t]| &= |\mathbb{E}[M_\sigma[t] | \sigma \in \Pi_{\pi,i+1}] - \mathbb{E}[M_\sigma[t] | \sigma \in \Pi_{\pi,i}]| \\ &\leq \max_{j \in \psi_i} |\mathbb{E}[M_\sigma[t] | \sigma \in \Pi_{\pi,i}^j] - \mathbb{E}[M_\sigma[t] | \sigma \in \Pi_{\pi,i}]| \\ &\leq \max_{j,k \in \psi_i} |\mathbb{E}[M_\sigma[t] | \sigma \in \Pi_{\pi,i}^j] - \mathbb{E}[M_\sigma[t] | \sigma \in \Pi_{\pi,i}^k]| \end{aligned} \quad (2.11)$$

with $\psi_i = \{\pi_l : l \geq i + 1\}$. We need an upper bound of

$$\left| \mathbb{E}[M_\sigma[t] | \sigma \in \Pi_{\pi,i}^j] - \mathbb{E}[M_\sigma[t] | \sigma \in \Pi_{\pi,i}^k] \right| \quad (2.12)$$

when both

$$\mathbb{P}(\sigma \in \Pi_{\pi,i}^j | \sigma \in \Pi_{\pi,i}) \neq 0 \quad \mathbb{P}(\sigma \in \Pi_{\pi,i}^k | \sigma \in \Pi_{\pi,i}) \neq 0$$

meaning that $j, k \in \psi_i$. The bound should be independent from j, k .

Let define a map $\phi_{j,k} : \Pi_{\pi,i}^j \rightarrow \Pi_{\pi,i}^k$ as follows. Take $\sigma \in \Pi_{\pi,i}^j$, and consider the position h such that $\sigma_h = k$. Recall $\sigma_{i+1} = j$. Then, the permutation $\phi_{j,k}(\sigma)$ agrees on all positions with σ , except in $i + 1$ and h , where $(\phi_{j,k}(\sigma))_{i+1} = k$, $(\phi_{j,k}(\sigma))_h = j$. Substantially, $\phi_{j,k}$ is the wiring exchange of a pair of directed edges, as depicted in figure 2.2. Notice that $\phi_{j,k}$ is a bijection (with inverse $\phi_{k,j}$)

⁴ *Azuma inequality*: Let Z_0, Z_1, \dots be a martingale sequence such that for each $k \geq 0$ holds $|Z_{k+1} - Z_k| \leq \alpha_k$, where the constant α_k may depend on k . Then, for all $l \geq 1$ and any $\lambda > 0$

$$\mathbb{P}(|Z_l - Z_0| \geq \lambda) \leq 2e^{-\frac{\lambda^2}{2 \sum_{k=0}^{l-1} \alpha_k^2}}$$


 Figure 2.2: the map $\phi_{j,k}$

which, since every edge-labeled graph in the ensemble has uniform probability, preserves probabilities. Therefore

$$\mathbb{E} \left[M_\sigma[t] \mid \sigma \in \Pi_{\pi,i}^k \right] = \mathbb{E} \left[M_{\phi_{j,k}(\sigma)}[t] \mid \sigma \in \Pi_{\pi,i}^j \right]$$

which we plug in (2.12) to get⁵:

$$\begin{aligned} & \left| \mathbb{E} \left[M_\sigma[t] \mid \sigma \in \Pi_{\pi,i}^j \right] - \mathbb{E} \left[M_{\phi_{j,k}(\sigma)}[t] \mid \sigma \in \Pi_{\pi,i}^j \right] \right| = \\ & = \left| \mathbb{E} \left[M_\sigma[t] - M_{\phi_{j,k}(\sigma)}[t] \mid \sigma \in \Pi_{\pi,i}^j \right] \right| \\ & \leq \mathbb{E} \left[\left| M_\sigma[t] - M_{\phi_{j,k}(\sigma)}[t] \right| \mid \sigma \in \Pi_{\pi,i}^j \right] \\ & \leq \max_{\sigma \in \Pi_{\pi,i}^j} \left| M_\sigma[t] - M_{\phi_{j,k}(\sigma)}[t] \right| \end{aligned} \quad (2.13)$$

To bound $\left| M_\sigma[t] - M_{\phi_{j,k}(\sigma)}[t] \right|$ we have to compute the number of messages that would be different, at time t , if the graph was wired with $\phi_{j,k}(\sigma)$ instead of σ , in the worst case. Let v, w be the vertexes to which the *out*-sockets $i+1, h$ belong, with $v \neq w$ in general. At time t , the rewiring given by $\phi_{j,k}$ has influenced the vertexes and edges of $\mathcal{N}_{v,-}^{t-1}$ and $\mathcal{N}_{w,-}^{t-1}$. In the worst case $\mathcal{N}_{v,-}^{t-1}$ contains $\sum_{l=1}^{t-1} k_{max}^l$ edges, therefore

$$\left| M_\sigma[t] - M_{\phi_{j,k}(\sigma)}[t] \right| \leq 2 \sum_{l=1}^{t-1} k_{max}^l \leq \frac{2k_{max}^t}{k_{max} - 1} \quad (2.14)$$

Notice now that the quantity in (2.14) by (2.13) bounds each of the terms (2.12) in (2.11). Moreover, this worst case bound is independent of the specific thresholds of the vertexes involved and how they may have been varied. Then we have found the coefficients α_i of the Azuma inequality (2.10):

$$\begin{aligned} \alpha_i &= \frac{2k_{max}^t}{k_{max} - 1} \quad \forall i \in \{0, 1, \dots, |E| - 2\} \\ \alpha_{|E|-1} &= 0 \end{aligned}$$

We have

$$\sum_{i=0}^{|E|-1} \alpha_i^2 \leq |E| \frac{4k_{max}^{2t}}{(k_{max} - 1)^2} \leq 4|E|k_{max}^{2t}$$

⁵ Recall that, for a random variable W , it holds $|\mathbb{E}[W]| \leq \mathbb{E}[|W|]$.

that finally gives the desired result

$$\mathbb{P}\left(|M[t] - \mathbb{E}[M[t]]| > |E|\frac{\epsilon}{2}\right) \leq 2 \exp\left(-\frac{(|E|\epsilon/2)^2}{4|E|k_{max}^{2t}}\right) = 2 \exp\left(-\frac{|E|\epsilon^2}{16k_{max}^{2t}}\right)$$

To get (2.7), we need to bound $|\mathbb{E}[m[t]] - \mu[t]|$. Using linearity, we recognize that $\mathbb{E}[m[t]] = (|E|)^{-1} \sum_e \mathbb{E}[x_e[t]] = \mathbb{E}[x_e[t]] = \mathbb{P}(x_{(v,u)}[t] = 1)$ where $(v, u) \in E$. Since $\mu[t] = \mathbb{P}(x_{(v,u)}[t] = 1 | \mathcal{N}_{u,+}^{t-1}$ is a tree) we can expand $\mathbb{P}(x_{(v,u)}[t] = 1)$ as:

$$\begin{aligned} \mathbb{P}(x_{v,u}[t] = 1) &= \mathbb{P}(x_{(v,u)}[t] = 1 | \mathcal{N}_{u,+}^{t-1} \text{ is a tree}) \mathbb{P}(\mathcal{N}_{u,+}^{t-1} \text{ is a tree}) \\ &\quad + \mathbb{P}(x_{(v,u)}[t] = 1 | \mathcal{N}_{u,+}^{t-1} \text{ is not a tree}) \mathbb{P}(\mathcal{N}_{u,+}^{t-1} \text{ is not a tree}) \end{aligned}$$

From lemma A.1 we have:

$$\begin{aligned} 1 - \frac{\gamma}{n} &\leq \mathbb{P}(\mathcal{N}_{u,+}^{t-1} \text{ is a tree}) \leq 1 \\ 0 &\leq \mathbb{P}(\mathcal{N}_{u,+}^{t-1} \text{ is not a tree}) \leq \frac{\gamma}{n} \end{aligned}$$

with $\gamma = k_{max}d_{max}^{2t}$. As every probability belongs to $[0, 1]$, we have:

$$\begin{aligned} \mathbb{E}[m[t]] &\leq \mu[t] + \frac{\gamma}{n} \\ \mathbb{E}[m[t]] &\geq \mu[t] \left(1 - \frac{\gamma}{n}\right) \geq \mu[t] - \frac{\gamma}{n} \end{aligned}$$

that is

$$|\mathbb{E}[m[t]] - \mu[t]| \leq \frac{\gamma}{n}$$

Choosing $n > 2\gamma/\epsilon$ we get (2.7).

Finally (2.8) follows straight from (2.6) and (2.7) □

2.5 Examples and comments

We conclude the Chapter with a few simple examples that show some aspects of the LMF-Dyn recursions and of the concentration theorem.

The random network model, the directed configuration model ensemble, has been described in all its generality in Subsection 2.1.1. In this Section we restrict to an homogeneous case, a regular graph with all the nodes sharing the same activation threshold. In short, for the graph $G = (V, E)$ with $n = |V|$ nodes, we assume

$$\begin{aligned} \forall v \in V : \quad d_v &= k_v = d \\ \forall v \in V : \quad r_v &= r. \end{aligned}$$

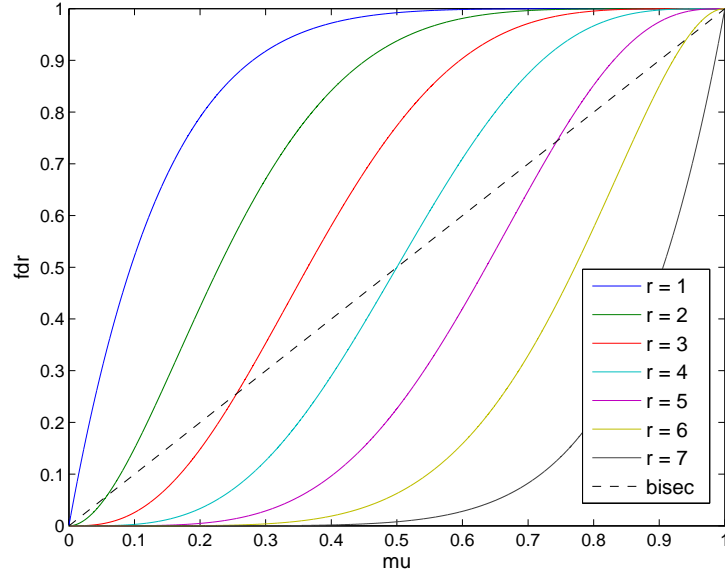


Figure 2.3: The LMF-Dyn (2.15) of a directed regular homogenous network, with $d = 7$ and various values of r .

We also assume $\alpha \in [0, 1]$ as the fraction of initially active agents, i.e. there are $\lceil \alpha n \rceil$ agents initially active in the network. Since G is regular, also the fraction of initially active edges is α , hence $m[0] = a[0] = \alpha$.

With these assumptions, the LMF-Dyn equations (2.4) and (2.5) become coincident and simplify to

$$\mu[t + 1] = f_{d,r}(\mu[t]), \quad \nu[t] = \mu[t] \quad (2.15)$$

where $f_{d,r}(x) = \sum_{i=r}^d \binom{d}{i} x^i (1-x)^{d-i}$. The initial condition is $\mu[0] = \nu[0] = \alpha$. In the following discussion we will only use the state variable $\mu[t]$, as it coincides with $\nu[t]$. Moreover with the short *activation* we will refer either to the fraction of activated agents and to the fraction of activated edges.

The plot in figure 2.3 represent the functions $f_{d,r}(\mu)$ with $d = 7$, for several values of r . The trivial cases with $r = 0$ and $r = d + 1 = 8$ are not represented, as they give the constant values of 1 and 0 respectively. In the examples that follow we will always use

$$d = 7 \quad r = 3.$$

The LMF-Dyn recursion (2.15) of the homogeneous network is completely specified by the choice of $d = 7$ and $r = 3$. To compare the predictions with some actual instances of the activation process, we shall specify the graphs $G = (V, E)$, by choosing the size n and extracting a random wiring E .

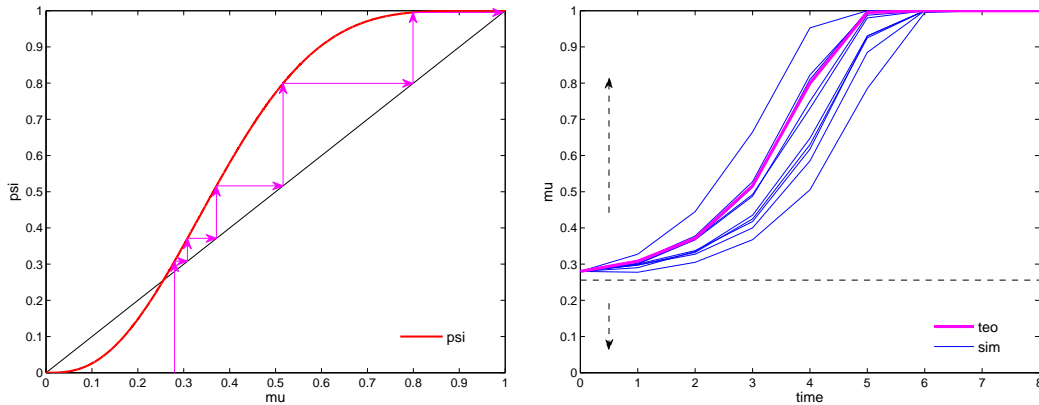


Figure 2.4: A directed regular homogenous network with $d = 7$ and threshold $r = 3$. Left: plot of the LMF-Dyn (2.15) which is simply $f_{d,r}$. Right: comparison with several simulations on random graphs with $n = 400$ agents.

We first compare the predicted dynamics for an initial activation $\alpha = 0.280$ with the simulations on a network with $n = 400$ agents, then we discuss the changes that happen if we vary the fraction α or the size n .

The function $f_{d,r}(\mu)$, with $d = 7$ and $r = 3$, is reproduced with a red line in the left plot of figure 2.4. From the plot we observe that the function has three fixed points in $[0, 1]$: two stables at the boundaries (0 and 1), and one unstable in $\mu_c = 0.256$. The same plot includes in magenta the predicted activation dynamics, iteration by iteration, for the initial condition $\mu[0] = \alpha = 0.280$. As $f_{d,r}(\mu) > \mu$ in $(\mu_c, 1)$, the activation is predicted to increase, with limit in $\mu = 1$. The plot on the right compares the prediction (the trajectory in magenta) with several simulations (in blue), for graphs with $n = 400$ agents and different wirings. The dashed horizontal line divides the interval $[0, 1]$ at μ_c : the dashed arrows are used to indicate the monotonicity interval of the LMF-Dyn function, indicating where the recursive law predicts an increasing or decreasing activation.

If we increment the size of the network, the simulated trajectories get closer to the prediction, as shown in the left plot of figure 2.5, for a network with $n = 2000$ agents. This behavior is in fact in agreement with the concentration theorem.

The right plot of figure 2.5 shows that the trajectories get closer to each other and to the prediction also when the initial activation is further away from the fixed point μ_c . The plot is made with $\alpha = 0.350$ and networks of $n = 400$ agents: we interpret this phenomenon as the effect of the larger bias toward activation.

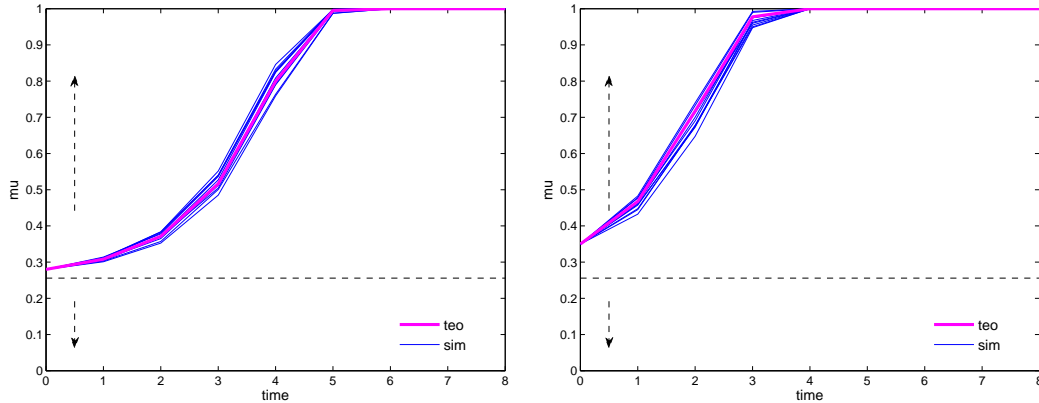


Figure 2.5: Simulations of a directed regular homogenous network with $d = 7$ and threshold $r = 3$. Left: $n = 2000$, $\alpha = 0.280$. Right: $n = 400$, $\alpha = 0.350$.

In the examples presented so far, all the simulated networks reached the complete activation. We can reasonably expect to obtain a symmetric behavior, with complete deactivation, in processes started with initial activations α a bit smaller than the fixed point μ_c .

If the initial activation α is very close to the fixed point $\mu_c = 0.256$, the network behavior is more difficult to predict, in particular for small networks. The simulations of figure 2.6 are done with $\alpha = 0.257$, slightly above the fixed point. The LMF-Dyn clearly predicts the full activation of the network, irrespective of its size n . The simulations are done for networks with size $n = 400$ (left plot) and $n = 40000$ (right plot). Some of the simulations with $n = 400$ reached complete activation, while a few get to the complete inactivation. On the larger network, apart from some outliers, the simulations got closer to the LMF-Dyn prediction. This behavior is again in agreement with the concentration result.

To help the reader visualize of the concentration, we draw a sort of “empirical LMF-Dyn” from the two sets of simulations of figure 2.6. From each simulation, we compute the points $(m[t], m[t + 1])$ and plot them on the cartesian plot in the left of figure 2.7, together with the theoretical LMF-Dyn function. It is possible to notice that the points corresponding to the simulations of networks $n = 40000$ are closer to the theoretical LMF-Dyn equation than those with $n = 400$. This facts is crucial in particular in the area closer to the critical value μ_c , area that is shown on the right side of figure 2.7.

We conclude the section by explicitly computing the quantities involved in the concentration theorem, for the network of the examples above. Since we have used regular networks, $d_{max} = k_{max} = d$. The expression (2.8), valid for

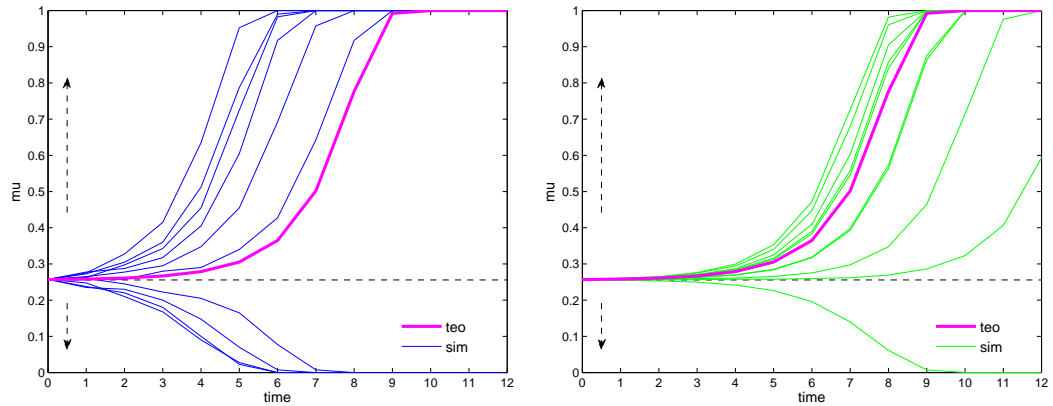


Figure 2.6: Simulations on a directed regular homogenous network with $d = 7$, threshold $r = 3$ and initial activation $\alpha = 0.257$. The unstable fixed point of $f_{d,r}(\mu)$ is in $\mu_c = 0.256$. Left: $n = 400$ agents. Right: $n = 40000$ agents.

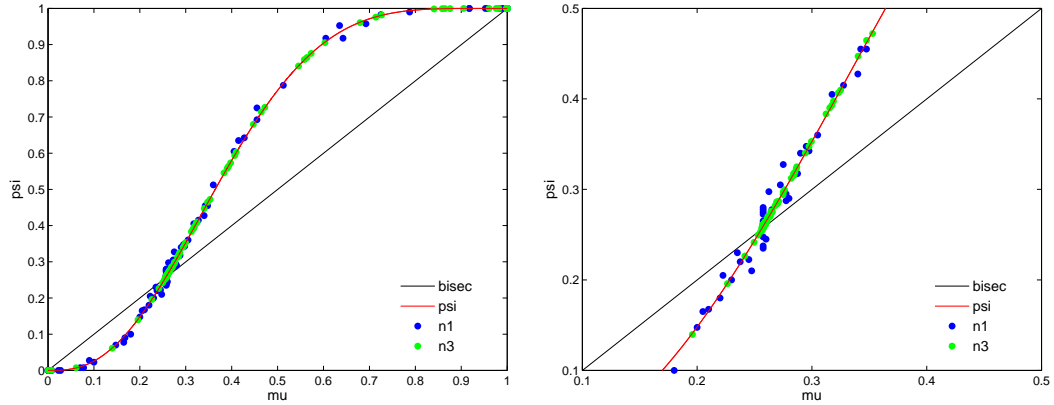


Figure 2.7: The “empirical LMF-Dyn” functions obtained from the simulations in figure 2.6, about a directed regular homogenous network with $d = 7$, threshold $r = 3$ and initial activation $\alpha = 0.257$. The blue points correspond to network of size $n = 400$, the green points to network of size $n = 40000$. Left: plot of $[0, 1] \times [0, 1]$. Right: zoom of the area closer to $\mu_c = 0.256$.

any $\epsilon > 0$ and $n > \frac{2\gamma}{\epsilon}$ becomes

$$\mathbb{P}(|m[t] - \mu[t]| > \epsilon) \leq 2e^{-\epsilon^2 n \delta},$$

with $\gamma = d^{2t+1}$ and $\delta = (16d^{2t-1})^{-1}$. Let suppose $\epsilon = 0.1$. With $d = 7$ and a network size of $n = 40000$, the hypothesis are satisfied if the time t is

$$t < \frac{\log(n\epsilon/2)}{2 \log d} - \frac{1}{2},$$

which gives $t < 1.453$. We can use the result for the first step only. We obtain:

$$\mathbb{P}(|m[1] - \mu[1]| > 0.1) \leq 2e^{-0.1^2 \times 40000 / (16 \times 7)} = 0.056.$$

The previous computation would suggest a very limited applicability of the concentration theorem. This is not surprising and does not contrasts with the concentration observed in the examples above. It should be remembered that the proof of the theorem is based on a worst case analysis, both in the Azuma inequality and in the approximation of the convergence part. Even for a small network size, the simulated trajectories are in good agreement with the predictions, for all the time interval of interest (that necessary to converge, for example, to the complete activation).

Chapter 3

Analysis of the Local Mean-Field dynamics

In the Linear Threshold Model (LTM) the agents of a network are endowed with binary states, which they update following the deterministic local rule (1.1). States may switch at any time and to determine their new states, the agents compare their personal activation threshold with the sum of the states of their neighbors. In the previous Chapter we analyzed the LTM on a large random network, the directed version of the configuration model. The goal was to estimate the fraction of agents active after each update round and indeed we found an accurate Local Mean Field (LMF) approximation. If the activation thresholds are constant, i.e. the threshold distributions are time invariant, the discrete-time LMF dynamical system is a non-linear autonomous system with scalar state. In this Chapter we analyze such system, studying its trajectories by characterizing its fixed points.

The Permanent Activation Linear Threshold Model (PALTM), with local rule (1.2), is the variant of the LTM with the state 1 being permanent. Thanks to the Equivalence Condition of Proposition 1.1, if the activation thresholds are constant the local mean field approximation can be easily extended to describe PALTM processes. The Chapter describes such extension.

The structure of the Chapter is the following. In Section 3.1 we introduce a new notation, recall the recursive equations of LMF dynamical system and discuss some properties of the scalar state function. In Section 3.2 we explain how to adapt the LMF approach to the Permanent Activation processes. In Section 3.3 we restrict the analysis to directed regular networks, assuming that the agents all have the same threshold, or at most two different thresholds. We observe a *selective activation* phenomenon: under certain conditions the activation process segregates among the agents with the smaller threshold. We provide several examples of the LMF dynamics, for both the LTM and the

PALTM, and compare it with numerical simulations. In the last Section we shortly extend the analysis to non-regular networks, and discuss the PALTM process with vanishing initial activation.

3.1 Basic properties of the LMF dynamical system

The Local Mean Field approach consists in a scalar dynamical system. In this Section we recall this system and discuss a few basic properties that descend from its state and output functions.

We remind that in the previous Chapter we denoted the joint *out,in*-degree distribution with $p_{d,k}$, whereas for the (now constant) conditional threshold distributions we used $p_{r|d,k}$. We also introduced the family of non-linear functions $f_{d,r} : [0, 1] \rightarrow [0, 1]$ whose expression is $f_{d,r}(x) = \sum_{i=r}^d \binom{d}{i} x^i (1-x)^{d-i}$.

Here we introduce the functions ψ and $\phi : [0, 1] \rightarrow [0, 1]$, that are two different *convex combinations*¹ of the functions $f_{d,r}(x)$:

$$\psi(x) = \sum_{d,k,r} \frac{k p_{d,k}}{\bar{k}} p_{r|d,k} f_{d,r}(x) \tag{3.1}$$

$$\phi(x) = \sum_{d,k,r} p_{d,k} p_{r|d,k} f_{d,r}(x). \tag{3.2}$$

Whenever necessary we shall consider the summations to run just over those d , k , and r for whom $p_{d,k} p_{r|d,k} > 0$.

In this Chapter and in Chapter 4 we will use a new notation for the sequences of the LFM dynamical system, more standard in the study of dynamical systems. The LMF approach allows to compute the sequences $\mu[t]$ and $\nu[t]$, introduced in Section 2.3 and defined by equation (2.2) and (2.3) respectively, that represent high probability predictions of the behavior of the fraction of active links and of active nodes respectively. From now on, the sequence $\mu[t]$ will be named as the *state sequence* $x[t]$ of the LMF dynamical system, whereas the *output sequence* $y[t]$ will refer to $\nu[t]$.

The recursive equations (2.4) and (2.5) of the Local Mean Field dynamical system have been derived for the Linear Threshold Model on the directed configuration model random graph. Using the definitions above and with the new

¹ Let \mathbf{v} be a vector in \mathbb{R}^N and consider the coefficients c_i , with $i \in \{1, \dots, N\}$ such that $c_i \geq 0$ and $\sum_{i=1}^N c_i = 1$. The quantity $\sum_{i=1}^N c_i v_i$ is said to be a *convex combination* of the elements of \mathbf{v} .

notation, the LMF dynamical system is

$$\begin{cases} x[t+1] = \psi(x[t]) \\ y[t+1] = \phi(x[t]) \end{cases} \quad (3.3)$$

with a given initial condition $x[0] = x_0$ and $y[0] = y_0$. Since the threshold distributions are constant, the above system is deterministic and autonomous: given the initial conditions its trajectory is determined.

The main properties of the LMF system descend from the properties of state function $\psi(x)$. The following lemmas, whose proofs are omitted, collect these properties. We call *fixed point* of a real function $g(x)$ any point x such that $g(x) = x$.

Lemma 3.1. *The function $\psi(x)$, defined in (3.1), is smooth and monotonically non-decreasing. The minimum is $\psi(0) = \sum_{d,k} \frac{k p_{d,k}}{k} p_{0|d,k}$, the maximum is $\psi(1) = 1 - \sum_{d,k} \frac{k p_{d,k}}{k} p_{d+1|d,k}$ and there exist at least one fixed point $x^* \in [0, 1]$. If there exists a triple $d, k > 0$ and $r \neq \{0, d+1\}$ such that $p_{d,k} p_{r|d,k} > 0$, then $\psi(x)$ is strictly increasing.*

Let the set P_ψ contain the fixed points of $\psi(x)$

$$P_\psi = \{x \in [0, 1] : x = \psi(x)\}$$

and assume it contains only isolated points. This non-empty set defines a partition² on the interval $[0, 1]$ and, since $\psi(x)$ is monotone, in each sub-interval of the partition the state sequence $x[t]$ is monotone.

Corollary 3.2. *Assume the set P_ψ contains isolated points. Consider the augmented set $\tilde{P}_\psi = P_\psi \cup \{0, 1\}$. The elements of \tilde{P}_ψ define a partition on the interval $[0, 1]$ and, under the function $\psi(x)$, any sub-interval maps into itself. Let $[x_i, x_{i+1}]$ be a sub-interval of the partition, then $\forall x \in (x_i, x_{i+1})$, either $\psi(x) > x$ or $\psi(x) < x$.*

Note that previous Corollary excludes limit cycles in the state dynamic. Since the sequence $x[t]$ admits a limit because is monotone and bounded, and the limit must be a fixed points.

Each fixed point x^* in P_ψ can be stable, unstable or half stable. Suppose $x^* \in (0, 1)$ and consider the two sub-intervals of the partition adjacent to x^* . The fixed point x^* is said to be *stable* if trajectories are converging to x^* from both sub-intervals, it is *unstable* if trajectories move away from both sides.

² A partition Q of an interval $[a, b]$ on the real line is a finite sequence of the form: $a = x_0 < x_1 < x_2 < \dots < x_n = b$. Every interval of the form $[x_i, x_{i+1}]$ is referred to as a sub-interval.

Otherwise it is a hybrid called *half-stable* [39]: sequences started on one side of it converge to it whereas on the other side move away. Fixed points possibly in 0 or 1 are either stable or unstable.

A simple sufficient condition for the stability of fixed points involves the first derivative of $\psi(x)$.

Lemma 3.3. *Let x^* be a fixed point of the state function ψ . If $\psi'(x^*) < 1$, then x^* is stable. If $\psi'(x^*) > 1$, then x^* is unstable.*

The monotonicity of the state function $\psi(x)$ of the autonomous system implies that the fraction of active edges evolves as a monotone sequence, from the initial condition x_0 until the first reachable fixed point. The output function $\phi(x)$ has the same properties described in Lemma 3.1 for $\psi(x)$. In particular $\phi(x)$ is non-decreasing, thus $y[t]$ is also a monotone sequence.

3.2 The extension of the LMF approach to the PALTM

The Local Mean Field (LMF) approach has been developed on a directed random network to approximate the behavior of the Linear Threshold Model (LTM) with local update rule (1.1). In this Section we extend the LFM approach to the Permanent Activation Linear Threshold Model with local rule (1.2), using Equivalence Condition in Proposition 1.1.

We stress that, with the exceptions of possible stubborn agents, the “active” state is not permanent in the LTM. Instead, the PALTM is the variant where the “active” state is permanent, thus the states of nodes and edges are non-decreasing sequences. Hence, in the PALTM quantities like the fraction of active agents or edges are non-decreasing sequences.

In its present form, the dynamical system (3.3) with state function $\psi(x)$ and output function $\phi(x)$ (equations (3.1) and (3.2) respectively) cannot be the LMF dynamical system of the PALTM. There is no guarantee for the sequences $x[t]$ and $y[t]$ to be non-decreasing, since there are x for which $\psi(x) < x$. Moreover, in the LTM the fraction of active nodes and edges evolves as a balance between agents turning active and agents turning not-active, and $\psi(x)$ represents this balance. In the PALTM no active agents can turn non-active and thus there is no balance. Therefore, possible LMF functions for the PALTM must take into account that deactivation is forbidden.

If the thresholds are constant the Equivalence Condition in Proposition 1.1 suggests a recipe to adapt the LMF dynamical system for the PALTM. The proposition guarantees that, if the thresholds are all constant and the initially

active agents are stubborn, the LTM produces a process equivalent to that obtained for the PALTM. We will use a little circle on top of the quantities adapted to represent a PALTM process with the LTM formalism.

Let assume that the degree sequences of the network are \mathbf{d} and \mathbf{k} , the threshold sequence is \mathbf{r} and the initial activation profile is $\mathbf{x}[0]$. We obtain a process equivalent to the PALTM if, in the same network, we first enforce stubbornness setting to zero the threshold of the agents with $x_v[0] = 0$ (i.e. regardless of the original value, if $x_v[0] = 0$ we set $\mathring{r}_v = 0$) and then we run the LTM. The adapted threshold sequence $\mathring{\mathbf{r}}$ is zero for the nodes initially active while it coincide with the original threshold sequence \mathbf{r} for all the other nodes:

$$\mathring{r}_v = (1 - x_v[0])r_v \quad \forall v.$$

The degree sequences and the initial activation profile remains the same, i.e. $\mathring{\mathbf{d}} = \mathbf{d}$, $\mathbf{k} = \mathbf{k}$ and $\mathring{\mathbf{x}}[0] = \mathbf{x}[0]$.

Regarding the LMF approximation, we shall adapt the degree and threshold statistics accordingly. Given the initial condition in the original network, we introduce the fractions

$$\alpha_{d,k,r} = \frac{|\{v \in V : d_v = d, k_v = k, r_v = r \text{ and } x_v[0] = 1\}|}{|\{v \in V : d_v = d, k_v = k, r_v = r\}|}.$$

The adapted threshold statistics, for every d and k , are

$$\begin{aligned} \mathring{p}_{0|d,k} &= p_{0|d,k} + \sum_{r>0} \alpha_{d,k,r} p_{r|d,k} \\ \mathring{p}_{r|d,k} &= (1 - \alpha_{d,k,r}) p_{r|d,k} \quad \text{if } r > 0. \end{aligned}$$

The degree statistic remains unchanged, i.e. $\mathring{p}_{d,k} = p_{d,k}$.

The adapted threshold distributions $\mathring{p}_{r|d,k}$ need to substitute $p_{r|d,k}$ in the expressions (3.1) and (3.2) of the state function $\psi(x)$ and output function $\phi(x)$ of the LMF dynamical system (3.3). Using $f_{d,0}(x) = 1$, we obtain the new functions

$$\begin{aligned} \mathring{\psi}(x) &= \sum_{d,k,r} \frac{k \mathring{p}_{d,k}}{\bar{k}} \mathring{p}_{r|d,k} f_{d,r}(x) \\ &= \sum_{d,k} \frac{k p_{d,k}}{\bar{k}} \left[\left(p_{0|d,k} + \sum_{r>0} \alpha_{d,k,r} p_{r|d,k} \right) + \sum_{r>0} (1 - \alpha_{d,k,r}) p_{r|d,k} f_{d,r}(x) \right] \end{aligned}$$

and

$$\begin{aligned} \mathring{\phi}(x) &= \sum_{d,k,r} \mathring{p}_{d,k} \mathring{p}_{r|d,k} f_{d,r}(x) \\ &= \sum_{d,k} p_{d,k} \left[\left(p_{0|d,k} + \sum_{r>0} \alpha_{d,k,r} p_{r|d,k} \right) + \sum_{r>0} (1 - \alpha_{d,k,r}) p_{r|d,k} f_{d,r}(x) \right]. \end{aligned}$$

The Local Mean Field dynamical system for the Permanent Adoption Linear Threshold Model with constant thresholds is

$$\begin{cases} x[t+1] = \overset{\circ}{\psi}(x[t]) \\ y[t+1] = \overset{\circ}{\phi}(x[t]). \end{cases}$$

The initial conditions of the system remain unchanged, i.e. $x[0] = x_0$ and $y[0] = y_0$. We additionally observe that x_0 and y_0 can be expressed with the fractions $\alpha_{d,k,r}$:

$$\begin{aligned} x_0 &= \sum_{d,k,r} \frac{kp_{d,k}}{\bar{k}} p_{r|d,k} \alpha_{d,k,r} \\ y_0 &= \sum_{d,k,r} p_{d,k} p_{r|d,k} \alpha_{d,k,r}. \end{aligned}$$

The expressions of $\overset{\circ}{\psi}(x)$, $\overset{\circ}{\phi}(x)$, x_0 and y_0 simplify if we assume $\alpha_{d,k,r} = \alpha$ for every d , k and r , i.e. if the initially active agents have been selected independently from their characteristics. In this special case we have

$$\begin{aligned} \overset{\circ}{\psi}(x) &= \alpha + (1 - \alpha) \psi(x) \\ \overset{\circ}{\phi}(x) &= \alpha + (1 - \alpha) \phi(x) \\ x_0 &= \alpha \\ y_0 &= \alpha. \end{aligned}$$

The functions $\overset{\circ}{\psi}(x)$ and $\overset{\circ}{\phi}(x)$ provide the correct LMF approximation for the PALTM on the directed configuration model with constant thresholds. They are the LMF functions of a modified LTM process equivalent to the PALTM process of interest. The state sequence $x[t]$ is always increasing. To see this, notice that $\min_x \overset{\circ}{\psi}(x) = \overset{\circ}{\psi}(0) \geq \alpha$ and observe that $\overset{\circ}{\psi}(x)$ is increasing. Hence, the initial condition x_0 always fall in the first sub-interval of the partition that $\overset{\circ}{\psi}$ defines on $[0, 1]$. Finally, in the first sub-interval the dynamic is increasing: for example $\overset{\circ}{\psi}(\alpha) \geq \alpha$.

We provide a simple example to stress the differences between $\psi(x)$ and $\overset{\circ}{\psi}(x)$, i.e. between a LTM process and the corresponding PALTM process. The example is similar to those in Section 2.5: we consider a directed random network where all the agents have the same out-degree $d = 7$, in-degree $k = 7$ and threshold $r = 3$, and we assume the fraction of initially active node is α . The two LMF state function to be compared have expression

$$\begin{aligned} \psi(x) &= f_{7,3}(x) && \text{LTM} \\ \overset{\circ}{\psi}(x) &= \alpha + (1 - \alpha) f_{7,3}(x) && \text{PALTM.} \end{aligned}$$

The state function $\psi(x)$ actually represents a LTM process on a modified network where fraction α of the nodes has threshold $r = 0$ (instead of $r = 3$). In fact we can write $\psi = \alpha f_{7,0}(x) + (1 - \alpha) f_{7,3}(x)$ and remember that $f_{7,0}(x) = 1$. In the following figures we also compare the state sequences $x[t]$, initialized with $x[0] = \alpha$ for both models.

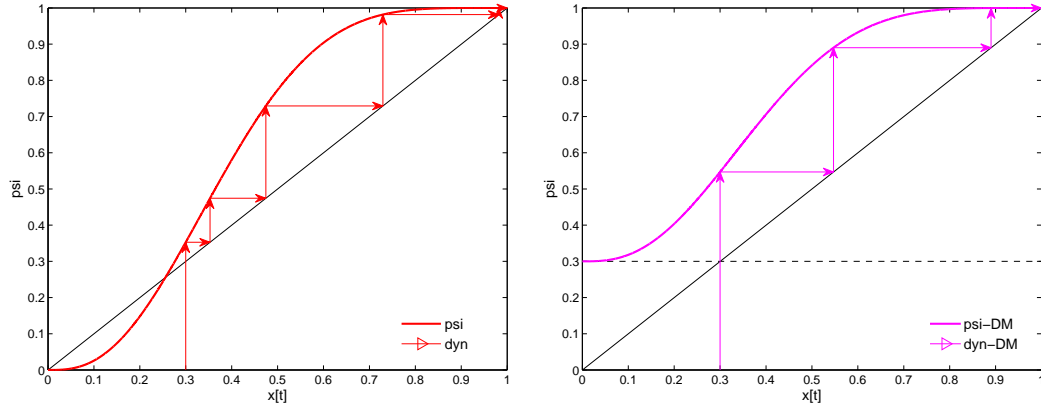


Figure 3.1: A simple comparison of the LMF state functions and sequences for the LTM and PALTM. On the left $\psi(x) = f_{7,3}(x)$ corresponding to the LTM. On the right $\psi(x) = \alpha + (1 - \alpha) f_{7,3}(x)$ corresponding to the PALTM. The initial activation is $\alpha = 0.300$.

In Figure 3.1 we compare the LFM state functions of the two models using $\alpha = 0.300$. The left plot regards the LTM and contains the state function $\psi(x)$ in solid red and the corresponding state sequence whose first elements are represented by the arrows. The sequence is increasing and converges to 1. The right plot regards the PALTM. The dashed horizontal line at height α represents the initially active agents, whose thresholds has been set to zero according to the procedure described above. The right plot contains the state function $\psi(x)$ in solid magenta and the first elements of the corresponding state sequence represented with the arrows and converging to 1. The initial activation is sufficient to activate all the network, for both the LTM and the PALTM process. However, the state sequences $x[t]$ have different elements.

The comparison for $\alpha = 0.220$ is done in Figure 3.2 and shows a deeper difference between the two activation models. The left plot, corresponding to the LTM, predicts that the network completely deactivates. The right plot instead shows that the network completely activates with the PALTM.

The LTM and the PALTM produce two different activation processes. Thanks to the Equivalence Condition, it is possible to obtain a LMF approximation of the PALTM process, using the LMF of an adapted LTM.

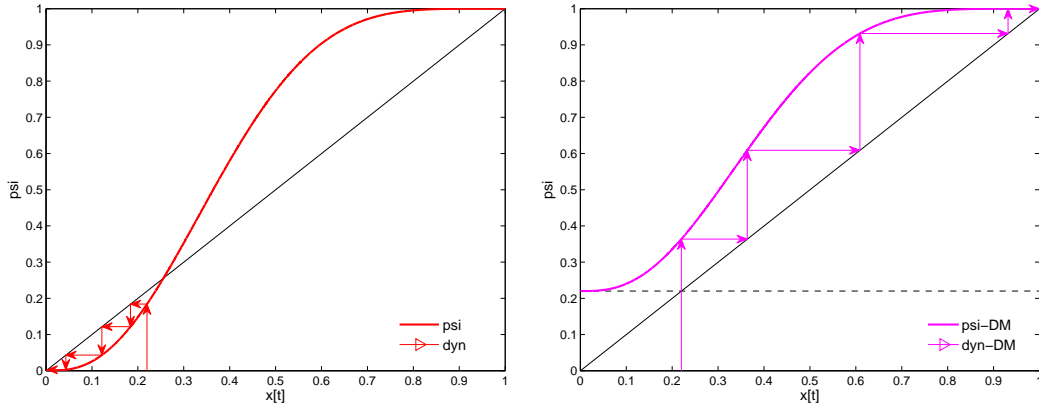


Figure 3.2: A second comparison of the LMF state functions and sequences for the LTM and PALTM. On the left $\psi(x) = f_{7,3}(x)$ corresponding to the LTM. On the right $\psi^\circ(x) = \alpha + (1 - \alpha) f_{7,3}(x)$ corresponding to the PALTM. The initial activation is $\alpha = 0.220$.

3.3 Regular networks

In this Section we analyze the Local Mean Field (LMF) dynamical system in directed regular networks, to estimate the evolution of the fraction of active agents. All the agents of the graphs G that will be considered have the same *out-degree* and *in-degree*, of value d_G and the degree distribution becomes trivial. The activation thresholds are allowed to assume values in $\{0, 1, \dots, d_G + 1\}$: we denote with p_r the corresponding threshold distribution. Additionally we assume that the initially active agents are chosen independently from the thresholds values and that the fraction of initially active agents is $\alpha \in [0, 1]$. We will also refer to α as the initial *seed* of the activation process.

With the above assumptions, the LMF state function (3.1) of the Linear Threshold Model (LTM) simplifies to

$$\psi(x) = \sum_r p_r f_{d,r}(x)$$

whereas the state function of the Permanent Adoption Linear Threshold Model (PALTM) becomes

$$\psi^\circ(x) = \alpha + (1 - \alpha) \psi(x) = \alpha + (1 - \alpha) \sum_r p_r f_{d,r}(x).$$

In a directed regular network the expressions of the output functions $\phi(x)$ and $\phi^\circ(x)$ coincide with the expressions above for $\psi(x)$ and $\psi^\circ(x)$ respectively.

Therefore the state sequence $x[t]$ and output sequence $y[t]$ of the LMF dynamical systems coincide, i.e.

$$y[t] \equiv x[t] \quad \forall t \geq 0.$$

The state sequence $x[t]$ is sufficient to analyze the LFM approximation of the LTM and PALTM processes. We remind that the sequence starts with $x[0] = \alpha$ for both models, then for the LTM the sequence follows the recursion $x[t+1] = \psi(x[t])$ while for the PALTM follows the recursion $x[t+1] = \psi^\circ(x[t])$.

Together with the evolution of the sequence $x[t]$, we are interested to estimate its limit value. We call *asymptotic* (or final) *activation* the limit

$$\beta := \lim_{t \rightarrow \infty} x[t].$$

The limit value depends on the seed α and we shall identify the function $\beta(\alpha)$.

We will consider a few example to describe the behavior of the predicted dynamic $x[t]$ and the asymptotic activation $\beta(\alpha)$. We will compare the LMF predictions with numerical simulations of the LTM and PALTM in random networks. To this end, we remind that $a[t]$ denotes the fraction of active agents in the network and with the standing assumptions $a[0] = \alpha$.

The remaining of the Section is organized as follows. First we assume that all the agents have the same threshold value r , to study the simplified homogenous case. Then we let the agents have one of two different thresholds value r_1 and r_2 . Finally we describe some heuristic for the general case where all the threshold values $\{0, 1, \dots, d_G + 1\}$ are available.

3.3.1 Networks with one threshold value

We start with the homogeneous case: all the agents in the graph G not only have the same *out*-degree and *in*-degree d_G , they also share the same activation threshold r . First we consider the LTM, then the PALTM.

The LMF state function for the LTM is $\psi(x) = f_{d_G, r}(x)$, with $f_{d, r}(x) = \sum_{i=r}^d \binom{d}{i} x^i (1-x)^{d-i}$. Therefore, the sequence $x[t]$ is given by the recursion

$$\begin{cases} x[t+1] & = f_{d_G, r}(x[t]) \\ x[0] & = \alpha. \end{cases}$$

The fixed points of the function $f_{d_G, r}(x)$ are described in the following Lemma.

Lemma 3.4. *The function $f_{d_G,r}(x)$, in the interval $[0, 1]$, has the following fixed points:*

$$\begin{cases} 1 & \text{if } r = 0 \\ \underline{0}, 1 & \text{if } r = 1 \\ 0, \underline{a}, 1 & \text{if } r \in \{2, 3, \dots, d_G - 1\} \\ 0, \underline{1} & \text{if } r = d_G \\ 0 & \text{if } r = d_G + 1. \end{cases}$$

The underlined fixed points are unstable, the remaining are stable.

The Lemma collects a few properties of the functions $f_{d,r}(x)$ described in the Appendix B, so we omit the proof.

Given the discussion of Section 3.1, the Lemma above and the Lemma 3.1 completely characterize the possible outcomes of the LTM process in a homogeneous network. Excluding the unstable equilibria, the LMF approximations predicts that the process ends with either the complete activation or the complete deactivation of the network, with $\beta \in \{0, 1\}$. If $r \in \{2, 3, \dots, d_G - 1\}$ there is a threshold condition: the limit value is 0 or 1 depending on whether the seed α is smaller or larger than the value \underline{a} .

Regarding the PALTM process, the state function of the LMF approximation is $\dot{\psi}(x) = \alpha + (1 - \alpha) f_{d_G,r}(x)$. The sequence $x[t]$ is given by the recursion

$$\begin{cases} x[t + 1] & = \alpha + (1 - \alpha) f_{d_G,r}(x[t]) \\ x[0] & = \alpha. \end{cases}$$

The asymptotic activation $\beta(\alpha)$ corresponds to the smallest fixed point of the recursion above. The most interesting cases arise for $r \in \{2, \dots, d_G - 1\}$ since the function $\beta(\alpha)$ is discontinuous in $\alpha_c \in (0, 1)$. The *critical seed* α_c is the value of α such that the function $\dot{\psi}(x)$ is tangent to the 45° line in its smallest fixed point. Therefore we obtain the critical seed from the system

$$\begin{cases} \dot{\psi}(x) & = x \\ \dot{\psi}'(x) & = 1 \end{cases}$$

which we rewrite as follows to highlight the role of α

$$\begin{cases} \alpha + (1 - \alpha) \psi(x) & = x \\ (1 - \alpha) \psi'(x) & = 1. \end{cases}$$

The critical seed distinguishes two regimes: for seeds $\alpha \leq \alpha_c$ the activation spreads to a limited fraction of the network, of order α ; for seeds $\alpha > \alpha_c$ the asymptotic activation is complete, with $\beta = 1$.

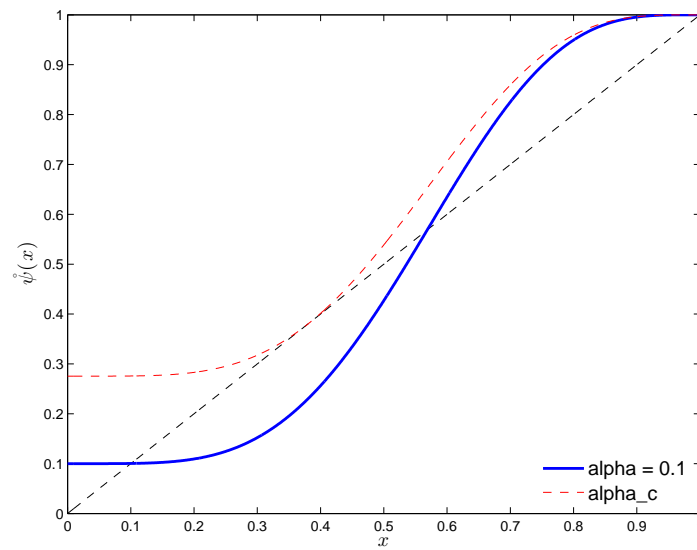


Figure 3.3: The function $\hat{\psi}(x) = \alpha + (1 - \alpha) f_{d_G, r}(x)$ with $d_G = 8$ and $r = 5$ for two values of α . In solid blue $\alpha = 0.100$; in dashed red α assumes the critical value $\alpha_c \approx 0.267$.

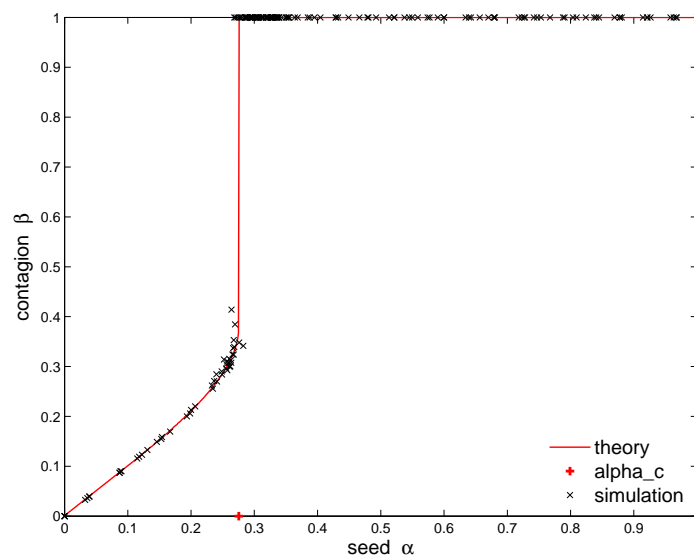


Figure 3.4: The function $\beta(\alpha)$ (red solid line) is compared with the simulations in a network with $n = 2000$ agents (black crosses). It is possible to recognize the phase transition for $\alpha_c \approx 0.276$.

In the PALTM example we choose $d_G = 8$ and $r = 5$ to which correspond $\alpha_c \approx 0.276$. Figure 3.3 represents the function $\psi(x) = \alpha + (1 - \alpha) f_{d_G, r}(x)$ for two values of α . The plot in solid blue is for $\alpha = 0.100$, smaller than the critical value. The corresponding smallest fixed point is approximately 0.100. The plot in dashed red is for $\alpha = \alpha_c$: observe that the plot is tangent to the bisector line. If α gets a little larger than α_c the smallest fixed point jumps to 1. This is the reason why the function $\beta(\alpha)$ is discontinuous in α_c .

Figure 3.4 contains the function $\beta(\alpha)$ of this example in the solid red. The black crosses in the Figure represent the final activation of the PALTM process on a random network with $n = 2000$ nodes. The black crosses represent the results of the simulations on a network with $n = 2000$ nodes. Each cross correspond to a simulations for $a[0] = \alpha$ and has height equal to the settling value of $a[t]$.

In Figure 3.5 we compare the dynamic behavior of the LMF state sequence $x[t]$ (red lines) with the evolution of the active fraction of agents $a[t]$ in the simulations (blue lines). The two plots use graphs of two different sizes. The graphs size is $n = 2000$ for the top plot and $n = 20000$ for the bottom plot; the initial condition is above the critical α_c . It is possible to observe that the simulations concentrate around the prediction, and are more concentrated for the larger graph.

3.3.2 Networks with two threshold values

In this Subsection we allow the agents to choose between two possible activation thresholds, r_1 and r_2 with $r_1 < r_2$. The parameter $q \in [0, 1]$ gives the fraction of agents with threshold r_2 ; the remaining fraction $1 - q$ has threshold r_1 . We remember that the fraction of initially activated agents is chosen independently from the thresholds and is α .

In general the presence of two different activation thresholds r_1 and r_2 increments the possible outcomes of the activation processes. On one hand, if the difference $r_2 - r_1$ is small (compared to the degree d_G) or the parameter q is close to the extremes of the interval $[0, 1]$, the processes have a behavior similar to the case of a single activation threshold. On the other hand, for larger differences $r_2 - r_1$ and parameters q chosen around 0.5 (i.e. to make the fractions of agents with thresholds r_1 and r_2 comparable) the function $\beta(\alpha)$ presents two discontinuities, and in the intermediate regime between them it is possible to observe the selective activation phenomenon.

We start to analyze the process corresponding to the LTM. The state function of the LMF approximation is

$$\psi(x) = (1 - q) f_{d_G, r_1}(x) + q f_{d_G, r_2}(x) \tag{3.4}$$

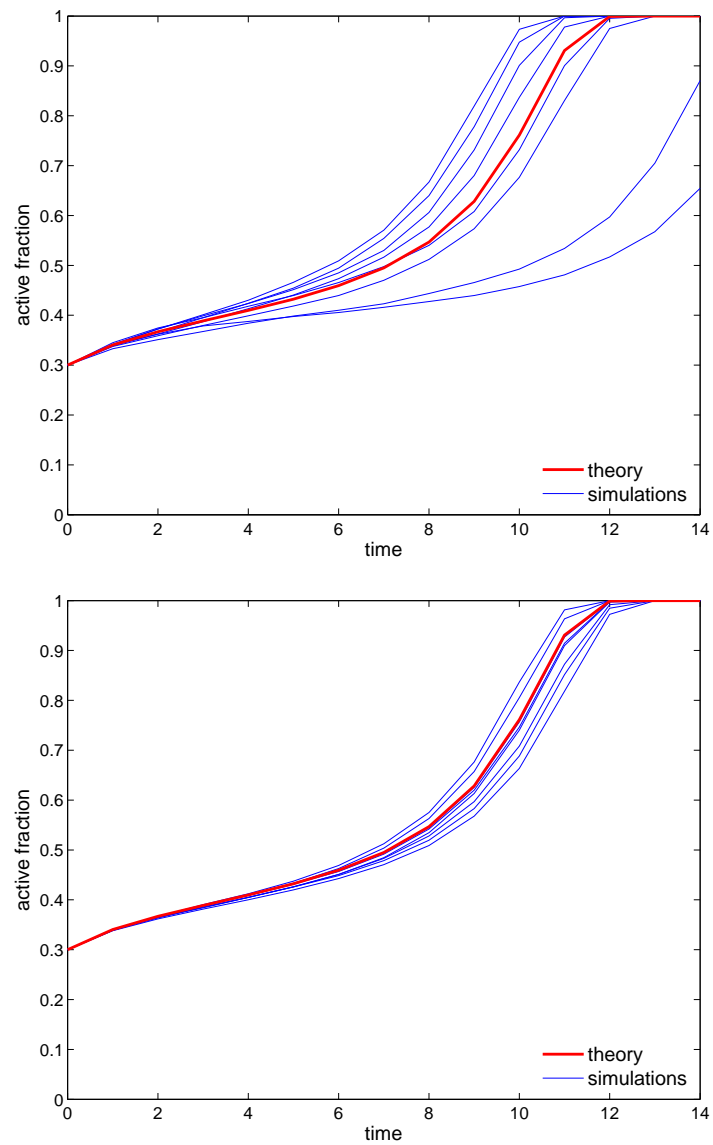


Figure 3.5: In blue, some simulations of the evolution of the PALTM process, in network $n = 2000$ agents (top) and $n = 20000$ (bottom) for an initial activation larger than the critical value. The red line represent the predicted dynamics. It is possible to notice the concentration of the simulated traces for larger graphs.

and the sequence $x[t]$ is obtained from the recursion

$$\begin{cases} x[t+1] &= \psi(x[t]) \\ x[0] &= \alpha. \end{cases}$$

The sequence $x[t]$ approximates the probability that a generic agent is active. However we expect that agents with different thresholds have different probability to be activated. To verify this, we introduce the specific output sequences $y_1[t]$ and $y_2[t]$, defined as

$$y_i[t] := \mathbb{P}(x_u[t] = 1 \mid \mathcal{N}_{u,+}^{t-1} \text{ is a tree; } r_u = r_i) \quad i \in \{1, 2\}. \quad (3.5)$$

The corresponding output functions are

$$\phi_i(x) = f_{d_G, r_i}(x) \quad i \in \{1, 2\}$$

and give the two output sequences as $y_i[t+1] = \phi_i(x[t])$.

The first example we consider is used to show the *selective activation* for the LTM. We set $d_G = 18$, $r_1 = 3$, $r_2 = 13$ and $q = 0.600$. The corresponding function $\psi(x)$ of equation (3.4) is drawn in red in Figure 3.6. It has five fixed points, that graphically coincide with the intersection between $\psi(x)$ and the 45° line: they are 0 , x_a , x_b , x_c and 1 , with x_a , x_b , $x_c \in (0, 1)$. To analyze the stability of those fixed point we use the graphical interpretation of the sufficient condition in Lemma 3.3. At any intersection between $\psi(x)$ and the 45° line, if the slope of $\psi(x)$ is less than that of the 45° line, the fixed point is stable. Else, if the slope of $\psi(x)$ is larger, it is unstable. The three fixed points 0 , $x_b \approx 0.400$ and 1 are stable, whereas $x_a \approx 0.090$ and $x_c \approx 0.680$ are unstable.

The final fraction of activated nodes, i.e. the function $\beta(\alpha)$, can be obtained from the fixed points and their stability. According to the Corollary 3.2, the fixed points of $\psi(x)$ define a partition on the interval $[0, 1]$. If α is an internal point of one subinterval of the partition, the sequence $x[t]$ will increase (or decrease) monotonically while remaining in that subinterval. Eventually $x[t]$ converges to the stable fixed point at the boundary of the subinterval. Informally, we call *basin of attraction* of a stable fixed point the set of initial conditions α that lead the sequence $x[t]$ to converge to that fixed point. From Figure 3.6 we recognize the basins of attraction of the stable fixed points: $[0, x_a)$ for 0 , (x_a, x_c) for x_b and $(x_c, 1]$ for 1 . For example, if the LTM process is started with a seed $\alpha \in (x_a, x_c)$, the LMF dynamics predicts a final activation $\beta = x_b$. Hence, the function $\beta(\alpha)$ is discontinuous: for $\alpha \notin \{x_a, x_c\}$ ³ it is

$$\beta(\alpha) = \begin{cases} 0 & \text{if } \alpha \in [0, x_a) \\ x_b & \text{if } \alpha \in (x_a, x_c) \\ 1 & \text{if } \alpha \in (x_c, 1]. \end{cases}$$

³We would have $\beta(x_a) = x_a$ and $\beta(x_c) = x_c$, but this are unstable solutions and we exclude them.

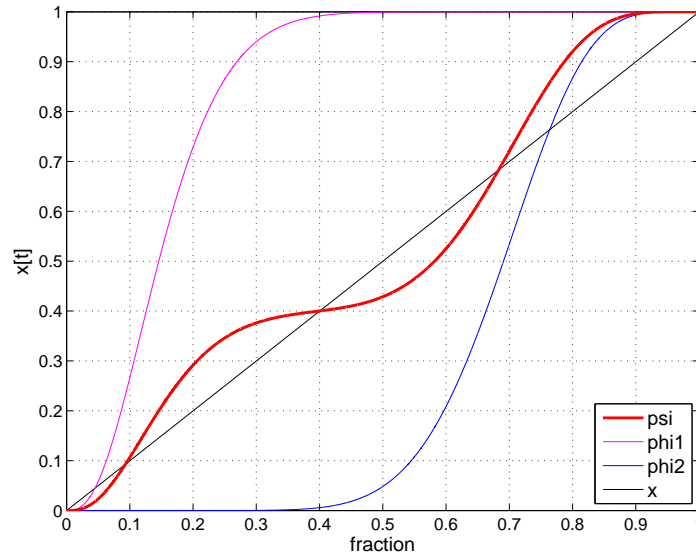


Figure 3.6: The state function $\psi(x) = (1 - q) f_{d_G, r_1}(x) + q f_{d_G, r_2}(x)$, for $d_G = 18$, $r_1 = 3$, $r_2 = 13$ and $q = 0.600$, is plotted in red. The specific output functions $\phi_1(x) = f_{d_G, r_1}(x)$ and $\phi_2(x) = f_{d_G, r_2}(x)$ are plotted in magenta and blue, respectively.

The Figure 3.6 also contains the functions ϕ_1 and ϕ_2 , in magenta and blue respectively. Given x , these two functions provide the expected activation of the nodes with thresholds r_1 and r_2 respectively. We observe that $\phi_1(x_b) \approx 1$ whereas $\phi_2(x_b) \approx 0$. This means that for $x[t] \approx x_b$ nearly all the agents with smaller thresholds are active, while most of those with larger threshold are not. This is an example of the *selective activation*: x_b is one of the possible limits of $x[t]$ (i.e. one of the value attained by $\beta(\alpha)$) in correspondence of which the two different kind of agents have a rather different chance of being activated.

The presence of agents with two different thresholds is not sufficient to obtain equilibria with selective activation. From various simulations, we observed that it is necessary to have comparable fractions of the two type of agents and a difference $r_2 - r_1$ not too small compared to d_G . To show that $r_2 - r_1$ should not be too small, we set $d_G = 13$, $r_1 = 5$, $r_2 = 9$ (so $r_2 - r_1 = 4$) and $q = 0.500$. The corresponding function $\psi(x)$ is represented in blue in Figure 3.7, and has only one fixed point in $(0, 1)$, unstable (the fixed point is located exactly in 0.500 for symmetry reasons in the expression of $\psi(x)$). We would have obtained a similar function if all the agents had the same threshold (for example $r = 7$, giving $\psi(x) = f_{d_G, r}(x)$): in the example the difference $r_2 - r_1 = 4$ is not enough to obtain an intermediate stable fixed point.

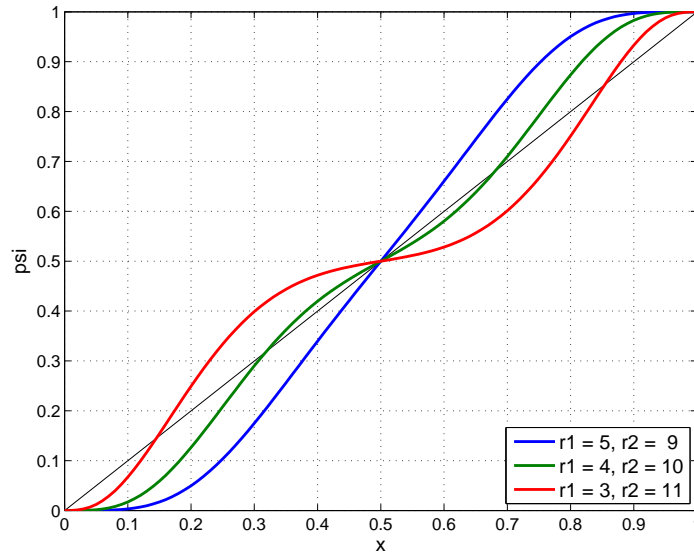


Figure 3.7: The state function $\psi(x) = (1-q) f_{d_G, r_1}(x) + q f_{d_G, r_2}(x)$ with $d_G = 13$ and $q = 0.500$ is plotted for three pairs of values of r_1 and r_2 , to show the effect of changing $r_2 - r_1$. In blue $r_1 = 5$, $r_2 = 9$; in green $r_1 = 4$, $r_2 = 10$; in red $r_1 = 3$, $r_2 = 11$.

The choice $r_1 = 4$ and $r_2 = 10$ gives a qualitatively different function $\psi(x)$, plotted in green in Figure 3.7. The difference $r_2 - r_1 = 6$ is enough to obtain three internal fixed points, with 0.500 being a stable one. The choice $r_1 = 3$ and $r_2 = 11$ ($r_2 - r_1 = 8$) give a function (drawn in red) similar to the green one except for the larger basin of attraction for the central fixed point, again in 0.500. Hence, we have two possible qualitative behaviors depending on the difference $r_2 - r_1$: for small differences we obtain functions $\psi(x)$ like the blue one, similar to those of networks where all the agents have the same threshold. For large differences, we have functions like the green and red ones that exhibit an internal stable equilibria and produce the selective activation phenomenon.

Analyzing the function $\psi(x)$ for different degrees d_G , we observed that the difference $r_2 - r_1$ shall be “large enough” compared to the degree d_G itself, to obtain the intermediate stable fixed point. Also the choice of q influences the qualitative shape of the function $\psi(x)$. A q close to zero or one makes one term in the combination (3.4) hide the other.

We move to the analysis of the PALTM, whose LMF approximation has

state function

$$\begin{aligned}\dot{\psi}(x) &= \alpha + (1-\alpha)\psi(x) \\ &= \alpha + (1-\alpha)((1-q)f_{d_G,r_1}(x) + qf_{d_G,r_2}(x)).\end{aligned}\tag{3.6}$$

The corresponding sequence $x[t]$ follows the recursion

$$\begin{cases} x[t+1] &= \dot{\psi}(x[t]) \\ x[0] &= \alpha.\end{cases}$$

The output functions for the PALTM, defined in equation (3.5), are

$$\dot{\phi}_i(x) = \alpha + (1-\alpha)f_{d_G,r_i}(x)$$

and by $y_i[t+1] = \dot{\phi}_i(x[t])$ give the activation probability of the agents with threshold r_i .

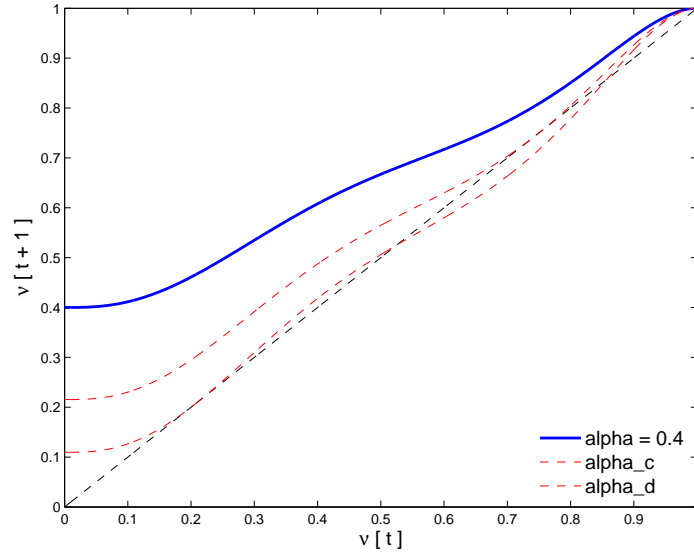


Figure 3.8: The function $\dot{\psi}(x) = \alpha + (1-\alpha)((1-q)f_{d_G,r_1}(x) + qf_{d_G,r_2}(x))$ with $d_G = 8$, $r_1 = 3$, $r_2 = 7$ and $q = 0.500$, for three values of α . In solid blue $\alpha = 0.400$; the lower dashed red curves uses $\alpha_c \approx 0.110$ while the upper dashed red curve uses $\alpha_d \approx 0.215$.

The function $\beta(\alpha)$ may exhibit one or two discontinuities (corresponding to one or two phase transitions) depending on the parameters d_G , r_1 , r_2 and q . Excluding trivial values for r_1 and r_2 , if the difference $r_2 - r_1$ is small the function $\beta(\alpha)$ has only one discontinuity only and its qualitative shape is similar to that obtained in the previous Subsection, with just one threshold.

If the difference $r_2 - r_1$ is not too small, there is a range of parameter q where the function $\beta(\alpha)$ shows two discontinuities (and correspondingly two phase transitions). Let α_c and α_d denote the position of the two discontinuities. For seeds $\alpha \leq \alpha_c$ the activation spreads to a limited fraction of the network, of order α ; for seeds $\alpha > \alpha_d$ the asymptotic activation is complete, with $\beta = 1$. For $\alpha \in (\alpha_c, \alpha_d)$ there is a new, intermediate regime. The final activation remains partial but larger than α , and nearly all the agents with the smaller threshold are activated. Hence a seed in (α_c, α_d) produces a selective final activation.

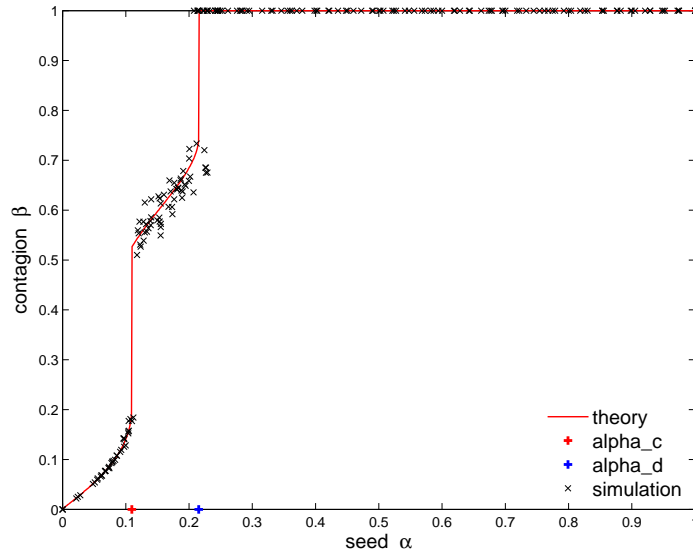


Figure 3.9: The function $\beta(\alpha)$ (red solid line) is compared with the simulations in a network with $n = 5000$ agents (black crosses). It is possible to recognize the two phase transitions for $\alpha_c \approx 0.110$ and $\alpha_d \approx 0.215$.

We present an example of what is described above, using $d_G = 8$, $r_1 = 3$, $r_2 = 7$ and $q = 0.500$. In Figure 3.8 the function $\dot{\psi}$ of equation (3.6) is drawn for three values of α . The solid blue curve is drawn with $\alpha = 0.400$, larger than α_d . Notice that its smallest fixed point is in 1, so $\beta(0.400) = 1$. Since $\dot{\psi}(x) = \alpha + (1 - \alpha)\psi(x)$, changing α shifts and rescales $\dot{\psi}(x)$. The two dashed red curves are for α equal to the critical values α_c and α_d : for this choice of α the corresponding functions are tangent to the 45° degree line in their smallest fixed point. The presence of two intervals in which the function $\psi(x)$ is convex is necessary (but not sufficient) for the the presence of two critical values for α .

Figure 3.9 shows in solid red the function $\beta(\alpha)$ obtained by the smallest fixed point of (3.6). The function is discontinuous in α_c and in α_d , marked with the red and blue plus signs. The black crosses in the Figure are the simulations

of the PALTM process on a random networks with $n = 5000$ agents. Again each cross corresponds has x -value $a[0] = \alpha$ and y -value the settling value of $x[t]$. The simulations confirm the predicted shape of the function $\beta(\alpha)$.

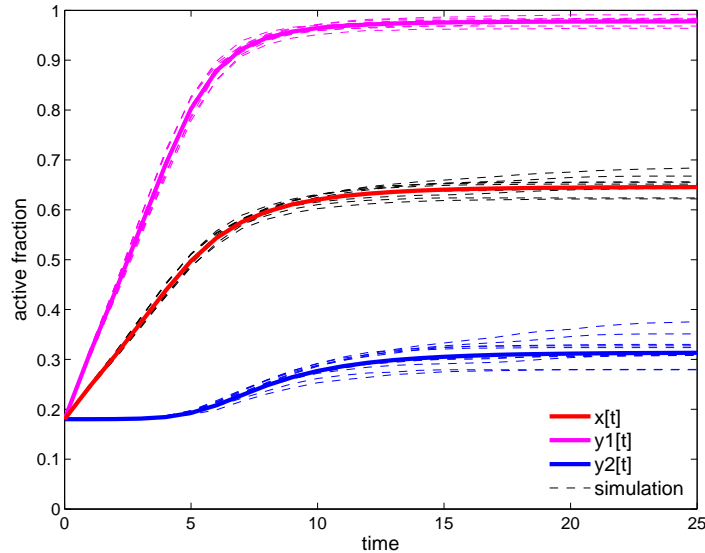


Figure 3.10: The evolution of the PALTM process for $\alpha = 0.200$ inside the interval (α_c, α_d) of the intermediate regime. The solid red line represent the predicted dynamics $x[t]$, the magenta solid line is $y_1[t]$, the blue solid line is $y_2[t]$. The dashed line are the corresponding simulated quantities.

The Figure 3.10 compares the predicted and simulated dynamic of the activation process for $\alpha = 0.200$, a value between α_c and α_d . Before we describe the simulations we introduce a few quantities to help the description. As usual let V be the set containing the agents of the network and consider the subset $V_1 = \{v \in V : r_v = r_1\}$ that contains the agent with threshold r_1 and the subset $V_2 = \{v \in V : r_v = r_2\}$. We remind that the (overall) fraction of active agents at each time step is $a[t] = |V|^{-1} \sum_{v \in V} x_v[t]$. For $i \in \{1, 2\}$ we introduce the fractions $a_i[t]$ of active agents among those with a specific threshold

$$a_i[t] = |V_i|^{-1} \sum_{v \in V_i} x_v[t].$$

Note that these quantities $a_i[t]$ correspond (and shall be compared) to the specific outputs $y_i[t]$ of the LMF dynamic.

In Figure 3.10 the dashes lines represent the simulations: the black dashed lines is $a[t]$, while the magenta and the blue dashed line are the fractions $a_1[t]$ and $a_2[t]$ respectively. The solid lines are the LMF sequences: $x[t]$ in red, $y_1[t]$

in magenta and $y_2[t]$ in blue. The LMF approach approximates well the actual simulations. Interestingly agents with different threshold activate at different time during the process. In fact, for $t = 5$ nearly the 80% of the agents with threshold r_1 are activated, while just a few agents further than those in the seed has activate among those with threshold r_2 .

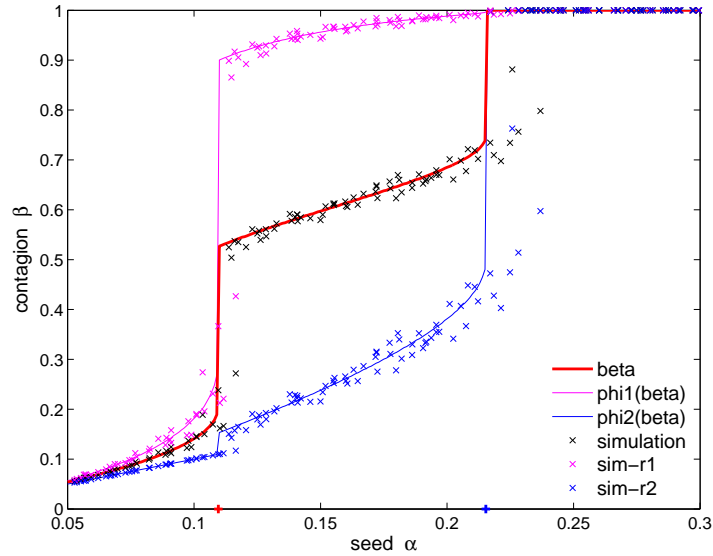


Figure 3.11: Comparison of the predicted asymptotic activation (solid red) and the results obtained from the simulations (black crosses) on a network with 5000 agents and with the initial seed in the range $[0.050, 0.300]$. In magenta and blue solid line represent the activation predicted for the agents with threshold $r_1 = 3$ and $r_2 = 7$ respectively. The magenta and blue crosses represent the corresponding sub-fractions obtained in the simulations.

The Figure 3.11 shows the asymptotic activation for seeds α about the interval (α_c, α_d) . Notice the α axis restricted to $[0.050, 0.300]$. The red solid line is the function $\beta(\alpha)$ while the black crosses correspond to the results of the simulations (the same results contained in Figure 3.9). The Figure also contains the prediction and simulation results specified for the agents with threshold r_1 or r_2 . The magenta solid line is the composed function $\phi_1(\beta(\alpha))$, that approximates the probability that an agent with threshold r_1 eventually gets activated in a process with seed α . The magenta crosses represent the sub-fraction of agents with threshold r_1 that is activated at the end of the simulation, i.e. the values to which $a_1[t]$ eventually converged in the simulations. Similarly for the agents with threshold r_2 (larger than r_1), the blue curve is the function $\phi_2(\beta(\alpha))$ and the blue crosses are the final values $a_2[t]$ obtained in the

simulations. The plot clearly shows the selective activation phenomenon. If the seed α belongs to the intermediate interval (α_c, α_d) the agents with smaller threshold are nearly all active in the final configuration of the activation process.

We stress that the presence of the intermediate regime, with selective activation, depend on the thresholds r_1, r_2 and on the parameter q . Using $q = 0.500$ but a smaller $r_2 - r_1$ (for example $r_1 = 4$ and $r_2 = 6$) we would obtain only one phase transition with behaviors similar the case where all the agents share the same threshold.

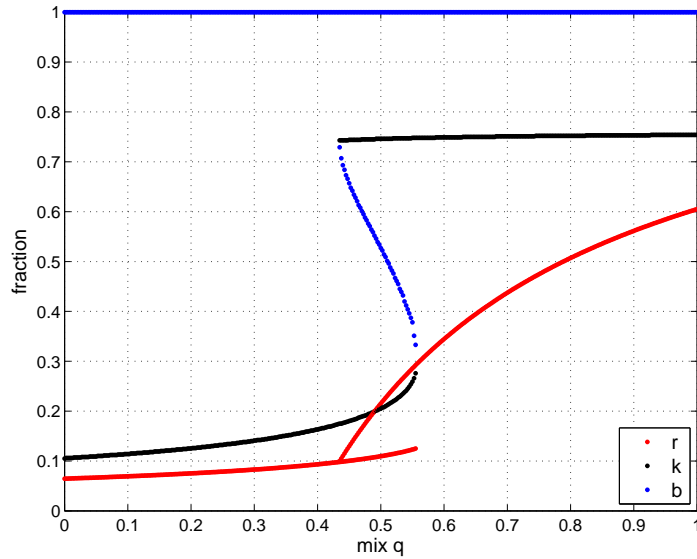


Figure 3.12: Bifurcation plot showing the dependance from the parameter q of the critical seeds $\alpha_{q,c}$ (and $\alpha_{q,d}$), in red dots. The corresponding left limits $\lim_{\alpha \rightarrow \alpha_{q,c}^-} \beta_q(\alpha)$ and $\lim_{\alpha \rightarrow \alpha_{q,d}^-} \beta_q(\alpha)$ and right limits $\lim_{\alpha \rightarrow \alpha_{q,c}^+} \beta_q(\alpha)$ and $\lim_{\alpha \rightarrow \alpha_{q,d}^+} \beta_q(\alpha)$ are represented with black and blue dots respectively.

Finally, with the last example we show that the parameter q is a bifurcation parameter, i.e. the number of discontinuities of the function $\beta(\alpha)$ (one or two) depends on q . We use again $d_G = 8$, $r_1 = 3$, $r_2 = 7$. Given $q \in [0, 1]$ we denote with $\psi_q(x)$ the corresponding LMF state function of the PALTM and with $\beta_q(\alpha)$ the asymptotic activation. We denote with $\alpha_{q,c}$ and $\alpha_{q,d}$ (where necessary) the the point of discontinuities of $\beta_q(\alpha)$. The red dots in Figure 3.12 represent the critical seeds $\alpha_{q,c}$ and (where present) $\alpha_{q,d}$ for a given parameter q . The plot shows a bifurcation in q : there are values of q , around 0.500, for which $\beta_q(\alpha)$ has two discontinuities, while for q closer to 0 or 1 there is only one discontinuity. The black dots represent the left limit $\lim_{\alpha \rightarrow \alpha_{q,c}^-} \beta_q(\alpha)$ while the blue dots correspond to the right limit $\lim_{\alpha \rightarrow \alpha_{q,c}^+} \beta_q(\alpha)$. If $\alpha_{q,d}$ is present,

black and blue dots also represent the left and right limit for $\alpha \rightarrow \alpha_{q,d}$.

The availability of two threshold values has increased the types of outcomes of the LTM and PALTM activation processes. Moreover, we observed the selective activation phenomenon for range of parameters.

3.3.3 Networks with several threshold values

The activation threshold of any agent in a regular network with degree d_G can assume any value in $\{0, 1, \dots, d_G+1\}$. The discussion of the previous Subsection generalizes trivially to networks with agents of threshold 0 and $d_G + 1$ further than those with thresholds r_1 and r_2 . Remember that those two special value enforce stubbornness and thus reduce the number of agents that may actually switch state due to the activation process.

The generalization is not trivial if there are agents with other threshold values. Assuming all the thresholds $\{0, 1, \dots, d_G+1\}$ are available to the agents in the network, the theoretical predictions show the following trends, for both the LTM and PALTM.

- If the fractions of agents with a certain threshold are approximately comparable, i.e. the statistics p_r is nearly uniform, we observed at most one discontinuity in $\beta(\alpha)$.

This fact is due to the properties of the function $f_{d,r}(x)$. Notice that $\sum_{r=1}^d f_{d,r}(x) = dx$, while $f_{d,0}(x) = 1$ and $f_{d,d+1}(x) = 0$. These properties imply that an uniform distribution of threshold among $\{0, 1, \dots, d_G + 1\}$ completely smooths out the concavities of $\psi(x)$ and $\dot{\psi}(x)$. For a nearly uniform threshold distribution, $\psi(x)$ has just one or two stable fixed point while there is just one value of α that makes $\dot{\psi}(x)$ tangent to the 45° degree line. Hence, $\beta(\alpha)$ has just one discontinuity.

- If few threshold values are present in the network and these values are sufficiently different, we could observe more phase transitions with selective activation.

The presence of more fixed point in $\psi(x)$ and more values of α such that $\dot{\psi}(x)$ is tangent to the 45° degree line is related to the convexities of $\sum_{r=1}^d p_r f_{d,r}(x)$. The convexities are more prominent if the distribution p_r is localized on few threshold values and these values are sufficiently distant one to the other compared to the degree d_G .

These observations are based on several simulations we have carried out, in networks with a sufficient number of agents: the larger the network, the closer the simulations become to the theoretical predictions.

3.4 Extensions to non-regular networks

We extend the previous analysis to non-regular networks, emphasizing the asymptotic activation for the Permanent Adoption Linear Threshold Model (PALTM). In a non-regular network the Local Mean Field (LMF) approximation requires both the state sequence $x[t]$ and the output sequence $y[t]$ because the LMF state function and output function are not generally coincident⁴. Anyway, the dynamic of the LMF systems is driven by the state function and we shall focus on its properties.

We stress that the presence of a unique phase transition, in both the Linear Threshold Model (LTM) and the PALTM, is related to the convexities of the functions $\psi(x)$ and $\dot{\psi}(x)$ respectively. This two functions shall be convex in the first part of the interval $[0, 1]$ and then concave. The following simple result provide a sufficient condition to preserve such convexity intervals in the LMF state functions of a network with agent of different degree. First we state the inflection points of the functions $f_{d,r}(x)$.

Lemma 3.5. *Consider $d \geq 2$ and $r \in \{1, 2, \dots, d\}$ and define*

$$\bar{x}_{d,r} := \frac{r-1}{d-1}.$$

In the interval $[0, 1]$, the function

$$f_{d,r}(x) = \sum_{i=r}^d \binom{d}{i} x^i (1-x)^{d-i}$$

has a unique inflection point in $\bar{x}_{d,r}$ and is strictly convex for $x \in (0, \bar{x}_{d,r})$ and strictly concave for $x \in (\bar{x}_{d,r}, 1)$.

Corollary 3.6. *Consider a convex combination of the functions $f_{d,r}(x)$ such that the terms with $d \geq 2$ and $r \in \{1, 2, \dots, d\}$ all share the same $\bar{x}_{d,r} = \bar{x}$. Then, the convex combination has a unique inflection point in $[0, 1]$ (in \bar{x}), is strictly convex in $(0, \bar{x})$ and strictly concave in $(\bar{x}, 1)$.*

The proofs descend directly from the properties of the functions $f_{d,r}(x)$, described in the Appendix B. If in the network all the agents v with out-degree $d_v \geq 2$ and threshold $r_v \notin \{0, d_v + 1\}$ share the same ratio $\frac{r_v-1}{d_v-1}$ the LMF state function $\psi(x)$ (for the LTM, $\dot{\psi}(x)$ for the PALTM) has an unique inflection point, being first convex then concave. Correspondingly, the state sequence

⁴They coincide if the joint degree distribution $p_{d,k}$ factorizes like $p_d p_k$.

$x[t]$ and its limit have qualitative behaviors analogous to those described in the Subsection 3.3.1 for regular network with just one threshold value.

We stress that the above condition is sufficient but far from being necessary. In fact, it is quite restrictive: in a regular network it allows only one non-trivial threshold.

3.4.1 Asymptotic activation in the PALTM

Assume there are no agents with threshold $r_v = 0$ or $r_v = d_v + 1$ and consider the PALTM. Further, assume that the initially active agents are selected uniformly and independently at random, forming a fraction α of the total size. The corresponding LMF state function is

$$\dot{\psi}(x) = \alpha + (1 - \alpha) \sum_{d,k,r} \frac{k}{\bar{k}} p_{d,k} p_{r|d,k} f_{d,r}(x) \quad (3.7)$$

while the output function is

$$\dot{\phi}(x) = \alpha + (1 - \alpha) \sum_{d,k,r} p_{d,k} p_{r|d,k} f_{d,r}(x). \quad (3.8)$$

where $f_{d,r}(x) = \sum_{i=r}^d \binom{d}{i} x^i (1-x)^{d-i}$ and $p_{0|d,k} = p_{d+1|d,k} = 0$ for every d and k .

We describe a method to recover the asymptotic activation $\beta = \lim_{t \rightarrow \infty} y[t]$ and its dependance on α , similar to the approach of [14]. For $\alpha \in [0, 1]$ we define the set S_α of fixed points of the corresponding function $\dot{\psi}(x)$:

$$S_\alpha = \left\{ x \in [0, 1] : x = \dot{\psi}(x) \right\}.$$

Lemma 3.7. *With the standing assumption, consider the family of functions $h_{d,r}(x) := \sum_{i=0}^{r-1} \binom{d}{i} x^i (1-x)^{d-i-1}$ and the function*

$$F(x, \alpha) = 1 - (1 - \alpha) \sum_{d,k,r} \frac{k}{\bar{k}} p_{d,k} p_{r|d,k} h_{d,r}(x).$$

For $\alpha \in [0, 1]$ the set of fixed points of $\dot{\psi}(x)$ is

$$S_\alpha = \{1\} \cup \{x \in [0, 1] : F(x, \alpha) = 0\}.$$

Proof. The fixed point condition

$$x = \alpha + (1 - \alpha) \sum_{d,k,r} \frac{k}{\bar{k}} p_{d,k} p_{r|d,k} f_{d,r}(x)$$

is satisfied in $x = 1$ for any α . Therefore, with some manipulation, it is possible to take in common a factor $1 - x$:

$$\begin{aligned} x - \alpha - (1 - \alpha) \sum_{d,k,r} \frac{k}{\bar{k}} p_{d,k} p_{r|d,k} f_{d,r}(x) &= 0 \\ x - 1 + (1 - \alpha) - (1 - \alpha) \sum_{d,k,r} \frac{k}{\bar{k}} p_{d,k} p_{r|d,k} f_{d,r}(x) &= 0 \\ (1 - x) \left(1 - (1 - \alpha) \frac{1 - \sum_{d,k,r} \frac{k}{\bar{k}} p_{d,k} p_{r|d,k} f_{d,r}(x)}{1 - x} \right) &= 0. \end{aligned}$$

We rename with $F(x, \alpha)$ the second term of the first member above, defined for $x \neq 1$

$$F(x, \alpha) = 1 - (1 - \alpha) \frac{1 - \sum_{d,k,r} \frac{k}{\bar{k}} p_{d,k} p_{r|d,k} f_{d,r}(x)}{1 - x}$$

and use $\sum_{d,k,r} \frac{k}{\bar{k}} p_{d,k} p_{r|d,k} = 1$ to obtain

$$F(x, \alpha) = 1 - (1 - \alpha) \sum_{d,k,r} \frac{k}{\bar{k}} p_{d,k} p_{r|d,k} \frac{1 - f_{d,r}(x)}{1 - x}.$$

The function $h_{d,r}(x)$ have been definite to satisfy $h_{d,r}(x) = \frac{1 - f_{d,r}(x)}{1 - x}$. Therefore

$$F(x, \alpha) = 1 - (1 - \alpha) \sum_{d,k,r} \frac{k}{\bar{k}} p_{d,k} p_{r|d,k} h_{d,r}(x).$$

□

The equation $F(x, \alpha) = 0$ describes a curve in $[0, 1] \times [0, 1]$. To draw the curve, it is possible to deduce an expression for α as function of x such that $F(x, \alpha(x)) = 0$:

$$\alpha(x) = 1 - \frac{1}{\sum_{d,k,r} \frac{k}{\bar{k}} p_{d,k} c_{r,d,k} h_{d,r}(x)}.$$

On the other hand, for a given α there might be multiple x for which $F(x, \alpha) = 0$ contributing to the set S_α . The sequence $x[t]$ with initial condition $x[0] = \alpha$ and recursion $x[t + 1] = \psi(x[t])$ by construction starts in the first subinterval of the partition defined by S_α . The sequence is thus increasing and the limit $x[t]$ is the smallest fixed point of $\psi(x)$ in $[0, 1]$:

$$\lim_{t \rightarrow \infty} x[t] = \min S_\alpha.$$

We are interested in $\beta(\alpha) = \lim_{t \rightarrow \infty} y[t]$: using $y[t + 1] = \phi(x[t])$, the short discussion above proves the following Corollary.

Corollary 3.8. *With the standing assumption, the final activation of the network β , estimated with the LMF approximation as a function of the initial activation α is*

$$\beta(\alpha) = \overset{\circ}{\phi}(\min S_\alpha)$$

3.4.2 Vanishing initial activation in the PALTM

So far we have considered strictly positive seeds $\alpha > 0$ (i.e. fractions of initially active agents). However, if the number of initially active agents is sub-linear with respect to the size n of the network then the seed α vanishes. In this Subsection we study the consequences of a vanishing initial activation for the PALTM.

We obtain a condition that allows a vanishing seed to trigger the activation of a non vanishing fraction of agents. The condition is analogous what described in [4] as *susceptibility* of the directed random network, and is related to an analogous condition described in [3, 30] for undirected random networks.

With the same hypothesis of the previous Subsection observe that to $\alpha = 0$ corresponds an asymptotic activation $\beta = 0$, thus $\beta(0) = 0$. However, $\alpha = 0$ hides the possible presence of a finite number of initially active agents in a infinite network. Since there is no guarantee that $\beta(\alpha)$ is continuous for $\alpha = 0$, the case with vanishing α need a dedicate treatment. Indeed we will study the limit $\lim_{\alpha \rightarrow 0^+} \beta(\alpha)$ and highlight the condition by which its value is $\beta(0)$.

We rewrite the LMF functions (3.7) and (3.8) for the PALTM using, for brevity, the corresponding functions (3.1) and (3.2) of the LTM:

$$\begin{aligned}\overset{\circ}{\psi}(x) &= \alpha + (1 - \alpha) \psi(x) \\ \overset{\circ}{\phi}(x) &= \alpha + (1 - \alpha) \phi(x).\end{aligned}$$

With the previous assumptions, for $x = 0$ we have $\psi(0) = 0$ and $\phi(0) = 0$, thus $\overset{\circ}{\psi}(0) = \alpha$ and $\overset{\circ}{\phi}(0) = \alpha$. Using the expressions above, we remind that

$$\begin{aligned}S_\alpha &= \{x \in [0, 1] : x = \alpha + (1 - \alpha)\psi(x)\} \\ \beta(\alpha) &= \alpha + (1 - \alpha) \phi(\min S_\alpha).\end{aligned}$$

To verify the continuity of $\beta(\alpha)$ in $\alpha = 0$, we must study the limit $\lim_{\alpha \rightarrow 0^+} \beta(\alpha)$, and we shall obtain 0. We have

$$\lim_{\alpha \rightarrow 0^+} \beta(\alpha) = \lim_{\alpha \rightarrow 0^+} \alpha + (1 - \alpha) \phi(\min S_\alpha)$$

and recall that ϕ is continuous. Hence, to prove the continuity of $\beta(\alpha)$ in 0, we shall study the limit $\lim_{\alpha \rightarrow 0^+} \min S_\alpha$ and obtain 0. We proceed in the following

way. We indeed assume $\lim_{\alpha \rightarrow 0^+} \min S_\alpha = 0^+$ to obtain a self-consistency condition. If the condition is violated and the limit is not 0^+ , $\beta(\alpha)$ is not continuous in 0.

Let assume $\lim_{\alpha \rightarrow 0^+} \min S_\alpha = 0^+$, which means that S_α contains a small positive element that tends to zero as α tends to zero. The small element has to be positive since S_α is a subset of $[0, 1]$. With this element in mind, we consider the first order expansion of $\psi(x)$ for $x \rightarrow 0$:

$$\psi(x) = \psi(0) + \psi'(0)x + o(x) = \psi'(0)x + o(x),$$

where we used $\psi(0) = 0$. The smallest element of S_α , i.e. $\min S_\alpha$, must solve the equation

$$x = \alpha + (1 - \alpha)(\psi'(0)x + o(x))$$

that with some manipulations becomes

$$\begin{aligned} \alpha &= x - (1 - \alpha)\psi'(0)x + o(x) \\ \alpha &= x(1 - \psi'(0) + \alpha\psi'(0) + o(1)). \end{aligned}$$

We obtain:

$$x = \frac{\alpha}{1 - \psi'(0) + \alpha\psi'(0) + o(1)}$$

Observing that $\alpha\psi'(0) + o(1)$ is small and assuming $1 - \psi'(0) \neq 0$, we can expand to the first order the fraction and finally obtain

$$x = \frac{\alpha}{1 - \psi'(0)} + o(\alpha).$$

From the previous expression we recognize the self-consistency condition. If $1 - \psi'(0) > 0$ and $\alpha \rightarrow 0^+$ then $x \rightarrow 0^+$, in agreement with the initial assumption. This x is the smallest positive element of S_α so $\lim_{\alpha \rightarrow 0^+} \min S_\alpha = 0^+$. Therefore $\lim_{\alpha \rightarrow 0^+} \beta(\alpha) = 0$ and $\beta(\alpha)$ is continuous in the origin. A first order expansion gives, for $\alpha \rightarrow 0^+$,

$$\beta(\alpha) = \left(1 + \frac{\psi'(0)}{1 - \psi'(0)}\right)\alpha + o(\alpha).$$

Remember that $\dot{\psi}(x) = \alpha + (1 - \alpha)\psi(x)$ (thus $\dot{\psi}'(x) = (1 - \alpha)\psi'(x)$). With $\alpha = 0$, the condition $\psi'(0) < 1$ means that the fixed point in $x = 0$ of $\dot{\psi}(x)$ is stable. However, the robustness with respect to a small change of α is of greater concern than the stability with respect to a small change of the state.

If $\psi'(0) < 1$ an infinitesimal seed cannot produce a non infinitesimal final activation.

On the other hand, if $1 - \psi'(0) < 0$, for $\alpha \rightarrow 0^+$ we obtain $x \rightarrow 0^-$. This solution x cannot belong to S_α , which thus does not contain any small positive element. Hence $\min S_\alpha$ is not vanishing and the limit $\lim_{\alpha \rightarrow 0^+} \alpha + (1 - \alpha) \phi(\min S_\alpha) = \lim_{\alpha \rightarrow 0^+} \beta(\alpha)$ is non zero. Therefore, if $1 - \psi'(0) < 0$ the function $\beta(\alpha)$ is discontinuous for $\alpha = 0$.

The first order analysis above is valid for $\psi'(x) \neq 1$. If $\psi'(0) > 1$ it is sufficient a vanishing fraction of active agents to trigger the activation of a non-vanishing fraction of agents. With the expression (3.1) of $\psi(x)$ we rewrite the condition $\psi'(0) > 1$:

$$\sum_{d,k,r} \frac{k}{\bar{k}} p_{d,k} p_{r|d,k} f'_{d,r}(0) > 1.$$

Since the first derivative of $f_{d,r}(x)$ in 0 satisfy the property

$$f'_{d,r}(0) = \begin{cases} d & \text{if } r = 1 \\ 0 & \text{if } r \neq 1 \end{cases}$$

we obtain:

$$\begin{aligned} \sum_{d,k} \frac{d k}{\bar{k}} p_{d,k} p_{1|d,k} &> 1 \\ \sum_{d,k} d k p_{d,k} p_{1|d,k} &> \sum_{d,k} k p_{d,k}. \end{aligned}$$

The expression above stress the special role of the agents with threshold 1 in the initial stage of the activation spreading, If they are in sufficient number (factor $p_{d,k} p_{1|d,k}$) and they link enough neighbors (factor $d k$) they are able to multiply the effects of a vanishing seed.

Chapter 4

Control of the Local Mean-Field dynamics

In this Chapter, we consider the Local Mean Field (LMF) dynamics of the Linear Threshold Model (LTM) in the general case with time-varying thresholds distributions, which can be used to control the activation processes at a statistical level. This flexibility allows us to adapt the network to the actual conditions it is facing at a specific time. On one hand, we may be interested in keeping the activation level stable or as limited as possible, like in the financial contagion example. A fraction of agents may suffer an exogenous temporary shock to their states which can induce the systemic instability. The ability to adjust the thresholds dynamically can help to reduce the activation spread and favor the recovery of the network.

On the other hand, the goal may be the full activation of the network, like in marketing application. Instead of increasing the number of active stubborn, one may dynamically lower the activation thresholds to some fraction of agents and facilitate the activation spread as little as necessary.

In this Chapter we will use vectors and vector valued functions to make the notation more compact. We use boldface letters to denote vectors. The “all-zero” and the “all-one” vectors with the appropriate dimension are denoted with $\mathbf{0}$ and $\mathbf{1}$ respectively. Let \mathbf{v}, \mathbf{w} be two vectors in \mathbb{R}^n , \mathbf{v}^\top is the transpose of \mathbf{v} and $\mathbf{v}^\top \mathbf{w} = \sum_i v_i w_i$. The short forms $\mathbf{v} \succ \mathbf{w}$ and $\mathbf{v} \succeq \mathbf{w}$ respectively mean that $v_i > w_i$ and $v_i \geq w_i$ for every i . Let $\mathbf{f}(x) : \mathbb{R} \rightarrow \mathbb{R}^n$ be a *vector valued* function, where each component is the function $f_i(x) : \mathbb{R} \rightarrow \mathbb{R}$; the symbol \mathbf{f}' denotes the vector whose components are the first derivatives of the components of \mathbf{f} , namely $(\mathbf{f}')_i = \frac{df_i}{dx}$. Finally, given the function $g(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}$ whose argument is a vector, the gradient $\nabla g(\mathbf{x})$ is the vector of partial derivatives with $(\nabla g)_i = \frac{\partial g}{\partial x_i}$.

4.1 Problem formulation

The equations of the LMF dynamical system were proved in Lemma 2.2 of Chapter 2, in the general case with time-varying threshold distributions. In Chapter 3 we changed notation to $x[t]$ for the edge dynamic and $y[t]$ for the vertex dynamic. The LMF dynamical system is

$$\begin{cases} x[t+1] = \sum_{d,k,r} \frac{k p_{d,k}}{k} p_{r|d,k}[t] f_{d,r}(x[t]) \\ y[t+1] = \sum_{d,k,r} p_{d,k} p_{r|d,k}[t] f_{d,r}(x[t]) \end{cases} \quad (4.1)$$

with initial condition $x[0] = x_0$, $y[0] = y_0$ and where $f_{d,r}(x) = \mathbb{P}(\text{Bin}(d, x) \geq r) = \sum_{i=r}^d \binom{d}{i} x^i (1-x)^{d-i}$.

The conditional threshold distributions $p_{r|d,k}[t]$ are allowed to change during the dynamic provided that they satisfy certain constraints, which may depend on the application. We call *natural* the group of constraints which define the “statistic” nature of the conditional thresholds distribution $p_{r|d,k}[t]$. They are

$$\begin{cases} p_{r|d,k}[t] \geq 0 & \forall d, k, r \\ \sum_r p_{r|d,k}[t] = 1 & \forall d, k \\ 0 \leq x[t], y[t] \leq 1 \end{cases} \quad (4.2)$$

and shall hold at any time $t \geq 0$ considered. For practical purposes we include $x[t], y[t] \in [0, 1]$ among the natural constraints. The joint degree distribution $p_{d,k}$ is assumed fixed in the LMF system (4.1). It shall clearly satisfy $p_{d,k} \geq 0$ for every d, k , and $\sum_{d,k} p_{d,k} = 1$.

Further constraints can be added when solving more specific problems. Constraints that do not mix variables of different time steps are preferable as they are easier to be treated. We express such constraints with a set $P_{t,x[t],y[t]}$ which contains the values of the control variables allowed at time t and may depend on the current state $x[t]$ and output $y[t]$. The additional, *specific* constraints, are collectively expressed as

$$p_{r|d,k}[t] \in P_{t,x[t],y[t]} \quad (4.3)$$

which should again hold for any time $t \geq 0$ considered.

To formulate an optimal control problem we shall define a cost function, also called *objective function*. Recall that the LMF dynamical system (4.1) is a deterministic discrete-time system. We fix a time horizon T and we consider a deterministic objective function, *additive over time*: the cost incurred at each iteration t accumulates over time. Let denote with $\gamma_t(x[t], y[t], p_{r|d,k}[t])$ the cost incurred at iteration t , whereas $\gamma_T(x[T], y[T])$ is the terminal cost. The total

cost is:

$$\Gamma(x[\cdot], y[\cdot], p_{r|d,k}[\cdot]) = \gamma_T(x[T], y[T]) + \sum_{t=0}^{T-1} \gamma_t(x[t], y[t], p_{r|d,k}[t]) \quad (4.4)$$

where we used $x[\cdot]$, $y[\cdot]$ to denote the sequences $x[t]$, $y[t]$ for $t \in \{0, 1, \dots, T\}$ and $p_{r|d,k}[\cdot]$ to denote the sequence of controls $p_{r|d,k}[t]$ for $t \in \{0, 1, \dots, T-1\}$.

We are ready to formulate the optimal control problem:

$$\begin{aligned} \min_{p_{r|d,k}[\cdot]} \quad & \Gamma(x[\cdot], y[\cdot], p_{r|d,k}[\cdot]) \\ \text{subject to} \quad & (4.1), (4.2), (4.3) \quad \forall t \in \{0, \dots, T-1\}. \end{aligned} \quad (4.5)$$

In problem (4.5), the state and output sequences $x[\cdot]$ and $y[\cdot]$ are considered like variables and the LMF dynamical system (4.1) is imposed as a set of equality constraints. By explicitly using the LMF equations in (4.4), the total cost can be expressed as a function $\Gamma = \Gamma(x_0, y_0, p_{r|d,k}[\cdot])$ of the initial conditions and the time-varying threshold sequences only: in this way the equality constraints are handled implicitly.

The LMF dynamical system (4.1) represents the expected activation behavior of a large directed configuration model random network, with joint degree distribution $p_{d,k}$ and time-varying thresholds distributions $p_{r|d,k}[\cdot]$. The optimal sequence of controls does not depend on the specific node states, but rather on their expectation under the LMF approximation. We remind that at time $t+1$ the expected activation state of an agent with degree d and threshold r is $f_{d,r}(x[t])$, with an expected impact on the network of $\frac{k}{\bar{k}} f_{d,r}(x[t])$, where k is the in-degree and \bar{k} the average in-degree.

Along the same line, the solution of the problem provides the optimal sequences $p_{r|d,k}[\cdot]$ of conditional threshold distributions which, subject to the constraints (4.2) and (4.3), minimize the total cost Γ . The optimal control sequences do not indicate on which specific nodes it is optimal to intervene and shall be interpreted in a statistical sense: they provide the pattern of thresholds which statistically gives the best overall behavior. In fact on a specific instance of the network and at time t , there are usually many ways to assign the individual thresholds r to the nodes with given out-degree d and in-degree k , each compliant with the threshold distribution $p_{r|d,k}[t]$.

4.2 A regular network case

In this Section and in the next one, we consider a simplified problem on a regular network where the available thresholds adjustments have been limited. We consider the Linear Threshold Model and assume that the initial configuration

of active agents can activate all agents in the network. The Local Mean Field Dynamic of the simplified network will be the object of numerical simulations, to investigate how to mitigate the activation outbreak by controlling the thresholds. We will also provide some analytic insights to understand the numerical results.

We consider a regular and directed configuration model random graph $G = (V, E)$, where every agent has the same out-degree and in-degree: $\forall v \in V, d_v = k_v = d_G$. Each agent $v \in V$ has a nominal activation threshold \bar{r}_v which can be increased by one unit, thus at any time $r_v[t] \in \{\bar{r}_v, \bar{r}_v + 1\}$. To increment a threshold of nominal value r for one unit of time, the cost is ω_r . We assume a budget bound $b[t]$ on the expenditure at any time t as well, i.e.:

$$|V|^{-1} \sum_{v \in V} (r_v[t] - \bar{r}_v) \omega_{\bar{r}_v} \leq b[t]. \quad (4.6)$$

There are no further restrictions on how and when the thresholds can be adjusted.

The initial activation profile of the network contains a fraction $x_0 \in [0, 1]$ of active agents. We suppose that the initial configuration is not stable under the dynamic induced by the Linear Threshold Model, i.e. the initial configuration will drive the network to full activation (or to a configuration with a large fraction of active agents) before a time horizon T .

We wish to reduce (or contain) the activation spread at the time horizon T , with minimal effort. To understand the kind of agents which would benefit most of a threshold increment, we consider the Local Mean Field approximation of the problem. We shall introduce the appropriate statistical quantities to reformulate the problem described above.

In a directed regular network the joint degree distribution becomes trivial ($p_{d_G, d_G} = 1$), thus we drop the dependence on d and k of the variables and the corresponding sums over d and k in the LMF equations. The nominal threshold distribution is given by the fractions \bar{p}_r , for $r \in \{0, 1, \dots, d_G + 1\}$, with $\bar{p}_r \geq 0$ and $\sum_r \bar{p}_r = 1$. The actual time-varying threshold distribution is given by the fractions $p_r[t]$. Nominal and actual distribution can be written in a vector form with $\bar{\mathbf{p}}, \mathbf{p}[t] \in \mathbb{R}^{d_G+2}$. Since the network is in-regular, the fraction of active “edges” coincides with that of active agents. Therefore it is enough to use the state equation of the LMF dynamical system and the state sequence $x[t]$ with $x[0] = x_0$. With the vector notation and considering the last comment, the natural constraints become

$$\begin{cases} \mathbf{p}[t] \succcurlyeq \mathbf{0} \\ \mathbf{1}^\top \mathbf{p}[t] = 1 \\ 0 \leq x[t] \leq 1 \end{cases} \quad \forall t \in \{0, \dots, T-1\} \quad (4.7)$$

The LMF dynamical system (4.1) simplifies to

$$x[t+1] = \sum_{r=0}^{d_G+1} p_r[t] f_{d_G,r}(x[t]) \quad \forall t \in \{0, \dots, T-1\}$$

with initial condition $x[0] = x_0 \in [0, 1]$ and where the functions $f_{d,r} : [0, 1] \rightarrow [0, 1]$ are defined as $f_{d,r}(x) = \mathbb{P}(\text{Bin}(d, x) \geq r) = \sum_{i=r}^d \binom{d}{i} x^i (1-x)^{d-i}$. We introduce the *vector valued* function $\mathbf{f}_d : [0, 1] \rightarrow [0, 1]^{d+2}$ such that $(\mathbf{f}_d(x))_r = f_{d,r}(x)$ to rewrite the LMF equation in a compact vectorial form:

$$x[t+1] = \mathbf{f}_{d_G}(x[t])^\top \mathbf{p}[t] \quad \forall t \in \{0, \dots, T-1\}. \quad (4.8)$$

The characteristic to increase each nominal threshold at most of one unit imposes a special structure on the actual time-varying threshold distribution, which the following re-parametrization captures effectively. We introduce d_G+1 new variables $u_r[t]$, one for each $r \in \{0, \dots, d_G\}$, collectively stacked in the vector $\mathbf{u}[t] \in \mathbb{R}^{d_G+1}$. Each variable $u_r[t]$ describes the fraction of agents of nominal threshold r which at time t has the increased threshold, i.e. $u_r[t] = |V|^{-1} \sum_{v \in V} (r_v[t] - \bar{r}_v) \delta_{r, \bar{r}_v}$ where δ_{r, \bar{r}_v} is the Kronecker delta. The time-varying threshold distribution becomes

$$\begin{cases} p_0[t] &= \bar{p}_0 - u_0[t] \\ p_r[t] &= \bar{p}_r - u_r[t] + u_{r-1}[t] \quad r \in \{1, \dots, d_G\} \\ p_{d_G+1}[t] &= \bar{p}_{d_G+1} + u_{d_G}[t] \end{cases} \quad (4.9)$$

The variables $u_r[t]$ need to satisfy the box constraints

$$u_r[t] \in [0, \alpha_r \bar{p}_r] \quad \forall r \in \{0, \dots, d_G\}, \quad \forall t \in \{0, \dots, T-1\} \quad (4.10)$$

where the parameters $\alpha_r \in [0, 1]$ make the constraints more stringent. The above parametrization will prove useful while discussing some insights about the optimized dynamic.

With some algebra, the previous parametrization can be expressed in a more compact form. We introduce the matrices $L \in \mathbb{R}^{(d_G+2) \times (d_G+1)}$ and $J \in \mathbb{R}^{(d_G+1) \times (d_G+2)}$ such that

$$\begin{aligned} L_{i,i} &= -1, \quad L_{i+1,i} = 1 \text{ for } i \in \{0, \dots, d_G\}, \text{ otherwise } L_{i,j} = 0 \\ J_{i,i} &= \alpha_i \text{ for } i \in \{0, \dots, d_G\}, \text{ otherwise } J_{i,j} = 0. \end{aligned}$$

We also introduce the family of functions $g_{d,r} : [0, 1] \rightarrow [0, 1]$:

$$\begin{aligned} g_{d,r}(x) &= \mathbb{P}(\text{Bin}(d, x) = r) = \binom{d}{r} x^r (1-x)^{d-r} \\ &= -(f_{d,r+1}(x) - f_{d,r}(x)) \end{aligned}$$

and the *vector valued* function $\mathbf{g}_d : [0, 1] \rightarrow [0, 1]^{d+1}$ such that $(\mathbf{g}_d(x))_r = g_{d,r}(x)$.

With the above definitions, the parametrization (4.9) become

$$\mathbf{p}[t] = \bar{\mathbf{p}} + L\mathbf{u}[t].$$

Using the expression above and observing that $\mathbf{f}_d(x)^\top L = -\mathbf{g}_d(x)$ the LMF state equation (4.8) takes the form

$$x[t+1] = \mathbf{f}_{d_G}(x[t])^\top \bar{\mathbf{p}} - \mathbf{g}_{d_G}(x[t])^\top \mathbf{u}[t] \quad \forall t \in \{0, \dots, T-1\} \quad (4.11)$$

that explicitly exploits $\mathbf{u}[t]$ and the structure of the case study. The specific box constraints (4.10) become

$$\mathbf{0} \preceq \mathbf{u}[t] \preceq J\bar{\mathbf{p}} \quad \forall t \in \{0, \dots, T-1\}. \quad (4.12)$$

To satisfy the natural constrains (4.7) it is enough to fulfill the box constraints above since it shall hold

$$\begin{cases} \bar{\mathbf{p}} \succeq \mathbf{0} \\ \mathbf{1}^\top \bar{\mathbf{p}} = 1 \\ 0 \leq x[t] \leq 1 \end{cases} \quad \forall t \in \{0, \dots, T\}. \quad (4.13)$$

With the new variable it is easy to express the budget bound (4.6) as $\sum_{r=0}^{d_G} \omega_r u_r[t] \leq b[t]$ which in vectorial form is

$$\boldsymbol{\omega}^\top \mathbf{u}[t] \leq b[t] \quad \forall t \in \{0, \dots, T-1\} \quad (4.14)$$

where $\boldsymbol{\omega} \in \mathbb{R}^{d_G+1}$ is the vector of weights ω_r . If the budgets $b[t]$ are larger than $\boldsymbol{\omega}^\top J\bar{\mathbf{p}}$ the constraint is never active. Notice that the box constraints (4.12) and the budget constraints (4.14) can be cast in the generic form (4.3). Further constraints (e.g. on the new variables \mathbf{u}) can be easily added if they do not mix different time steps.

A general total cost was introduced in (4.4), under the sole hypothesis of being *additive over time*. The corresponding of (4.4) for a regular network is a function $\Gamma(x[\cdot], \mathbf{p}[\cdot])$ where $x[\cdot]$ is the sequences of $x[t]$ for $t \in \{0, 1, \dots, T\}$ and $\mathbf{p}[\cdot]$ is the time-varying threshold distribution sequence for $t \in \{0, 1, \dots, T-1\}$. Given the new parametrization, we can express the objective function of the case study directly using the sequence of variables $\mathbf{u}[t]$ for $t \in \{0, 1, \dots, T-1\}$, denoted with $\mathbf{u}[\cdot]$:

$$\Gamma(x[\cdot], \mathbf{u}[\cdot]) = \gamma_T(x[T]) + \sum_{t=0}^{T-1} \gamma_t(x[t], \mathbf{u}[t])$$

where γ_t is the running cost and γ_T the terminal cost.

To keep things simple, we assume that the running cost function γ_t and the terminal cost function γ_T are linear and are expressed as linear combinations of their arguments with constant positive coefficients. In particular the vector $\mathbf{c}_u \succ \mathbf{0}$ contains the coefficients of the control variables $\mathbf{u}[t]$ while $c_x \geq 0$ is the coefficient of the state $x[t]$, for $t \in \{1, \dots, T-1\}$. There is no cost associated to the state $x[0]$ since it is not possible to optimize on x_0 . Instead, for $t = T$ the terminal cost coefficient is $c_T \geq 0$. In short, we have:

$$\begin{aligned} \gamma_T(x) &= c_T x \\ \gamma_t(x, \mathbf{u}) &= \begin{cases} t = 0 & \mathbf{c}_u^\top \mathbf{u} \\ t > 0 & c_x x + \mathbf{c}_u^\top \mathbf{u} \end{cases} \end{aligned}$$

With these assumptions and definitions, the total cost becomes

$$\Gamma(x[\cdot], \mathbf{u}[\cdot]) = \mathbf{c}_u^\top \mathbf{u}[0] + \sum_{t=1}^{T-1} [c_x x[t] + \mathbf{c}_u^\top \mathbf{u}[t]] + c_T x[T]. \quad (4.15)$$

Bringing all together, we considered a large directed regular random network with degrees d_G and with a fraction x_0 of initially active nodes. We assumed the agents' thresholds to have a limited adjustability (as described before) and the nominal threshold distribution to be $\bar{\mathbf{p}}$. The optimal control problem of the case study is thus

$$\begin{aligned} \min_{\mathbf{u}[\cdot]} & \quad (4.15) \\ \text{subject to} & \quad (4.11), (4.12), (4.13), (4.14) \end{aligned} \quad (4.16)$$

The solution of such problem is the *optimal global control law* sequence $\mathbf{u}^*[\cdot]$.

The optimization problem (4.16) can be treated as an *open-loop optimal control problem* by minimizing the objective function with respect to all the controls at once (from $t = 0$ to $t = T - 1$). We stress that if we plug the dynamic law (4.11) in the objective function (4.15) this last only depends on the control sequence $\mathbf{u}[\cdot]$ and on the initial parameter x_0 , thus we avoid the equality constraints in the optimization problem.

A few simulations about the problem (4.16) are presented in the next Section, where a comparison with two sub-optimal approaches presented in the following subsections are shown as well. All the simulations are run using MATLAB[®] software. Our code adopts the the function `multistart` of the “Global Optimization Toolbox” which is based on the solver `fmincon` from the “Optimization ToolboxTM”. We provide a bunch of feasible control sequences to the function `multistart` to be used as start points for the solver. For each start point, the solver `fmincon` is called and a local optimal solution is found

using the “interior point” method. Finally, `multistart` groups the optimal points that are closer than a certain tolerance and ranks them according to the objective function. Since the problem may have local minima, using more starting points increase the chance to identify the global optimal sequence.

4.2.1 Model Predictive Control

The objective function (4.15) depends explicitly and implicitly on the control vectors $\mathbf{u}[0], \mathbf{u}[1], \dots, \mathbf{u}[T - 1]$. If the time horizon T is large, the global optimization problem (4.16) may be hard to solve. One alternative consists in the adoption of a sub-optimal control strategy. In this perspective, the problem (4.16) is suitable for the *Model Predictive Control* (hereafter MPC) approach, which is based on a sequence of online numerical optimization.

The MPC is an online optimization technique which follows the process and analyzes a limited number of control variables at any step. The MPC implementation is based on two tuning parameters: the prediction horizon T_x and the control horizon T_u , whose integer values satisfy $1 \leq T_u \leq T_x \leq T$.

At a generic time t , the state $x[t]$ becomes the initial condition to predict the process evolution for further T_x time steps. The predictions are used to solve a smaller optimization problem where the only $T_u \leq T_x$ control vectors ($\mathbf{u}[t], \dots, \mathbf{u}[t + T_u - 1]$) need to be optimally chosen. By default the remaining control vectors $\mathbf{u}[t + T_u], \dots, \mathbf{u}[t + T_x - 1]$ are set to zero. Once these control vectors are chosen, the first element (e.g. $\mathbf{u}[t]$) is input to the system, while the other elements are discarded. The process moves forward to $t + 1$ following the system dynamic and the selected control $\mathbf{u}[t]$: we obtain the new state $x[t + 1]$, to be used to start a new optimization problem. The prediction and control “windows” are moved one step further: for this reason the MPC approach is also known as *receding horizon* strategy. The MPC technique is sub-optimal because the problem is partitioned in a sequence of sub-problems solved consecutively. However, in the simulations of our problem we observed that the MPC algorithm could usually find control sequences with performances close to those of the global optimization algorithm, but with lighter computations.

The MPC algorithm can be organized as a “main” procedure and a “sub-routine”: we describe the two procedures using the pseudo-code Algorithm 4.1 in the following. Assuming that the problem data and coefficients ($d_G, \bar{\mathbf{p}}, \dots$) are known, the main procedure requires the *prediction horizon* T_x and the *control horizon* T_u to be set. In the subroutine optimization these parameters will determine the lengths of the prediction and control windows which, starting at t , specify the time steps contributing to the objective functions and the optimizable control vectors respectively.

The algorithm starts with the initial condition $x[0] = x_0$ and proceeds with

the sequential solutions of small optimization problems until the end of the time interval $[0, T]$ is reached. Let us assume that at a time t and the system state is $x[t]$. In the following we describe what the main procedure of the MPC algorithm will do at any iteration.

1. First it computes the *effective* prediction and control horizon, \hat{T}_x and \hat{T}_u respectively, as

$$\begin{aligned}\hat{T}_x &= \min\{T_x, T - t\} \\ \hat{T}_u &= \min\{T_u, T - t\}.\end{aligned}$$

The prediction and control horizon are usually much shorter than the interval of interest. In our implementation, we reduce the receding horizons as we approach the end of the interval of interest.

2. Then the main procedure calls the subroutine with the state $x[t]$, the effective receding horizons \hat{T}_x and \hat{T}_u and the portion of interest of the budget bound sequence $b[t]$. The subroutine (described later) computes and returns the control $\mathbf{u}[t]$.
3. The main procedure applies the control vector $\mathbf{u}[t]$ to the system and computes the next state $x[t+1]$. The time counter then advances by one step, and the procedure is repeated until the horizon is reached at time T .

To describe the subroutine, we denote the variables involved with a hat on top, and use the letter s for the internal time variable. The subroutine receives the effective prediction horizon \hat{T}_x and the effective control horizon \hat{T}_u together with the initial condition $\hat{x}_0 = x[t]$ and the portion of interest of the budget bounds: $\hat{b}[s] = b[t + s]$ for $s \in \{0, \dots, \hat{T}_u - 1\}$. We assume that the nominal threshold distribution and the coefficient and constraints, which do not change between different iterations, are known and thus are not provided explicitly. Let $\hat{x}[\cdot] \in [0, 1]^{\hat{T}_x+1}$ denote the sequence of states $\hat{x}[0], \hat{x}[1], \dots, \hat{x}[\hat{T}_x]$ and let $\hat{\mathbf{u}}[\cdot] \in \mathbb{R}^{(d_G+1) \times \hat{T}_u}$ denote the sequence of control vectors $\hat{\mathbf{u}}[0], \dots, \hat{\mathbf{u}}[\hat{T}_u - 1]$. The dynamical system equations in the subroutine predict the process until time \hat{T}_x , with the controls available only from $s = 0$ to $s = \hat{T}_u - 1$:

$$\begin{cases} \hat{x}[0] &= \hat{x}_0 \\ \hat{x}[s+1] &= \mathbf{f}_{d_G}(\hat{x}[s])^\top \bar{\mathbf{p}} - \mathbf{g}_{d_G}(\hat{x}[s])^\top \hat{\mathbf{u}}[s] & s \in \{0, \dots, \hat{T}_u - 1\} \\ \hat{x}[s+1] &= \mathbf{f}_{d_G}(\hat{x}[s])^\top \bar{\mathbf{p}} & s \in \{\hat{T}_u, \dots, \hat{T}_x - 1\}. \end{cases} \quad (4.17)$$

It holds

$$0 \leq \hat{x}[s] \leq 1 \quad s \in \{0, \dots, \hat{T}_x\} \quad (4.18)$$

while the available controls are subject to the constraints

$$\mathbf{0} \preceq \hat{\mathbf{u}}[s] \preceq J\bar{\mathbf{p}} \quad s \in \{0, \dots, \hat{T}_u - 1\} \quad (4.19)$$

$$\boldsymbol{\omega} \cdot \hat{\mathbf{u}}[s] \leq \hat{b}[s] \quad s \in \{0, \dots, \hat{T}_u - 1\}. \quad (4.20)$$

In the time steps $s \in \{\hat{T}_u, \dots, \hat{T}_x - 1\}$ no control is available, i.e. the controls vectors are set to zero by default.

The objective function in the MPC subroutine, corresponding to (4.15), is

$$\hat{\Gamma}(\hat{x}[\cdot], \hat{\mathbf{u}}[\cdot], \hat{T}_x, \hat{T}_u) = \mathbf{c}_u^\top \sum_{s=0}^{\hat{T}_u-1} \hat{\mathbf{u}}[s] + c_x \sum_{s=1}^{\hat{T}_x-1} \hat{x}[s] + c_T \hat{x}[\hat{T}_x] \quad (4.21)$$

The MPC subroutine solves the optimization problem

$$\begin{aligned} \min_{\hat{\mathbf{u}}[\cdot]} \quad & (4.21) \\ \text{subject to} \quad & (4.17), (4.18), (4.19), (4.20) \end{aligned} \quad (4.22)$$

whose solution is the sequence $\hat{\mathbf{u}}^*[\cdot]$. Finally the first element $\hat{\mathbf{u}}^*[0]$ is returned to the main procedure.

The optimization problem (4.22) in the MPC subroutine is again a *open-loop optimal control problem*: the simulations are again based on the function `multistart` and the solver `fmincon`, as described for the global optimization. In fact, note that, if $\hat{T}_x = \hat{T}_u = T$ and $t = 0$ (so that $\hat{x}_0 = x_0$), the specific subroutine optimization problem (4.22) and the global optimization problem (4.16) coincide. The results of the simulations done using the MPC approach are presented in the next Section.

4.2.2 Linearization method

We present a heuristic method based on the linearization of the objective function that we developed to reproduce the sub-optimal control sequences found by the MPC approach. We call *linearization method* such heuristic approach that shall be thought as a linear approximation of the MPC approach with control horizon $T_u = 1$. We continue to use the notation introduced in the previous Subsection.

To describe the heuristic, we start with an instance of the MPC subroutine which shall solve the problem (4.22) and return the control vector $\hat{\mathbf{u}}[0]$. We assume the effective prediction horizons is generically $\hat{T}_u \geq 1$. We remind that the subroutine objective function $\hat{\Gamma}(\hat{x}[\cdot], \hat{\mathbf{u}}[\cdot], \hat{T}_x, \hat{T}_u)$ of equation (4.21) depends on the control variables $\hat{\mathbf{u}}[0], \hat{\mathbf{u}}[1], \dots, \hat{\mathbf{u}}[\hat{T}_u - 1]$ both explicitly and implicitly via the dynamics (4.17).

Algorithm 4.1. Model Predictive Control scheme

```

1: procedure MAIN( $x_0, T, T_x, T_u$ )
2:   Set  $t = 0, x[0] = x_0$ , initialize variables;
3:   while  $t < T$  do
4:      $\hat{T}_u \leftarrow \min\{T_u, T - t\}$ ;
5:      $\hat{T}_x \leftarrow \min\{T_x, T - t\}$ ;
6:      $\mathbf{u}[t] \leftarrow \text{SUBROUTINE}(x[t], \hat{T}_u, \hat{T}_x, b[t + s] \text{ for } s \in \{0, \dots, \hat{T}_u - 1\})$ ;
7:      $x[t+1] \leftarrow \mathbf{f}_{d_G}(x[t])^\top \bar{\mathbf{p}} - \mathbf{g}_{d_G}(x[t])^\top \mathbf{u}[t]$ ;
8:      $t \leftarrow t + 1$ ;
9:   end while
10:  return  $\mathbf{u}[\cdot], x[\cdot]$ ;
11: end procedure

12: procedure SUBROUTINE( $\hat{x}_0, \hat{T}_u, \hat{T}_x, \hat{b}[s] \text{ for } s \in \{0, \dots, \hat{T}_u - 1\}$ )
13:  Initialize variables;
14:  Solve the optimization problem (4.22);
15:  return  $\hat{\mathbf{u}}^*[0]$ ;
16: end procedure

```

Let suppose for an instant that the objective function can be written as an explicit affine combination¹ of the control variables

$$\hat{\Gamma}(\hat{x}[\cdot], \hat{\mathbf{u}}[\cdot], \hat{T}_x, \hat{T}_u) = w + \mathbf{v}_0^\top \hat{\mathbf{u}}[0] + \mathbf{v}_1^\top \hat{\mathbf{u}}[1] + \dots + \mathbf{v}_0^\top \hat{\mathbf{u}}[\hat{T}_u - 1] \quad (4.23)$$

where $w \in \mathbb{R}$ and $\mathbf{v}_s \in \mathbb{R}^{d_G+1}$ are the appropriate coefficient.

Such an objective function would dramatically simplify the optimization problem (4.22). On one hand the equality constraint have been already accounted for. On the other hand all the inequality constraints (box constraints (4.19) and budget bounds (4.20)) do not mix different time steps. Hence, the problem (4.22) can be exactly decoupled in \hat{T}_u independent sub-problems, one for each $s \in \{0, \dots, \hat{T}_u - 1\}$, looking for the best $\hat{\mathbf{u}}[s]$ to optimize $\mathbf{v}_s^\top \hat{\mathbf{u}}[s]$ with constraints $\mathbf{0} \preceq \hat{\mathbf{u}}[s] \preceq J\bar{\mathbf{p}}$ and $\boldsymbol{\omega} \cdot \hat{\mathbf{u}} \leq \hat{b}$. The subroutine shall eventually return $\hat{\mathbf{u}}[0]$ thus just the sub-problem for $s = 0$ needs to be actually solved! Using $\hat{\mathbf{u}}$ for $\hat{\mathbf{u}}[0]$ and setting $\hat{b} = b[t]$ the problem is

$$\begin{aligned} \min_{\hat{\mathbf{u}}} \quad & \mathbf{v}_0^\top \hat{\mathbf{u}} \\ \text{s.t.} \quad & \mathbf{0} \preceq \hat{\mathbf{u}} \preceq J\bar{\mathbf{p}} \\ & \boldsymbol{\omega} \cdot \hat{\mathbf{u}} \leq \hat{b}. \end{aligned} \quad (4.24)$$

¹An *affine combination* is just a linear combination to which a constant term is added.

The hypothetical problem above belongs to the important class of *linear optimization problem*, because both the objective and constrains are affine functions of the optimization variable. In principle it is possible to solve linear optimization problem (also called *linear programs*) exactly; refer to [12, p. 146] and [10] for further details. Later, we will show that the simple structure of problem (4.24) makes it possible to build an optimal solution “by hand”, without invoking any solver.

The explicit MPC objective function in general does not depend linearly on the control variables. The idea behind the heuristic method is to linearly approximate the objective function $\hat{\Gamma}(\hat{x}[\cdot], \hat{\mathbf{u}}[\cdot], \hat{T}_x, \hat{T}_u)$ and obtain the vector \mathbf{v}_0 of the expansion (4.23), then solve the linear program (4.24) to get the optimal control $\hat{\mathbf{u}}^*[0]$. The pseudocode of the linearization method is contained in Algorithm 4.2 and is organized with a “main” and “subroutine” structure as the MPC Algorithm 4.1. The heuristic has only one tuning parameter called *lookahead* T_x , analogous to the MPC prediction horizon. The correspondent of the control horizon is not necessary since in effect it would be equal to one. The other coefficients and constrains are assumed to be known implicitly. The linearization method splits the global optimization in the sequential solution of T sub-problems, similarly to what the MPC was doing. Each subproblem is linearly approximated further: each call of the subroutine computes the coefficient \mathbf{v}_0 of first-order objective function expansion and solves the corresponding linear program. We describe in the following how to obtain the coefficient \mathbf{v}_0 at each step and solve the corresponding linear program.

We start with the first order approximation of the objective function (4.21) where \hat{T}_u is substituted by 1. The substitution is without loss of generality since we only need the analogous of the constant vector coefficient \mathbf{v}_0 of the affine objective function (4.23), whose value is independent from the choice of \hat{T}_u . We remark that this is the reason why the linearization method only requires the lookahead parameter T_x : an equivalent of T_u would not affect the result. When comparing the linearization method with the MPC approach, we should remember that fairness would suggest setting $T_u = 1$ in the MPC. For simplicity (and to avoid useless computations) from now on we forget about \hat{T}_u and work on the objective function $\hat{\Gamma}(\hat{x}[\cdot], \hat{\mathbf{u}}, \hat{T}_x, 1)$, which depends on only the control vector $\hat{\mathbf{u}} = \hat{\mathbf{u}}[0]$.

We wish to linearize $\hat{\Gamma}(\hat{x}[\cdot], \hat{\mathbf{u}}, \hat{T}_x, 1)$ around $\hat{\mathbf{u}} = \mathbf{0}$ for a generic lookahead $\hat{T}_x \geq 1$. The sub-routine state sequence $\hat{x}[\cdot] = \{\hat{x}[0], \hat{x}[1], \dots, \hat{x}[\hat{T}_x]\}$ follows the LMF dynamical system

$$\begin{cases} \hat{x}[1] &= \mathbf{f}_{d_G}(\hat{x}[0])^\top \bar{\mathbf{p}} - \mathbf{g}_{d_G}(\hat{x}[0])^\top \hat{\mathbf{u}} \\ \hat{x}[s+1] &= \mathbf{f}_{d_G}(\hat{x}[s])^\top \bar{\mathbf{p}} \end{cases} \quad s \in \{1, \dots, \hat{T}_x - 1\} \quad (4.25)$$

with initial condition $\hat{x}[0] = \hat{x}_0 = x[t]$ (assuming the main procedure at time t).

Algorithm 4.2. Linearization Method scheme

```

1: procedure MAIN( $x_0, T, T_x$ )
2:   Set  $t = 0, x[0] = x_0$ , initialize variables;
3:   while  $t < T$  do
4:      $\hat{T}_x \leftarrow \min\{T_x, T - t\}$ ;
5:      $\mathbf{u}[t] \leftarrow$  SUBROUTINE( $x[t], \hat{T}_x, b[t]$ );
6:      $x[t+1] \leftarrow \mathbf{f}_{d_G}(x[t])^\top \bar{\mathbf{p}} - \mathbf{g}_{d_G}(x[t])^\top \mathbf{u}[t]$ ;
7:      $t \leftarrow t + 1$ ;
8:   end while
9:   return  $\mathbf{u}[\cdot], x[\cdot]$ ;
10: end procedure

11: procedure SUBROUTINE( $\hat{x}_0, \hat{T}_x, \hat{b}$ )
12:   Compute the linearization coefficient  $\mathbf{v}_0$  using Proposition 4.1;
13:   Solve the Linear Program (4.24) using Proposition 4.2;
14:   return  $\hat{\mathbf{u}}^*[0]$ ;
15: end procedure

```

The objective function $\hat{\Gamma}(\hat{x}[\cdot], \hat{\mathbf{u}}, \hat{T}_x, 1)$ depends implicitly on the control $\hat{\mathbf{u}}$ via the recursive dynamical system above. We shall express the objective function as the composed scalar function $\hat{\Gamma}(\hat{\mathbf{u}})_{\hat{x}_0, \hat{T}_x} : \mathbb{R}^{d_G+1} \rightarrow \mathbb{R}$ only of the control variable $\hat{\mathbf{u}}$, using the initial condition \hat{x}_0 and the number of prediction steps \hat{T}_x as parameters. We write

$$\hat{\Gamma}(\hat{\mathbf{u}})_{\hat{x}_0, \hat{T}_x} = \hat{\Gamma}(\hat{x}[\cdot], \hat{\mathbf{u}}, \hat{T}_x, 1) = \mathbf{c}_\mathbf{u}^\top \hat{\mathbf{u}} + c_x \sum_{s=1}^{\hat{T}_x-1} \hat{x}[s] + c_T \hat{x}[\hat{T}_x] \quad (4.26)$$

where to each $\hat{x}[s]$ we shall explicitly substitute the recursion (4.25).

If the effective lookahead is just one step (i.e. $\hat{T}_x = 1$) the objective function $\hat{\Gamma}(\mathbf{0})_{\hat{x}_0, 1}$ is already affine in the control $\hat{\mathbf{u}}$, without any approximation. In fact the state sequence is simply $\hat{x}[\cdot] = \{\hat{x}[0], \hat{x}[1]\}$, with $\hat{x}[0] = \hat{x}_0$ and

$$\begin{aligned} \hat{x}[1] &= \mathbf{f}_{d_G}(\hat{x}[0])^\top \bar{\mathbf{p}} - \mathbf{g}_{d_G}(\hat{x}[0])^\top \hat{\mathbf{u}} \\ &= \mathbf{f}_{d_G}(\hat{x}_0)^\top \bar{\mathbf{p}} - \mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}}. \end{aligned}$$

Using the definition (4.26) we obtain

$$\begin{aligned} \hat{\Gamma}(\mathbf{0})_{\hat{x}_0, 1} &= \mathbf{c}_\mathbf{u}^\top \hat{\mathbf{u}} + c_T \hat{x}[1] \\ &= \mathbf{c}_\mathbf{u}^\top \hat{\mathbf{u}} + c_T \mathbf{f}_{d_G}(\hat{x}_0)^\top \bar{\mathbf{p}} - c_T \mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}} \\ &= c_T \mathbf{f}_{d_G}(\hat{x}_0)^\top \bar{\mathbf{p}} + (\mathbf{c}_\mathbf{u} - c_T \mathbf{g}_{d_G}(\hat{x}_0))^\top \hat{\mathbf{u}} \end{aligned} \quad (4.27)$$

which is an affine function of $\hat{\mathbf{u}}$. Therefore in the special case with $\hat{T}_x = 1$ the sub-routine optimization problem is already linear without any approximation.

For any other $\hat{T}_x > 1$ the objective function $\hat{\Gamma}(\hat{\mathbf{u}})_{\hat{x}_0, \hat{T}_x}$ is not linear and not even convex. To see this, it is enough to look at $\hat{x}[2]$

$$\hat{x}[2] = \mathbf{f}_{d_G}(\hat{x}[1])^\top \bar{\mathbf{p}} = \mathbf{f}_{d_G}(\hat{x}_0)^\top \bar{\mathbf{p}} - \mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}})^\top \bar{\mathbf{p}},$$

and remember that each non-trivial component of \mathbf{f}_{d_G} is a non-convex function. However, the objective $\hat{\Gamma}(\hat{\mathbf{u}})_{\hat{x}_0, \hat{T}_x}$ remains differentiable since composition of differentiable functions. We consider its first order Taylor expansion in the neighborhood of the origin $\hat{\mathbf{u}} = \mathbf{0}$:

$$\hat{\Gamma}(\hat{\mathbf{u}})_{\hat{x}_0, \hat{T}_x} = \hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x} + \nabla \hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x}^\top \hat{\mathbf{u}} + o(\|\hat{\mathbf{u}}\|) \quad (4.28)$$

where the vector $\nabla \hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x}$ is the gradient of $\hat{\Gamma}(\hat{\mathbf{u}})_{\hat{x}_0, \hat{T}_x}$ evaluated in $\hat{\mathbf{u}} = \mathbf{0}$. If the control variables are “small”, i.e. if $\|\hat{\mathbf{u}}\| \rightarrow 0$, the first order expansion above shall approximate the exact value fairly well.

In the next Proposition, we compute the gradient of the objective function. We introduce some extra notation to improve the legibility. Given the initial condition $\hat{x}_0 \in [0, 1]$ and the nominal threshold distribution $\bar{\mathbf{p}}$, we denote with \hat{x}_s the “uncontrolled” sequence $\hat{x}_{s+1} = \mathbf{f}_{d_G}(\hat{x}_s)^\top \bar{\mathbf{p}}$ for any $s \geq 0$. Let $\mathbf{f}'_{d_G}(x)$ be the vector valued function with the first derivatives of $\mathbf{f}_{d_G}(x)$, i.e. $(\mathbf{f}'_{d_G}(x))_r = f'_{d,r}(x)$ where

$$\begin{aligned} f'_{d,0}(x) &= f'_{d,d+1}(x) = 0 \quad \forall x \in [0, 1] \\ f'_{d,r}(x) &= \binom{d}{r} r x^{r-1} (1-x)^{d-r} \quad r \neq \{0, d+1\}. \end{aligned}$$

We define the coefficients

$$h_s(\hat{x}_0) := \prod_{i=1}^{s-1} (\mathbf{f}'_{d_G}(\hat{x}_i)^\top \bar{\mathbf{p}})$$

with the assumption that an empty product gives one.

Proposition 4.1. *Let $\hat{x}_0 \in [0, 1]$ and $\hat{T}_x \in \mathbb{N}$ and consider the objective function (4.26) where the state sequence $\hat{x}[\cdot]$ is explicitly substituted by the recursion (4.25). The first order Taylor expansion (4.28) has constant term*

$$\hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x} = c_x \sum_{s=1}^{\hat{T}_x-1} \hat{x}_s + c_T \hat{x}_{\hat{T}_x}$$

and gradient

$$\nabla \hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x} = \mathbf{c}_u - c_x \mathbf{g}_{d_G}(\hat{x}_0) \sum_{s=1}^{\hat{T}_x-1} h_s(\hat{x}_0) - c_T h_{\hat{T}_x}(\hat{x}_0) \mathbf{g}_{d_G}(\hat{x}_0).$$

Moreover, if $\hat{T}_x = 1$ the expansion is exact.

Proof. For $\hat{T}_x = 1$ it is enough to recognize the terms from (4.27). In the $\hat{T}_x > 1$ case first we expand $\hat{x}[s]$ to the first order in $\hat{\mathbf{u}}$ for any $s \in \{1, \dots, \hat{T}_x\}$, working on the recursive dynamic (4.25). The state $\hat{x}[1]$ is already linear in $\hat{\mathbf{u}}$. We rewrite the first line of (4.25) using the notation for the uncontrolled state sequence, i.e. $\hat{x}_1 = \mathbf{f}_{d_G}(\hat{x}_0)^\top \bar{\mathbf{p}}$ where $\hat{x}_0 = \hat{x}[0]$. We obtain

$$\hat{x}[1] = \hat{x}_1 - \mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}}. \quad (4.29)$$

For all the subsequent states we need the first order expansion of the update function $\mathbf{f}_{d_G}(x)^\top \bar{\mathbf{p}}$:

$$\mathbf{f}_{d_G}(x_0 + (x - x_0))^\top \bar{\mathbf{p}} = \mathbf{f}_{d_G}(x_0)^\top \bar{\mathbf{p}} + \mathbf{f}'_{d_G}(x_0)^\top \bar{\mathbf{p}}(x - x_0) + o(x - x_0)$$

Substituting (4.29) in the expression above we obtain the expansion of the state $\hat{x}[2]$

$$\begin{aligned} \hat{x}[2] &= \mathbf{f}_{d_G}(\hat{x}[1])^\top \bar{\mathbf{p}} \\ &= \mathbf{f}_{d_G}(\hat{x}_1 - \mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}})^\top \bar{\mathbf{p}} \\ &= \mathbf{f}_{d_G}(\hat{x}_1)^\top \bar{\mathbf{p}} - \mathbf{f}'_{d_G}(\hat{x}_1)^\top \bar{\mathbf{p}}(\mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}}) + o(\mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}}) \end{aligned}$$

We recognize $\hat{x}_2 = \mathbf{f}_{d_G}(\hat{x}_1)^\top \bar{\mathbf{p}}$ and $h_2(\hat{x}_0) = \mathbf{f}'_{d_G}(\hat{x}_1)^\top \bar{\mathbf{p}}$. For the remainder, using the Cauchy-Schwartz inequality and that $\mathbf{g}_{d_G}(\hat{x}_0) \in [0, 1]^{d_G+1}$ we have

$$|\mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}}| \leq \|\mathbf{g}_{d_G}(\hat{x}_0)\| \|\hat{\mathbf{u}}\| \leq (d_G + 1)\|\hat{\mathbf{u}}\|.$$

Therefore, the state $\hat{x}[2]$ has first order expansion

$$\hat{x}[2] = \hat{x}_2 - h_2(\hat{x}_0) \mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}} + o(\|\hat{\mathbf{u}}\|). \quad (4.30)$$

We further expand $\hat{x}[3]$ and use it to infer the general expansion expression. We plug the expression (4.30) of $\hat{x}[2]$ into $\mathbf{f}_{d_G}(x)^\top \bar{\mathbf{p}}$ to obtain

$$\begin{aligned} \hat{x}[3] &= \mathbf{f}_{d_G}(\hat{x}[2])^\top \bar{\mathbf{p}} \\ &= \mathbf{f}_{d_G}\left(\hat{x}_2 - \mathbf{f}'_{d_G}(\hat{x}_1)^\top \bar{\mathbf{p}}(\mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}}) + o(\|\hat{\mathbf{u}}\|)\right)^\top \bar{\mathbf{p}} \\ &= \mathbf{f}_{d_G}(\hat{x}_2)^\top \bar{\mathbf{p}} + \mathbf{f}'_{d_G}(\hat{x}_2)^\top \bar{\mathbf{p}}\left(-\mathbf{f}'_{d_G}(\hat{x}_1)^\top \bar{\mathbf{p}}(\mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}}) + o(\|\hat{\mathbf{u}}\|)\right) \\ &\quad + o\left(-\mathbf{f}'_{d_G}(\hat{x}_1)^\top \bar{\mathbf{p}}(\mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}}) + o(\|\hat{\mathbf{u}}\|)\right) \\ &= \hat{x}_3 - \mathbf{f}'_{d_G}(\hat{x}_2)^\top \bar{\mathbf{p}} \mathbf{f}'_{d_G}(\hat{x}_1)^\top \bar{\mathbf{p}}(\mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}}) + \mathbf{f}'_{d_G}(\hat{x}_2)^\top \bar{\mathbf{p}} o(\|\hat{\mathbf{u}}\|) \\ &\quad + \mathbf{f}'_{d_G}(\hat{x}_1)^\top \bar{\mathbf{p}} o(\mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}}). \end{aligned}$$

As before we recognize

$$\hat{x}[3] = \hat{x}_3 - h_3(\hat{x}_0) \mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}} + o(\|\hat{\mathbf{u}}\|).$$

Similarly the expansion of the generic $\hat{x}[s]$ is

$$\hat{x}[s] = \hat{x}_s - h_s(\hat{x}_0) \mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}} + o(\|\hat{\mathbf{u}}\|). \quad (4.31)$$

To obtain the Taylor series (4.28) we plug the expansions (4.31) in the objective function (4.26):

$$\begin{aligned} \hat{\Gamma}(\hat{\mathbf{u}})_{\hat{x}_0, \hat{T}_x} &= \mathbf{c}_u^\top \hat{\mathbf{u}} + c_x \sum_{s=1}^{\hat{T}_x-1} \hat{x}[s] + c_T \hat{x}[\hat{T}_x] \\ &= \mathbf{c}_u^\top \hat{\mathbf{u}} + c_x \sum_{s=1}^{\hat{T}_x-1} (\hat{x}_s - h_s(\hat{x}_0) \mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}}) \\ &\quad + c_T \hat{x}_{\hat{T}_x} - c_T h_{\hat{T}_x}(\hat{x}_0) \mathbf{g}_{d_G}(\hat{x}_0)^\top \hat{\mathbf{u}} + o(\|\hat{\mathbf{u}}\|) \\ &= \left[\mathbf{c}_u^\top - c_x \mathbf{g}_{d_G}(\hat{x}_0) \sum_{s=1}^{\hat{T}_x-1} h_s(\hat{x}_0) - c_T h_{\hat{T}_x}(\hat{x}_0) \mathbf{g}_{d_G}(\hat{x}_0) \right]^\top \hat{\mathbf{u}} \\ &\quad + c_x \sum_{s=1}^{\hat{T}_x-1} \hat{x}_s + c_T \hat{x}_{\hat{T}_x} + o(\|\hat{\mathbf{u}}\|) \end{aligned}$$

Grouping the terms concludes the proof. \square

The Proposition (4.1) identifies the coefficient of the affine approximation of $\hat{\Gamma}(\hat{\mathbf{u}})_{\hat{x}_0, \hat{T}_x}$. The linear optimization problem (4.24) with the differential $\nabla \hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x}^\top \hat{\mathbf{u}}$ as objective function is:

$$\begin{aligned} \min_{\hat{\mathbf{u}}} \quad & \nabla \hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x}^\top \hat{\mathbf{u}} \\ \text{s.t.} \quad & \mathbf{0} \preceq \hat{\mathbf{u}} \preceq J\bar{\mathbf{p}} \\ & \boldsymbol{\omega} \cdot \hat{\mathbf{u}} \leq \hat{b}. \end{aligned} \quad (4.32)$$

The problem above is enough simple to solve, that it is possible to build an optimal solution following an *ad-hoc* procedure. For any $\hat{b} \geq 0$ the problem is *feasible* i.e. the constraints $\mathbf{0} \preceq \hat{\mathbf{u}} \preceq J\bar{\mathbf{p}}$ and $\boldsymbol{\omega} \cdot \hat{\mathbf{u}} \leq \hat{b}$ have a non-empty common intersection. The optimal solution of the problem (4.32) is characterized by the following proposition, which we state using a generic notation.

Proposition 4.2. *Let $\mathbf{a}, \mathbf{c}, \bar{\mathbf{x}}$ be vectors in \mathbb{R}^n with $\mathbf{a} \succcurlyeq \mathbf{0}$ and $\bar{\mathbf{x}} \succcurlyeq \mathbf{0}$. Let $b > 0$ and consider the feasible linear program (LP)*

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^n} \quad & \mathbf{c}^\top \mathbf{x} \\ \text{s.t.} \quad & \mathbf{0} \preccurlyeq \mathbf{x} \preccurlyeq \bar{\mathbf{x}} \\ & \mathbf{a}^\top \mathbf{x} \leq b. \end{aligned} \tag{4.33}$$

Consider a point $\mathbf{x}^ \in \mathbb{R}^n$. For every $i \in \{j : c_j \geq 0\}$, let $x_i^* = 0$. For every $i \in \{j : c_j < 0 \text{ and } a_j = 0\}$, let $x_i^* = \bar{x}_i$. For every $i \in \{j : c_j < 0 \text{ and } a_j > 0\}$, let the components $x_i^* \geq 0$ be saturated² following the order given by the descending sequence of $\frac{-c_i}{a_i}$, until either $x_i^* = \bar{x}_i$ or $\sum_{i \in I} a_i x_i^* \leq b$. The point \mathbf{x}^* obtained is feasible and optimal.*

The proof of the proposition uses the following two lemmas.

Lemma 4.3. *Let $\mathbf{a}, \mathbf{c}, \bar{\mathbf{x}}$ be vectors in \mathbb{R}^n such that $\mathbf{a} \succcurlyeq \mathbf{0}$ and $\bar{\mathbf{x}} \succcurlyeq \mathbf{0}$, and let $b > 0$. Consider the feasible Linear Program (4.33), and let $\mathbf{x}^* \in \mathbb{R}^n$ be an optimal solution. If $c_i > 0$ then $x_i^* = 0$. If $c_i = 0$ and $x_i^* > 0$, then any $\hat{\mathbf{x}}$, such that $\hat{x}_j = x_j^*$ for $j \neq i$ and $0 \leq \hat{x}_i < x_i^*$, is also optimal.*

Proof. By absurd suppose that a feasible \mathbf{x}^* is optimal and $\exists i : c_i > 0$ and $x_i^* > 0$. We isolate the contribute of x_i^* in the objective function and in the constraint $\mathbf{a}^\top \mathbf{x} \leq b$:

$$\begin{aligned} \mathbf{c}^\top \mathbf{x}^* &= c_i x_i^* + \sum_{j \neq i} c_j x_j^* \\ \mathbf{a}^\top \mathbf{x}^* &= a_i x_i^* + \sum_{j \neq i} a_j x_j^*. \end{aligned}$$

Consider the point $\hat{\mathbf{x}}$ with $\hat{x}_i = 0$ while $\hat{x}_j = x_j^*$ for $j \neq i$. Since \mathbf{x}^* is feasible, $\hat{\mathbf{x}}$ is also feasible: the box constraints remain satisfied and $\mathbf{a}^\top \hat{\mathbf{x}} \leq \mathbf{a}^\top \mathbf{x}^* \leq b$. Moreover, $\mathbf{c}^\top \hat{\mathbf{x}} < \mathbf{c}^\top \mathbf{x}^*$: $\hat{\mathbf{x}}$ has an objective smaller than \mathbf{x}^* which cannot be optimal. Hence any optimal solution \mathbf{x}^* must have $x_i^* = 0$ if $c_i > 0$.

Similarly, suppose \mathbf{x}^* is optimal, with $x_i^* > 0$ for i for which $c_i = 0$. Then any $\hat{\mathbf{x}}$ such that $\hat{x}_j = x_j^*$ for $j \neq i$ and $0 \leq \hat{x}_i < x_i^*$ is also optimal because $\hat{\mathbf{x}}$ is feasible and the objective remains unchanged. \square

Lemma 4.4. *Let $\mathbf{a}, \mathbf{c}, \bar{\mathbf{x}}$ be vectors in \mathbb{R}^n such that $\mathbf{a} \succcurlyeq \mathbf{0}$, $\mathbf{c} \prec \mathbf{0}$ and $\bar{\mathbf{x}} \succcurlyeq \mathbf{0}$, and let $b > 0$. Consider the feasible Linear Program (4.33), and let $\mathbf{x}^* \in \mathbb{R}^n$ be an optimal solution. If $b \geq \mathbf{a}^\top \bar{\mathbf{x}}$ the optimal solution is $\mathbf{x}^* = \bar{\mathbf{x}}$. Else the optimal solution is such that $\mathbf{a}^\top \mathbf{x}^* = b$ and the components x_i^* are sequentially saturated according to the ranking given by the values $-c_i a_i^{-1}$ taken in decreasing order.*

²A variable is ‘‘saturated’’ if it takes the maximum value allowed by the constraints.

Proof. Since $a_i > 0$ for every i , we change the optimization variables and the sign of the objective function, to formulate an equivalent maximization problem. Consider the vectors \mathbf{y} , $\bar{\mathbf{y}}$ and $\hat{\mathbf{c}} \in \mathbb{R}^n$ such that, for every i , $y_i = a_i x_i$, $\bar{y}_i = a_i \bar{x}_i$ and $\hat{c}_i = -c_i a_i^{-1}$. The minimization problem above is equivalent to the maximization problem

$$\begin{aligned} \max_{\mathbf{y} \in \mathbb{R}^n} \quad & \hat{\mathbf{c}}^\top \mathbf{y} \\ \text{s.t.} \quad & \mathbf{0} \preceq \mathbf{y} \preceq \bar{\mathbf{y}} \\ & \mathbf{1}^\top \mathbf{y} \leq b. \end{aligned}$$

If $b \geq \mathbf{1}^\top \bar{\mathbf{y}}$, the point $\bar{\mathbf{y}}$ is feasible and, since $\hat{\mathbf{c}} \succ \mathbf{0}$, optimal.

If $b < \mathbf{1}^\top \bar{\mathbf{y}}$ an optimal solution \mathbf{y}^* is such that $\mathbf{1}^\top \mathbf{y}^* = b$. To see this suppose by absurd that $\mathbf{1}^\top \mathbf{y}^* < b$. Using $b < \mathbf{1}^\top \bar{\mathbf{y}}$ and $\mathbf{y}^* \preceq \bar{\mathbf{y}}$ there exists i such that $y_i^* < \bar{y}_i$. Hence, a new point $\tilde{\mathbf{y}} \preceq \bar{\mathbf{y}}$ with $\tilde{y}_j = y_j^*$ for any $j \neq i$ and $\tilde{y}_i = y_i^* + \min\{\bar{y}_i - y_i^*, b - \mathbf{1}^\top \mathbf{y}^*\} > y_i^*$ is feasible and has larger objective $\mathbf{c}^\top \tilde{\mathbf{y}} > \mathbf{c}^\top \mathbf{y}^*$ since $\mathbf{c} \succ \mathbf{0}$. Then \mathbf{y}^* cannot be optimal: absurd.

Now suppose by absurd that \mathbf{y}^* is optimal and there exists a pair of indexes i, j such that $\hat{c}_i > \hat{c}_j$, $y_i^* < \bar{y}_i$ and $y_j^* > 0$. Consider the quantity $\epsilon = \min\{\bar{y}_i - y_i^*, y_j^*\} > 0$ and a new vector $\tilde{\mathbf{y}}$ with components $\tilde{y}_k = y_k^*$ for any $k \neq i, j$, $\tilde{y}_i = y_i^* + \epsilon$ and $\tilde{y}_j = y_j^* - \epsilon$. The point $\tilde{\mathbf{y}}$ is feasible by construction with $\mathbf{1}^\top \tilde{\mathbf{y}} = \mathbf{1}^\top \mathbf{y}^* = b$. Since \mathbf{y}^* is optimal by hypothesis, it shall hold

$$\begin{aligned} \hat{\mathbf{c}}^\top \mathbf{y}^* &\geq \hat{\mathbf{c}}^\top \tilde{\mathbf{y}} \\ \sum_{k \neq i, j} \hat{c}_k y_k^* + \hat{c}_i y_i^* + \hat{c}_j y_j^* &\geq \sum_{k \neq i, j} \hat{c}_k \tilde{y}_k + \hat{c}_i \tilde{y}_i + \hat{c}_j \tilde{y}_j \\ \hat{c}_i y_i^* + \hat{c}_j y_j^* &\geq \hat{c}_i y_i^* + \hat{c}_i \epsilon + \hat{c}_j y_j^* - \hat{c}_j \epsilon \\ \hat{c}_j \epsilon &\geq \hat{c}_i \epsilon \end{aligned}$$

and since $\epsilon > 0$ we have $\hat{c}_i \leq \hat{c}_j$, absurd. Therefore, if \mathbf{y}^* is optimal for any pair i, j with $\hat{c}_i > \hat{c}_j$ the following must hold

$$\begin{cases} y_i^* < \bar{y}_i & \text{implies } y_j^* = 0 \\ y_j^* > 0 & \text{implies } y_i^* = \bar{y}_i \end{cases}$$

The proof is completed by going back to the original variables. □

Proof of Proposition 4.2. Apply Lemma 4.3 for every $i \in \{j : c_j \geq 0\}$. Each x_i^* with $i \in \{j : c_j < 0 \text{ and } a_j = 0\}$ does not interfere with the linear constraint $\mathbf{a}^\top \mathbf{x} \leq b$ but improves the objective, therefore we set $x_i^* = \bar{x}_i$. The remaining components are those with index in $\{j : c_j < 0 \text{ and } a_j > 0\}$. Apply Lemma 4.4 on the reduced problem with just those variables, and conclude. □

The Propositions (4.1) and (4.2) explain how the heuristic linearization method identifies the controls that shall be implemented to reduce the subroutine objective function. The key element is the gradient $\nabla\hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x}$ and the values of its components; we shall further discuss the role of the initial point \hat{x}_0 and of the lookahead parameter \hat{T}_x . We remind that for $T_x = 1$ there is no approximation because the objective function is already affine.

From Proposition (4.1), the gradient expression is

$$\begin{aligned} \nabla\hat{\Gamma}(\mathbf{0})_{\hat{x}_0, 1} &= \mathbf{c}_u - c_T \mathbf{g}_{d_G}(\hat{x}_0), \\ \nabla\hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x} &= \mathbf{c}_u - c_x \mathbf{g}_{d_G}(\hat{x}_0) \sum_{s=1}^{\hat{T}_x-1} h_s(\hat{x}_0) - c_T h_{\hat{T}_x}(\hat{x}_0) \mathbf{g}_{d_G}(\hat{x}_0) \\ &= \mathbf{c}_u - \left[c_x \sum_{s=1}^{\hat{T}_x-1} h_s(\hat{x}_0) + c_T h_{\hat{T}_x}(\hat{x}_0) \right] \mathbf{g}_{d_G}(\hat{x}_0). \end{aligned}$$

Each control variable \hat{u}_r produces a gross benefit on the objective function (the component r of the second term above) at a positive cost $(\mathbf{c}_u)_r$. The component r of the gradient $\nabla\hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x}$ is the net “benefit” coefficient associated to the control variable \hat{u}_r when \hat{u}_r tends to 0. Notice that the gross benefits depend on the initial point \hat{x}_0 and on the lookahead parameter \hat{T}_x .

The last of the expressions above clarifies and distinguishes the two roles. The scalar quantity included in the square brackets depends on the lookahead parameter \hat{T}_x and on \hat{x}_0 . As a rule of thumb, we would expect that the larger the lookahead, the larger the factor. This is trivially true if $c_x = c_T$. The factor is scalar and does not depend on the index r of the control.

The vector $\mathbf{g}_{d_G}(\hat{x}_0)$ depends on \hat{x}_0 only. The value of each component is $g_{d_G, r}(\hat{x}_0)$ and depends heavily on the index r . We briefly remind the definition of the family of functions $g_{d, r}(x)$, which is

$$g_{d, r}(x) = \binom{d}{r} x^r (1-x)^{d-r}.$$

For a given x , the largest component of $\mathbf{g}_d(x)$ is the one with

$$r \in \mathbb{N} \cap [(d+1)x - 1, (d+1)x].$$

Given the gradient $\nabla\hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x}$, we shall use the control variables \hat{u}_r for which the corresponding component of the gradient are negative and more “convenient”. Proposition (4.2) explains what “convenient” means if the weights ω_r of the budget bounds are not uniform.

If we assume $\boldsymbol{\omega}$ is uniform (proportional to $\mathbf{1}$), more convenient is simply “more negative”. Moreover if \mathbf{c}_u is uniform, it is easier to understand the different role of \hat{x}_0 and \hat{T}_x . The ranking of the components of the gradient, from the smallest (and potentially negative) to the largest, depends on the

initial state \hat{x}_0 of the subroutine. The lookahead parameter determines which components among the smallest are actually negative: the thumb rule says that \hat{T}_x should be large enough to allow enough “benefits” to build over time and overcome the control cost.

We shall remember all these observations when we compare the control sequence suggested by the heuristic linearization method with those computed by the MPC or with the optimal solution.

4.3 Numerical simulations

In this Section, we present four numerical examples regarding the case study described in the previous Section. The numerical optimization shall identify the optimal control sequence by solving problem (4.16) for a given set of parameter. To approximate the optimal solution with a smaller computational effort we proposed the sub-optimal Model Predictive Control strategy in Subsection (4.2.1). Further, in Subsection (4.2.2) we developed the Linearization Method heuristic that shall approximate the MPC problem and help to characterize the optimal solution.

The simplified case study still contains a discrete number of parameters which we recall briefly here. In all four examples, we set $d_G = 8$. The controlled dynamical system is studied in a time interval $[0, T]$ with given T . The state sequence is $x[\cdot]$, the control sequence is $\mathbf{u}[\cdot]$ and the initial condition $x[0]$. For each example we shall specify:

- the nominal threshold distribution $\bar{\mathbf{p}}$;
- the control’s upper bound $J\bar{\mathbf{p}}$ of the box constraints $\mathbf{0} \preceq \mathbf{u}[t] \preceq J\bar{\mathbf{p}}$;
- the weights $\boldsymbol{\omega}$ and the sequence $b[t]$ of the budget constraints $\boldsymbol{\omega}^\top \mathbf{u}[t] \leq b[t]$.

In all the examples we use $\boldsymbol{\omega} = \mathbf{1}$ and set $b[t] = b$ for every t , with b to be specified.

Let the shorthand $\psi(x)$ denote the state equation of the uncontrolled LMF dynamic:

$$\psi(x) = \mathbf{f}_{d_G}(x)^\top \bar{\mathbf{p}}.$$

We define the LMF dynamical system “with maximally feasible control” as the system which chooses, for a given state $x[t]$, the feasible controls able to produce the smallest possible new state $x[t+1]$. The state function of such a system is

$$\check{\psi}(x) = \min_{\mathbf{u} \in \mathcal{U}} \{ \mathbf{f}_{d_G}(x)^\top \bar{\mathbf{p}} - \mathbf{g}_{d_G}(x)^\top \mathbf{u} \}$$

where \mathcal{U} is the set of feasible control vectors \mathbf{u} ,

$$\mathcal{U} = \{\mathbf{u} : \mathbf{0} \preceq \mathbf{u} \preceq J\bar{\mathbf{p}}, \boldsymbol{\omega}^\top \mathbf{u} \leq b\}.$$

Since $\mathbf{u} \succeq \mathbf{0}$ and for any $x \in [0, 1]$, $\mathbf{g}_{d_G}(x) \succeq \mathbf{0}$ and we observe that

$$\mathbf{f}_{d_G}(x)^\top \bar{\mathbf{p}} - \mathbf{g}_{d_G}(x)^\top \mathbf{u} \leq \psi(x)$$

for any $\mathbf{u} \in \mathcal{U}$ and for every $x \in [0, 1]$. As expected, for any control vector \mathbf{u} and any x the controlled state function is smaller than the uncontrolled $\psi(x)$. In other words, the area between the two curves $\check{\psi}(x) \leq \psi(x)$ contains the envelope of every feasible, controlled state function.

The linear objective function of the optimal control problem requires the control cost coefficients \mathbf{c}_u , the state cost coefficient c_x and the terminal cost coefficient c_T . In all the examples we use $\mathbf{c}_u = \mathbf{1}$. Once all the parameters are set, the optimization problem (4.16) of the case study is completely defined.

The two sub-optimal approaches to the control problem need three further parameters. For the MPC we shall specify the prediction horizon T_x and the control horizon T_u . The heuristic Linearization Method only needs the lookahead parameter T_x to which we assign the same value of the prediction horizon. Except for one case (Example 4, scenario A) these parameters are set to $T_x = 12$ and $T_u = 4$.

We stress that the examples presented are not exhaustive of the possible scenarios. They have been selected to provide interesting insights about the case study described in the previous Section and the corresponding optimization problem (4.16). The optimal solution will always be a control sequence able to reduce the state. In all the examples we will describe the optimal solution found and compare it with the control sequences identified by the MPC and constructed with the linearization heuristic. The first example has mainly the didactical purpose to introduce the reader to the graphical representation of the results. In the second and third examples we set a stringent budget bound, to force the optimization to choose fewer controls and thus simplify the analysis of the control pattern. In all the first three examples the MPC approach reproduces the optimal solution. Instead, the heuristic fails in the third example.

Finally, the fourth example is planned to test the reliability of the sub-optimal MPC and of the heuristic Linearization Method. It contains 3 scenarios where the cost coefficients c_x and c_T are progressively reduced (also, in the first scenario $T_x = 8$ instead of 12).

4.3.1 Example 1

We use the first example to introduces the format of the figures and tables used to visualize and report the problem information and results. In all the following examples we will adopt the same style.

In this example the time interval of interest lasts until time $T = 20$. The components of the nominal threshold distribution $\bar{\mathbf{p}}$ and of the upper bounds $J\bar{\mathbf{p}}$ on the control variables are in Table 4.1. Regarding the budget bound we fix $b = 0.300$ and observe that the budget constraint is not trivial since $\omega^\top J\bar{\mathbf{p}} = 0.750 > b = 0.300$.

r	0	1	2	3	4	5	6	7	8	9
\bar{p}_r	0	0.026	0.211	0.211	0.211	0.211	0.053	0.053	0.026	0
$(J\bar{\mathbf{p}})_r$	0	0.020	0.158	0.158	0.158	0.158	0.039	0.039	0.020	

Table 4.1: The nominal threshold distribution $\bar{\mathbf{p}}$ and the control's upper bounds $J\bar{\mathbf{p}}$ of the Example 1.

The above information specify the controlled LMF dynamical system together with its constraints. Using $d_G = 8$ and the nominal threshold distribution we easily obtain the uncontrolled state equation $\psi(x)$; using the bounds we can compute the state equation “with maximal feasible control” $\check{\psi}(x)$. This two state functions are represented in the main plot area of Figure 4.1, with the blue solid line and the red solid line respectively. The area between the two curves, shaded in magenta, contains all the feasible update laws: for any pair $(x[t], x[t+1])$ in the shaded area there exist a feasible control vector \mathbf{u} such that $x[t+1] = \mathbf{f}_{d_G}(x[t])^\top \bar{\mathbf{p}} - \mathbf{g}_{d_G}(x)^\top \mathbf{u}$.

The bottom area of Figure 4.1 represents the segments in $[0, 1]$ where the state dynamic is monotone regardless of the control vector \mathbf{u} chosen. The intervals where $\check{\psi}(x) \leq \psi(x) < x$ are colored in light blue and contain a left arrow: the dynamic there is always decreasing. For the intervals where the dynamic is always increasing, namely where $x < \check{\psi}(x) \leq \psi(x)$, we use light red and draw a right arrow. We call these segments *monotonous intervals*: in this example they are $(0, 0.240)$ (decreasing) and $(0.386, 0.968)$ (increasing).

In the intervals where $\check{\psi}(x) < x \leq \psi(x)$ the state variable can either increase or decrease, depending on the control variables chosen. We color such intervals with light magenta and (if space permits) we draw a pair of left/right arrow. In this example there are two of these interval: $(0.240, 0.386)$ and $(0.968, 1.000)$. Note that they correspond to the intervals where the 45° line intersects the area shaded in magenta.

The initial condition of this example is set to $x_0 = 0.350$, that falls in the central, non monotone interval. If the system is not controlled, the state

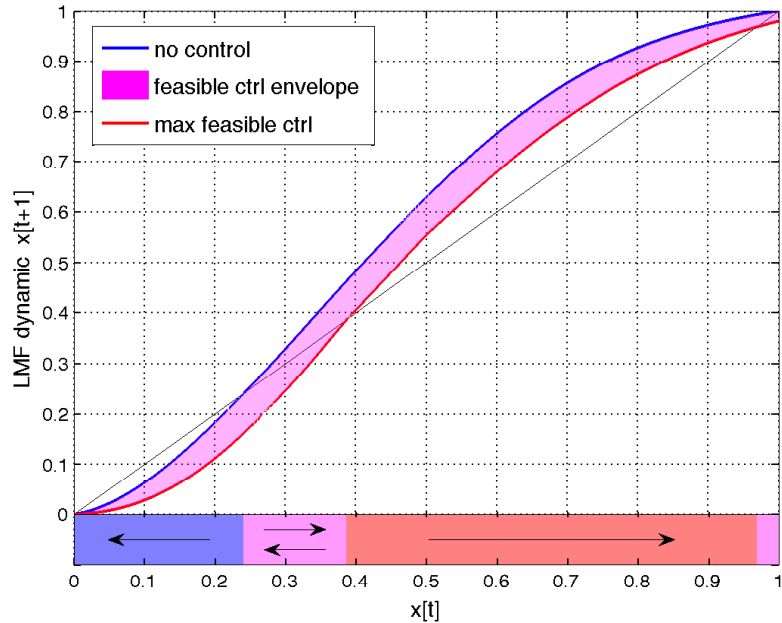


Figure 4.1: Envelope of the controlled state functions of Example 1.

will soon reach one, as shown in Figure 4.2 by the blue solid line. On the other hand, there is a feasible control sequence able to decrease the state: the red solid line represent the trajectory obtained with the maximally feasible controlled dynamic $\check{\psi}$. In order to reduce the activation, it is sufficient that the control sequence drives the state the the monotonous decreasing interval: from there on the system will automatically drag the state close to zero.

The values of the cost coefficients \mathbf{c}_u , c_x and c_T which define the optimal control problem are given in Table 4.2. The first part of the same table also contains the time horizon parameters of the sub-optimal solution strategies.

The second part of Table 4.2 contains the values of the objective function obtained by the numerical optimization with the difference approaches. For reference, we include the objective function's figure corresponding to the uncontrolled dynamic given by $\psi(x)$, which clearly is an upper bound of the optimal objective. We estimate the value of the objective function corresponding to the maximally feasible control policy, without knowing that control sequence. The lower bound neglects the control cost (i.e. we use $\mathbf{c}_u = \mathbf{0}$) and therefore it also is a lower bound for the optimal objective; since $\mathbf{c}_u = \omega$ the upper bound is obtained with $\mathbf{c}_u^\top \mathbf{u}[t] = \omega^\top \mathbf{u}[t] \leq b$. As a sanity check, observe that the optimal objective is between the lower bound of the maximally controlled dynamic and the objective of the uncontrolled dynamic (see Table 4.2). Moreover, the sub-optimal approaches have a larger or equal objective.

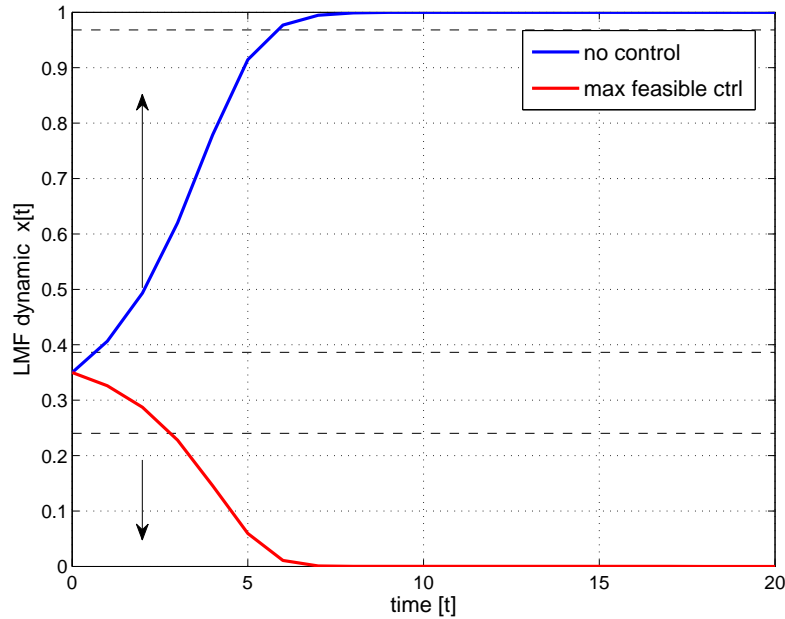


Figure 4.2: The uncontrolled and “maximally controlled” trajectories of Example 1, starting from $x_0 = 0.350$.

We proceed with the description of the numerical results obtained in this example, graphically represented in Figure 4.3. We will first describe the optimal solution and then compare the sub-optimal approaches. In both plots we use the same marker conventions: quantities regarding the optimal solution are in black circles, for the MPC we use blue diamonds and for the heuristic magenta squares.

The left plot of Figure 4.3 contains the state trajectories, with the state $x[t]$ plotted against the time t . As a reference we draw the uncontrolled and maximally controlled state trajectories, and the monotonicity intervals, like in Figure 4.2. The optimal trajectory in black line and circles (hidden beneath the MPC’s one) clearly shows that the optimization problem is solved by a control sequence that drives the state toward zero. The optimal state dynamic is similar to the maximally controlled dynamic: they coincide except for a little difference in the time steps $\{4, 5, 6, 7, 8, 9\}$. The MPC state trajectory (in blue line and diamonds) is identical to the optimal one; the heuristic (in magenta line and squares) is very close to both.

The right plot of Figure 4.3 represents graphically the elements $u_r[t]$ of the control sequence $\mathbf{u}[t]$ that are larger than zero, to show which control variable has been “used”. On the horizontal axis there is time; on the vertical axis there is the threshold index r from 0 to $d_G = 8$ to which correspond a sequence $u_r[t]$.

Example 1	
$(\mathbf{c}_u)_r$	$\forall r \quad 1$
c_x	0.700
c_T	4.000
T_x	12
T_u	4
Γ no control	16.029
Γ maximally feasible ctrl	≤ 6.741 ≥ 0.741
Γ Global optimization	2.036
Γ MPC optimization	2.036
Γ Linearization m. opt.	2.066

Table 4.2: Optimal control problem coefficient, parameter, and objective function results of Example 1.

Actually for each r there are three parallel sequences of markers, one for the sequence $u_r[t]$ of the optimal solution, one for the sequence $u_r[t]$ found by the MPC and one for that of the heuristic linearization method. Marker and colors are the same used in the left plot. We use a full marker for an element $u_r[t]$ whose value is close to “full scale”, i.e. more than 95% of the corresponding upper bound $(J\bar{\mathbf{p}})_r$. An empty marker represents a control variables only used partially (at least 5% of the upper bound).

In this example the control patterns of the optimal solution and MPC coincide, the heuristic differs in just two positions. The optimal solution uses selectively at most two control at a time and we register that the budget bound is an active constraint up to $t = 2$. After $t = 4$ it is not necessary to activate any control: the system is already in the monotone decreasing interval that will bring the state to zero. Looking at the actual values, the MPC control sequence presents the same figures of the optimal solution. The heuristic as well except for the two elements already mentioned.

To analyze the optimal solution and understand why the sub-optimal MPC and heuristic agree or disagree we shall remember the problem formulation. The optimal solution solves the problem (4.16) which is the global open-loop optimization over all the time interval $[0, T]$ by selecting T control vectors and “paying” the terminal cost at time T .

The MPC approach solves a sequence of smaller optimization problems. At time t the MPC shall optimize the dynamic in the interval $[t, t+T_x]$ choosing T_u control elements and paying the terminal cost at time $t+T_x$. Then only the first control vector $\mathbf{u}[t]$ is actually implemented while the other are discarded and

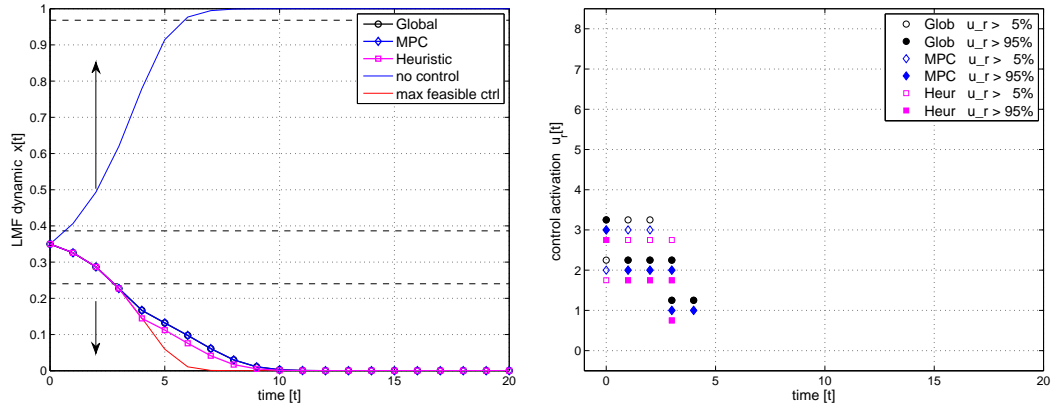


Figure 4.3: Optimization results of Example 1. Left: plot of the trajectories $x[\cdot]$. Right: a graphical representation of the control sequence $\mathbf{u}[t]$.

the time counter advances of one unit. Compared to the global problem, the MPC has a shorter prediction horizon and can coordinate up to T_u consecutive control vectors. The MPC approach may fail if it is necessary a longer “sight”, i.e. to coordinate the controls of further consecutive steps in order to improve the solution.

In the linearization heuristic the MPC limitation are further highlighted. The heuristic Linearization Method tries to reproduce the MPC approach by linearizing the objective function. The linearization has the “side effect” to break down any temporal coordination: any nonlinear benefit of coupling the control planning at subsequent steps is neglected. This would also happen in the MPC if we set $T_u = 1$ and in fact the heuristic substantially approximates this case. The linearization of the objective function around $\mathbf{u}[t] = \mathbf{0}$ also decouples the choice of the control variables with different r . Hence the linearization method ranks the control elements $u_r[t]$ according to the immediate benefit on the objective function neglecting any coordination between control elements at the same time or at subsequent time steps.

The benefits on the objective function are computed with a linear approximation of the dynamic for T_x steps. For large lookahead T_x the non-linearities build up and the approximation may fail or become even counterproductive.

The optimal solution of the problem is to drive the state toward zero. In this example, it suffices to enter the monotonous decreasing interval and the result can be obtained at no control cost. That is in fact what does the optimal solution. At $t = 3$ the state is already in the monotonicity interval but, according to the optimal solution, it is still convenient to activate some control elements at $t = 3$ and $t = 4$, to reduce the state more rapidly. From $t = 5$ no control

needs to be activate: the dynamic still decrease and reaches zero, though less quickly that the maximally controlled dynamic.

The MPC reproduces exactly the optimal solution: the control variables, the state dynamic and the objective function coincide up to the third decimal digit. The heuristic linearization method is very close to the MPC solution and hence to the optimal solutions too. From the control pattern, we can observe a little, but significative difference. Compared to the MPC and optimal control sequence, the heuristic activates partially the control $u_3[3]$ and not the control $u_1[4]$. Knowing how the heuristic “thinks” we can explain this with the lack of coordination between subsequent times. The heuristic judged convenient to activate $u_3[3]$ with the budged left from $u_1[3]$ and $u_2[3]$ while the MPC knew that $u_1[4]$ was slightly better, thanks to the longer control horizon.

4.3.2 Example 2

The nominal threshold distribution, the box constraints and the budget bound of this second example have been chosen to produce a dynamical system with a “slow” behavior. We study the process on the interval $[0, T]$ with $T = 40$. Table 4.3 collects the nominal threshold distribution $\bar{\mathbf{p}}$ and of the control’s upper bound $J\bar{\mathbf{p}}$. Activating all the controls of time t would cost $\boldsymbol{\omega}^\top J\bar{\mathbf{p}} = 0.760$. To simplify the analysis of the simulations we want to force the optimized dynamic to use only the most convenient control variables thus we fix a stringent budget bound with $b = 0.170$.

r	0	1	2	3	4	5	6	7	8	9
\bar{p}_r	0.004	0.077	0.230	0.115	0.115	0.115	0.115	0.115	0.115	0
$(J\bar{\mathbf{p}})_r$	0.002	0.034	0.207	0.103	0.103	0.103	0.103	0.052	0.052	

Table 4.3: The nominal threshold distribution $\bar{\mathbf{p}}$ and the control’s upper bounds $J\bar{\mathbf{p}}$ for the Example 2.

Figure 4.4 represents the envelope of the feasible controlled state functions corresponding to $\bar{\mathbf{p}}$ and the constraints given. There is a monotonous interval for $(0.011, 0.131)$ with decreasing dynamic, whereas in the interval $(0.131, 1)$ the state can increase or decrease. The initial condition is $x_0 = 0.500$ and falls in the non monotone interval. Without control the state sequence will slowly reach one, as shown by the blue line in Figure 4.5. The dynamic with maximally feasible control (red line) tends to zeros, with an initial slow decrease. In the interval around $[0.350, 0, 600]$ the quantity $x - \check{\psi}(x)$ is positive but quite small then any feasible decreasing dynamic would be very slow.

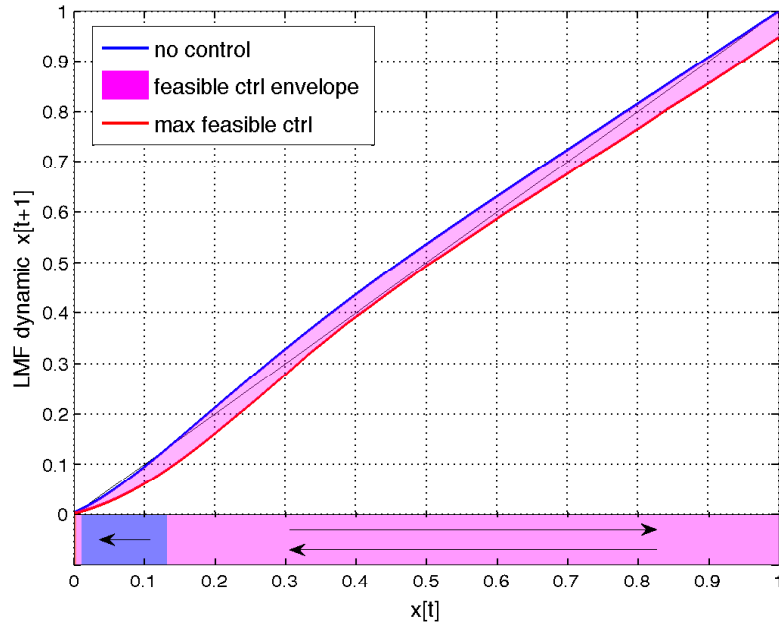


Figure 4.4: Envelope of the controlled state functions of Example 2.

The cost coefficients and \mathbf{c}_u , c_x and c_T and the parameters T_u and T_x of the sub-optimal strategies take the same values of the previous example. They are collected for reference in Table 4.4 together with the values of the objective functions computed in this example.

The optimal solution remains to drive the state to zero. It is drawn in black in the left plot of Figure 4.6 but it is hidden by the other curves. The optimal state trajectory lies against the “maximally controlled” trajectory until $t = 30$. This implies that until $t = 29$ the budget bound is an active constraints as verified with a numerical check. In the right plot of the Figure we see for example that $u_2[25]$ is not completely used otherwise the budget bound would be exceeded ($\omega_2(J\bar{\mathbf{p}})_2 = 0.207 > b = 0.170$). At $t = 29$ the state enters the monotonous interval with decreasing dynamic. From $t = 30$ we have $u_2[t] = 0$ even if the budget bound is not saturated ($\omega_0\bar{p}_0 + \omega_1\bar{p}_1 = 0.036 < b = 0.170$). On one hand the control are not convenient and it is better to save on the control costs; on the other hand the state has enough time to decrease to zero before time T , when the terminating cost is payed.

The MPC and the heuristic identify the optimal solution up to negligible differences, as we may notice from the objective function values in Table 4.4. From the optimal solution we know that it is better to reduce the state and reach zero. Notice that at least in the first steps the suboptimal strategies are insensible to the presence of the monotonicity interval due to their limited

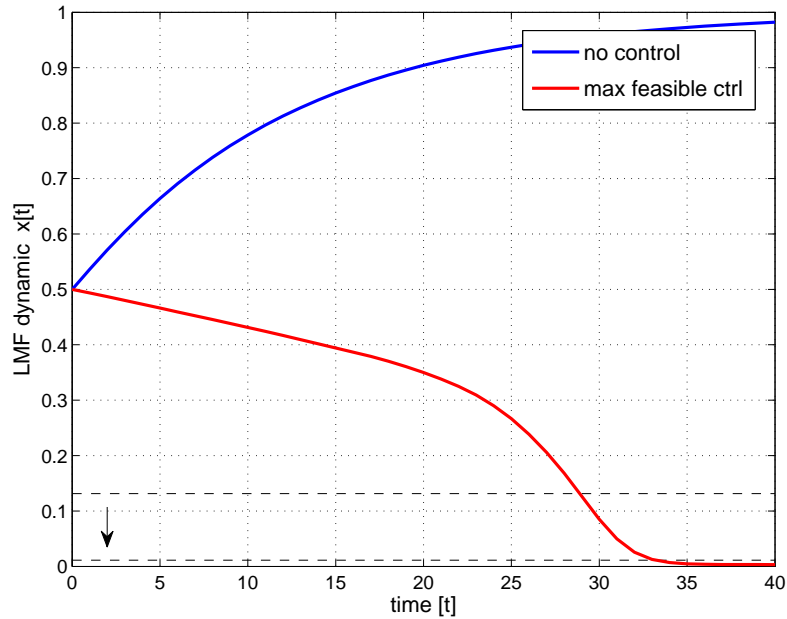


Figure 4.5: The uncontrolled and “maximally controlled” trajectories of Example 2, starting from $x_0 = 0.500$.

prediction horizon. Moreover, at time t the sequential optimization in these strategies “imagines” to pay the terminal cost at $t + T_x$. In the first time steps the suboptimal strategies cannot guess that by controlling completely the system the state will reach zero and that the terminal cost will be negligible. However both suboptimal approaches do that: they manage to perceive an advantage in controlling the dynamic and we conclude that this advantage is not small.

This example features a very limited budgeted bound to force the choice of the most efficient control variables. At time t the controls elements $u_r[t]$ are active only for one or two values of r which should be those that make the state decrease the most. The heuristic chooses the control following the myopic ranking of the partial derivatives of the objective function because $\omega = \mathbf{1}$. Since in this example the heuristic represents well the optimal solution, the linear approximation has to be consistent: the optimal solution pattern shall be related to the same ranking. Comparing the left and right plot of Figure 4.6, the control pattern follows the decreasing behavior of the state sequence: the control $u_r[t]$ that are active at time t are those with $r/d_G \approx x[t]$. This is not casual.

Example 1	
$(\mathbf{c}_u)_r$	$\forall r \quad 1$
c_x	0.700
c_T	4.000
T_x	12
T_u	4
Γ no control	27.297
Γ maximally feasible ctrl	≤ 14.503 ≥ 7.703
Γ Global optimization	13.017
Γ MPC optimization	13.018
Γ Linearization m. opt.	13.019

Table 4.4: Optimal control problem coefficient, parameter, and objective function results of Example 2.

The controlled LMF dynamical system has state equation

$$x[t+1] = \mathbf{f}_{d_G}(x[t])^\top \bar{\mathbf{p}} - \mathbf{g}_{d_G}(x[t])^\top \mathbf{u}[t].$$

Given $x[t]$, the largest component of $\mathbf{g}_{d_G}(x[t])$ is the one with r in the interval $[(d+1)x[t] - 1, (d+1)x[t]]$, which means roughly $r/(d_G+1) \approx x[t]$. Given the state $x[t]$, the control $u_r[t]$ that contribute the most to the decrement of $x[t+1]$ is the one such that r maximizes $g_{d,r}(x[t])$. The example has been built to stress this aspect.

4.3.3 Example 3

In the third example the heuristic fails to approximate the MPC solution and hence the optimal solution. With respect to the previous example, we slightly modified the nominal threshold distribution and the box constraints to obtain an interval with monotonous increasing state dynamic, see Table 4.5 and Figure 4.7. The time interval remains $[0, T]$ with $T = 40$ and we maintain the tight budget bound with $b = 0.170$. The cost coefficients c_x and c_T (see Table 4.6) have been reduced. The control cost coefficient \mathbf{c}_u and the parameters T_u and T_x remains the same.

Figure 4.7 shows the envelope of the state functions for the controlled dynamical systems. There is a prominent monotonicity interval with increasing dynamic in $(0.364, 0.567)$, which serves as a sort of “no return” interval. The state can be increased or decreased in the intervals $(0.021, 0.364)$ and $(0.567, 1)$. The initial condition is set to $x_0 = 0.350$, which is just before the monotone

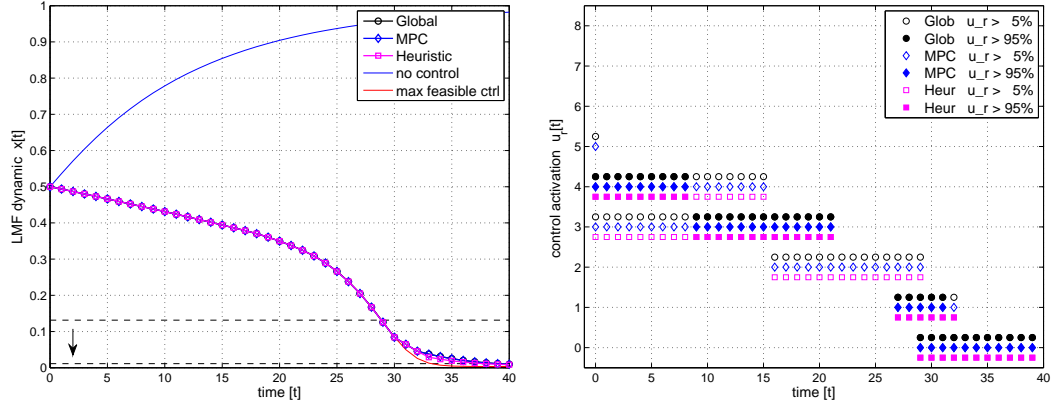


Figure 4.6: Optimization results of Example 2. Left: plot of the trajectories $x[\cdot]$. Right: a graphical representation of the control sequence $\mathbf{u}[t]$.

r	0	1	2	3	4	5	6	7	8	9
\bar{p}_r	0.023	0.057	0.241	0.115	0.115	0.115	0.115	0.115	0.103	0
$(J\bar{\mathbf{p}})_r$	0.010	0.026	0.217	0.103	0.103	0.103	0.103	0.052	0.047	

Table 4.5: The nominal threshold distribution $\bar{\mathbf{p}}$ and the control's upper bounds $J\bar{\mathbf{p}}$ for the Example 3.

Example 1	
$(\mathbf{c}_u)_r$	$\forall r \quad 1$
c_x	0.250
c_T	2.500
T_x	12
T_u	4
Γ no control	10.890
Γ maximally feasible ctrl	≤ 8.421 ≥ 1.621
Γ Global optimization	5.685
Γ MPC optimization	5.693
Γ Linearization m. opt.	10.871

Table 4.6: Optimal control problem coefficient, parameter, and objective function results of Example 3.

increasing interval, without control the state increases and slowly reaches one. If the state increases a little from its initial condition and enters in 0.364 the

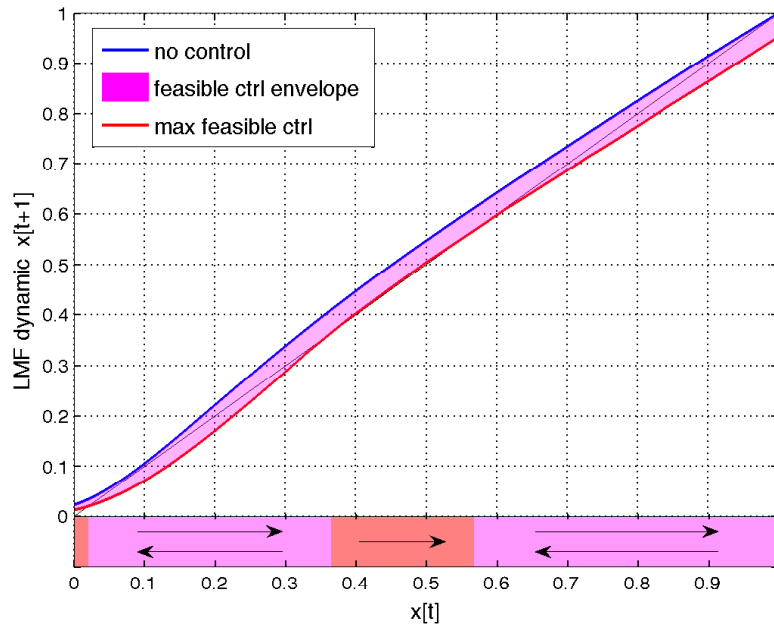


Figure 4.7: Envelope of the controlled state functions of Example 3.

monotonous interval, it will cross it and will never return to values smaller than 0.567. Controlling the dynamic from the initial condition still allows to drive the state close to zero.

The optimal solution is to reduce the state which levels to a limit value that is not the smallest reachable. The behavior of the optimal solution, state trajectories and control pattern, are shown in Figure 4.8. The MPC identifies the right solution, similar to what it did in the Example 2.

The heuristic linearization method fails to approximate the MPC solution. The “mistake” happens at the first step: the heuristic gives $x[1] > x[0]$ while the MPC obtains $x[1] < x[0]$, see the left plot of Figure 4.8. Moreover, the state $x[1]$ obtained by the heuristic falls inside the monotonous increasing interval: the heuristic has no chance to recover the mistake. The control patterns on the right of Figure 4.8 show that $u_2[0]$ is active for the the MPC and optimal solution but not for the heuristic. The heuristic does not judge $u_2[0]$ convenient to reduce the objective function. There are two possible reasons for this, related to the linearization: the use of $u_2[0]$ may be useful only if coordinated with controls at later time; the linearized subroutine objective function at time $t = 0$ may be inaccurate *per se*.

The second issue is indeed true and can be verified easily. We remind that the subroutine objective function comes from a gradient. We shall compare it with the corresponding finite difference quotient for various increments. For

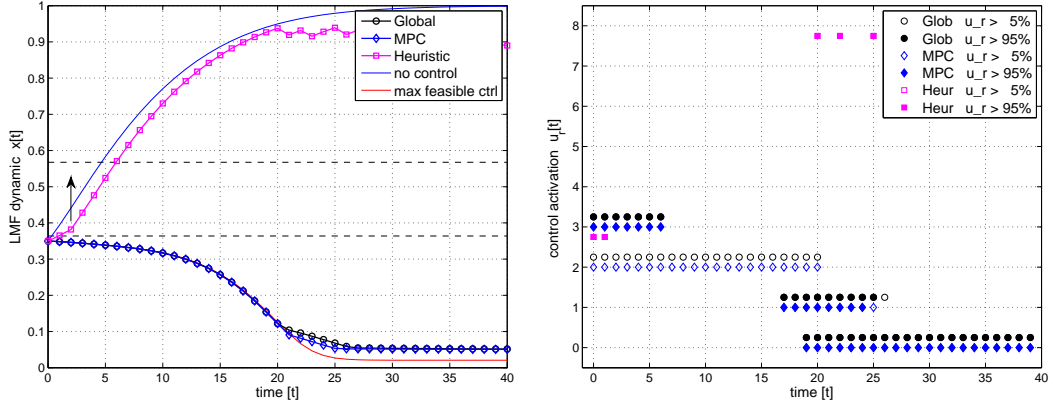


Figure 4.8: Optimization results of Example 3. Left: plot of the trajectories $x[\cdot]$. Right: graphical representation of the control sequence $\mathbf{u}[t]$.

each r let introduce the difference quotients

$$\Delta \hat{\Gamma}_{z,r} = \frac{\hat{\Gamma}(z \bar{p}_r \mathbf{e}_r)_{\hat{x}_0, \hat{T}_x} - \hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x}}{z \bar{p}_r}.$$

where z is the increment coefficient and \mathbf{e}_r is a vector of the canonical basis. In Table 4.7 we compare the gradient with the difference quotients for $z = 5\%$, $z = 50\%$ and $z = 95\%$. For $r = 2, 3, 4$ there are some differences, symptom of non-linearities. The case of $r = 2$ is delicate: the component of the gradient is positive, while the difference quotient for $z = 50\%$ and $z = 95\%$ is negative. Assuming the control variables are decoupled, the heuristic cannot judge useful the control $u_2[0]$ from the gradient value.

The mistake done by the heuristic is related to the values of the cost coefficient c_x and c_T that have been reduced with respect to the previous example. Each component of the gradient $\nabla \hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x}$, computed in Proposition 4.1, can be expressed as

$$\left(\nabla \hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x} \right)_r = (\mathbf{c}_u)_r - c_x A - c_T B$$

where A, B are factor that depend on r , $\hat{x}[0]$ and \hat{T}_x . The reduction of c_x and c_T reduces the margin of the negative components of the gradient and makes the approximation more sensible to the presence of non-linearities.

4.3.4 Example 4

In the examples considered so far the solution found by the MPC has been consistent with the optimal solution. In this last example we force the MPC

r	$\left(\nabla\hat{\Gamma}(\mathbf{0})_{\hat{x}_0, \hat{T}_x}\right)_r$	$H_{0.05,r}$	$H_{0.50,r}$	$H_{0.95,r}$
0	0.881	0.881	0.881	0.881
1	0.488	0.488	0.487	0.486
2	0.035	0.031	-0.006	-0.049
3	-0.039	-0.041	-0.061	-0.082
4	0.300	0.300	0.291	0.282
5	0.699	0.698	0.697	0.695
6	0.919	0.919	0.919	0.919
7	0.988	0.988	0.988	0.988
8	0.999	0.999	0.999	0.999

Table 4.7: Example 3, comparison of the heuristic objective function gradient at $t = 0$ with the difference quotients for 3 values of the relative increment z .

to find a different solution. We consider three 3 sub-cases where we gradually reduce the cost coefficients. The optimal solution remains to drive the state toward no activation in all the three cases. The heuristic fails in the second sub-case and completely miss the MPC solution. In the third sub-case the MPC miss the global solution.

Table 4.8 contains the nominal threshold distribution $\bar{\mathbf{p}}$ and the control’s upper bound $J\bar{\mathbf{p}}$. The controls u_0 and u_{d_G} are not available in this example while the controls u_1, u_2 and u_3 have a small upper bound. The remaining control variables are upper bounded only by the structure of the problem. The time interval of interest is $[0, T]$ with $T = 40$. Regarding the budget bounds $\boldsymbol{\omega}^\top \mathbf{u}[t] \leq b$ we fix $\boldsymbol{\omega} = \mathbf{1}$ and $b = 0.500$. Note that $\boldsymbol{\omega}^\top J\bar{\mathbf{p}} = 0.6620$: the budget bound is not trivial but not very restrictive either.

r	0	1	2	3	4	5	6	7	8	9
\bar{p}_r	0.007	0.033	0.293	0.133	0.133	0.133	0.133	0.133	0	0
$(J\bar{\mathbf{p}})_r$	0	0.003	0.059	0.067	0.133	0.133	0.133	0.133	0	

Table 4.8: The nominal threshold distribution $\bar{\mathbf{p}}$ and the control’s upper bounds $J\bar{\mathbf{p}}$ for the Example 4.

The Figure 4.9 represents the LMF dynamical systems corresponding to the nominal threshold distribution and to the control constraints of this example. There is a “monotonous” interval with decreasing state dynamic for approximately $(0.010, 0.154)$. In the interval $(0.154, 1)$ the state can increase or decrease. However, the threshold distribution and control constraints make the curve $\check{\psi}(x)$ close to the 45° line: it means that it is not an easy task to decrease

the state.

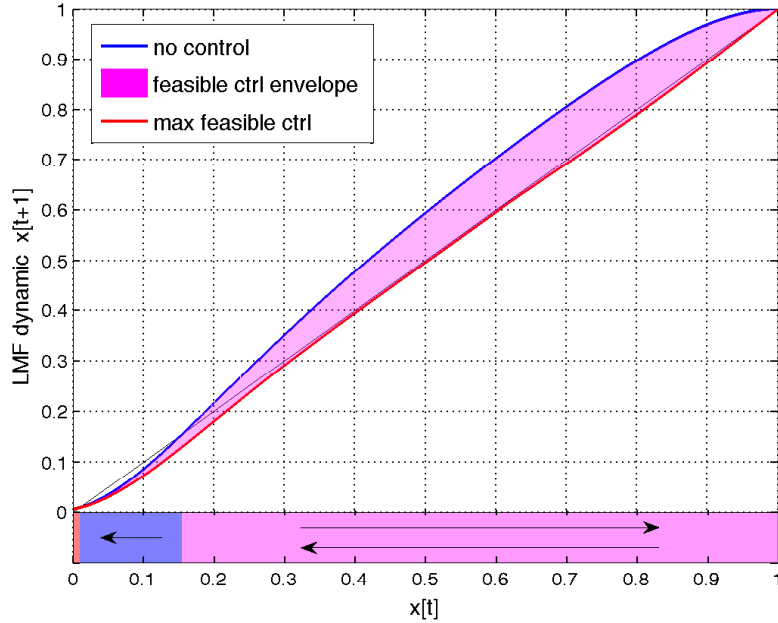


Figure 4.9: Envelope of the controlled state functions of Example 4.

We set the initial condition to $x_0 = 0.350$. If the system is not controlled, the activation fraction will soon reach one, as represented in Figure 4.10 by the solid blue line. On the other hand, the state dynamic corresponding to the maximally feasible controlled LMF system of law $\check{\psi}(x)$ is decreasing and reaches zero. Notice that the controlled trajectory shall just enter the interval with monotone decreasing dynamic to further decrease to zero.

The three sub-cases of this Example are called *A*, *B* and *C*. In all of them we use the same control cost coefficient \mathbf{c}_u whereas the state cost coefficients c_x and c_T are progressively reduced. Regarding the parameters of the sub-optimal approaches, we set the MPC's control horizon T_u to 4 in all the three case. The MPC's prediction horizon coincides with the lookahead parameter of the Linearization method: T_x is set to 8 in *A* and to 12 in *B* and *C*. The objective function cost coefficients and the parameters for the three sub-cases are collected in the first part of Table 4.9. The second part of the table contains the optimal values and the estimates of the objective functions.

In the following we describe the results obtained for the three sub-cases using the Figures 4.11, 4.12 and 4.13. In each figure, the left plot contains the controlled state dynamic while the right plot is the usual graphical representation of the control patterns. The optimal solution is always to use the control variables in order to drive the state dynamic to zero. The state trajectories

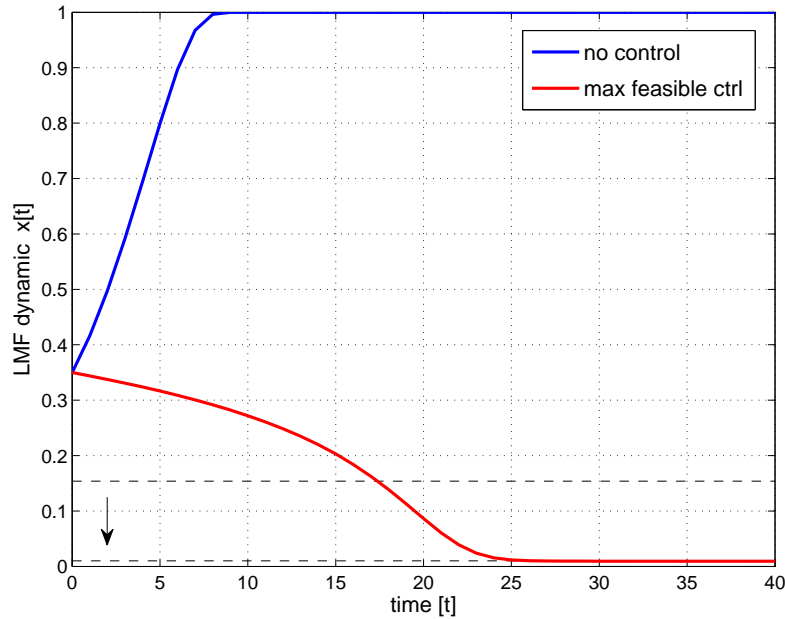


Figure 4.10: The uncontrolled and “maximally controlled” trajectories of Example 4, starting from $x_0 = 0.350$.

of the optimal solutions are close to the trajectory of the maximally feasible controlled system in all the three sub-cases. Only in the interval $[17, 26]$ there is a little difference: for those steps the state dynamic is about to enter the monotonous decreasing interval. In all the three sub-cases the optimal solutions use most of the available control, from the beginning up to time $t = 12$; then the control action are gradually reduced and limited to the controls with smaller r . With the exception of the last step, in scenario *A* and *B* no control are active from $t = 24$. In the last scenario, u_1 remains partially active until the end.

Analyzing in more detail the case *A*, we observe that the MPC is in good agreement with the optimal solution. We can see this from the state dynamic and controls adopted, in Figure 4.11, and comparing the objective function values in Table 4.9. The Linearization method reproduces the MPC solution fairly well. We note that the heuristic never activates the control u_6 and this produces a slower decrement of the state dynamic. The reason is that described in the previous example.

In the case *B* we reduce the costs coefficient c_x and c_T and increase the prediction horizon (and lookahead) T_x . The MPC solution remains in agreement with the global, optimal solution but now the control u_6 is not used by the MPC. Clearly the MPC judges u_6 not convenient: the reason may be that the

	A	B	C
$(\mathbf{c}_u)_r$	$\forall r \ 1$	$\forall r \ 1$	$\forall r \ 1$
c_x	1.200	1.000	0.500
c_T	25.000	15.000	4.000
T_x	8	12	12
T_u	4	4	4
Γ no control	69.230	51.858	22.429
Γ maximally feasible ctrl	≤ 26.529 ≥ 6.529	≤ 25.386 ≥ 5.386	≤ 22.660 ≥ 2.660
Γ Global optimization	14.432	13.197	10.429
Γ MPC optimization	14.572	14.302	22.377
Γ Linearization m. opt.	16.071	49.810	22.399

Table 4.9: Optimal control problem coefficient, parameter, and objective function results for the three scenarios A , B and C of Example 4.

prediction horizon T_x is smaller than T or the limited control coordination.

In the case B the Linearization Method fails to approximate the MPC and thus the global solution. The heuristic considers not convenient to control the activation dynamic and the differences start to build up from the first few steps. The heuristic does not use the control variable u_5 and this makes the state increase. After the first few steps all the control variables become inconvenient so they are all set to zero.

We further reduce the cost coefficients c_x and c_T in the sub-case C and this prevents the MPC approach to identify the optimal solution. The differences start to build up since the beginning: the MPC does not use the control variable u_5 and this makes the state increase dynamic increase since the beginning. After a few steps, the MPC judges as inconvenient to control the dynamic and the state trajectory quickly reaches one.

The cost coefficients c_x and c_T used in case C make still optimal to control the dynamic but with a smaller advantage over not controlling it. This can be observed in Table (4.9) by comparing the values of the uncontrolled objective function and the optimal objective function. The optimal solution is computed by optimizing the choice of all the control variables together whereas the MPC subroutine has a shorter prediction and control horizon. Even if the MPC does not approximate the subroutine objective function, in case C it judges more convenient not to control the system.

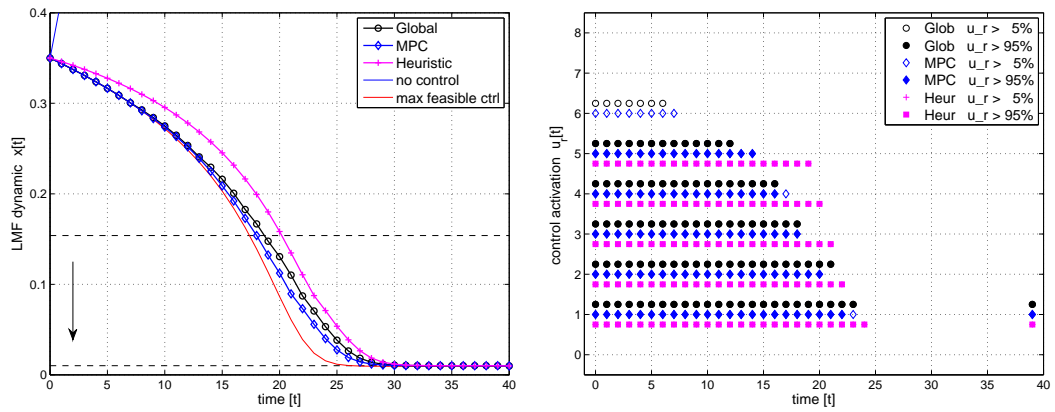


Figure 4.11: Optimization results for the scenario *A* of Example 4. Left: plot of the trajectories $x[\cdot]$. Right: graphical representation of the control sequence $\mathbf{u}[t]$.

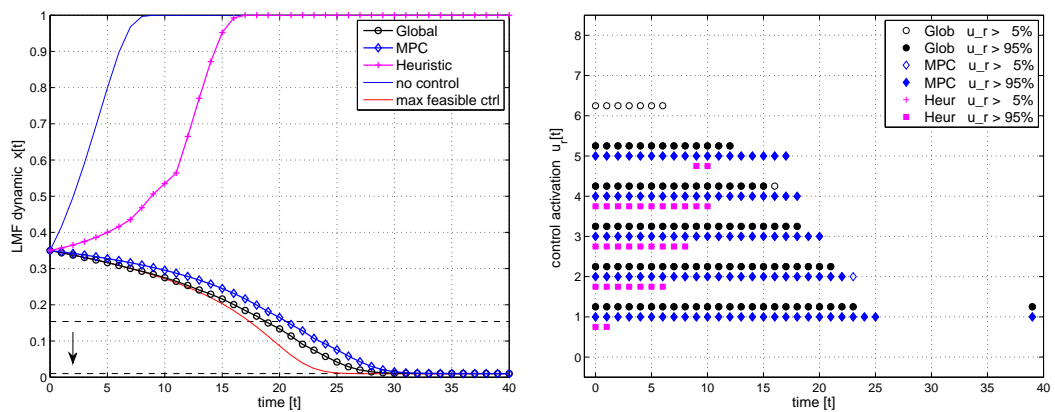


Figure 4.12: Optimization results for the scenario *B* of Example 4. Left: plot of the trajectories $x[\cdot]$. Right: graphical representation of the control sequence $\mathbf{u}[t]$.

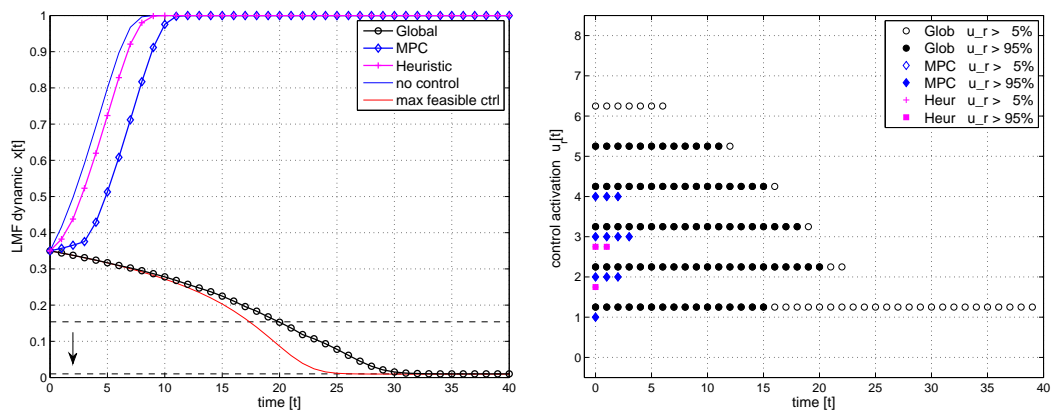


Figure 4.13: Optimization results for the scenario C of Example 4. Left: plot of the trajectories $x[\cdot]$. Right: graphical representation of the control sequence $\mathbf{u}[t]$.

Conclusion and future work

We have started the Dissertation with the description of two similar activation processes for a network of agents. The Linear Threshold Model and its permanent version are used to describe binary decision processes, where the choice of one agent depends on the aggregate choices of the neighboring agents. This type of processes can be found in a variety of contexts (e.g. the modeling of financial contagion) and may exhibit cascading phenomena.

The social and economic networks of interest are usually large, and often only statistical informations (e.g. the degree distribution) about the agents are available. One approach to avoid the use of the detailed topological structure of those large network consists in the adoption of a random network model. The random network models can incorporate the degree distribution and are generally suitable to reproduce the average features of the large networks.

We studied the Linear Threshold Model on a directed configuration model random network and we analyzed the transient dynamic of the activation process. We adopted a Local Mean-Field approach based on two recursive equations and proved a concentration theorem. If the size of the network grows large, the behavior of the Linear Threshold Model converges exponentially fast to its Local Mean-Field approximation.

We analyzed extensively the Local Mean-Field dynamical system in the special case of constant thresholds. Simulations confirmed the dynamics of the Local Mean-Field approximation. We observed a selective percolation phenomena on a regular graph with at most two different, fixed threshold values.

The last Chapter was devoted to the control of the Linear Threshold Model, as to influence the choices made by the agents in the network. The first step in this direction was the control of the Local Mean-Field dynamical system. We formulated an optimal control problem, whose solution shall provide statistical information about the types of nodes that are more likely to bias the network toward the complete activation or inactivation. We considered a regular network, where the threshold of each agent can be increased by one unit subject to some constraints. We discussed two sub-optimal approaches and observed that the Model Predictive Control gave results aligned with the optimal solution.

In the simplified control problem, we found that the type of agents which bias the network the most are those for which the threshold increment is more likely to make the difference between activation and deactivation.

Future directions

This work can be extended in several directions; here we just mention a few of them. *First of all, we shall clear the issues with the proof of Lemma 2.2 and generalize it for non-regular graphs. We envision that this can be done using a “Branching Process” network by studying the expected activation of the root node and by bounding the difference with the respect to the original random network.*

It would be interesting to compare the behavior of the process on different network topologies. The analysis has been done on the directed configuration model random network, because it simplifies the computation of the Local Mean-Field recursion, which continues to hold for time varying threshold. On the *undirected* configuration model, the Local Mean-Field recursion is more complicated, and the edge dynamic assumes a special “message passing” meaning. The work [5] regarding the permanent version of the Linear Threshold Model on undirected random regular graph is relevant to this extension. Further research is also needed to investigate whether the variable thresholds can be correctly handled on a undirected network.

Another direction for future research is the search for phenomena similar to the selective percolation in other network structure. It would also be interesting to quantify the heterogeneity necessary to observe such selective percolation.

The work regarding the control of network with variable thresholds can be extended in a few way. First, we shall check the results for different cost functions. Second, we shall allow the presence of nodes with different degrees and check how this influences the optimal solution. Finally, we may think to vary not only the thresholds but also the degree of the nodes.

Appendix A

Graph Theory

A multi-agent systems can be described with a graph where each node represents an agent, endowed with a state variable. These state variables evolve following a (typically simple) dynamical system, where the variable of one agent is depends on those of the neighboring agents. The pattern of dependencies is represented by the edges of the graph and for this reason we are mainly interested in directed graphs. We consider static graphs, i.e. the topology of the interactions does not change as time unfolds. In this Appendix we recall the basic concepts and notation used throughout the Dissertation, with the attempt to establish a minimal but coherent language. The definitions are inspired by [7, 11, 19, 37]; we refer to the books [7, 11] for further details.

A.1 Fundamentals

A *graph* G is an ordered pair of disjoint sets (V, E) such that E is a subset of the set $V \times V$ of the ordered pairs of V . The elements of V are the *vertices* (or *nodes*) of G , whereas E contains the *edges* (or *directed edges* or *arcs*) of the graph. For this reason the graph $G = (V, E)$ may also be called *directed graphs* or *digraphs*. We consider finite graphs, which means the set V and E are finite. The number of vertices is the *size* $n = |V|$ of the graph; the number of edges is denoted simply by $|E|$. We will use the letters v, w, u to indicate generic nodes in G and the letter e for the edges. We may occasionally write $v \in G$ as a shortcut of $v \in V$ for the graph $G = (V, E)$.

Each edge represents a connection between two nodes, not necessarily distinct. The edge $(u, v) \in E$ is said to be *joining* u to v or *directed from* u to v , and we say that u is *adjacent*¹ to v . The vertices u and v are the *end-points* of the edge (u, v) : u is the *starting vertex* and v is the *terminating vertex* of the

¹If both (u, v) and (v, u) are in E we say that u and v are *adjacent*.

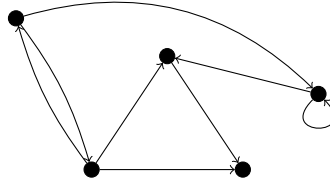


Figure A.1: A graph.

edge. An edge of the form (u, u) is said to be a *self-loop* or simply a *loop*. A *oriented graph* is a graph with no self-loop and where, for each pair of nodes $u \neq v$ at most one of (u, v) and (v, u) is present in E . It is natural to draw graphs: a node is represented by a dot or circle whereas an edge (u, v) is represented by an arrow with tail on u and head on v . With the term *topology* we intend the generic structure of the graph, nodes and connections. In the thesis we deal only with *static* networks, i.e. graphs G on top of which some dynamic process may happen but whose node and edge set do not change as time goes by.

For the purpose of the thesis, our attention is mainly devoted to directed graphs. When introducing the fundamentals of graph theory, authors (e.g. Bollobás [11]) usually call “graph” what we call here “undirected graph”. An *undirected graph* ${}^uG = (V, {}^uE)$ is an ordered pair of disjoint sets $(V, {}^uE)$ such that uE is a subset of the set $V^{(2)}$ of unordered pairs of V . Essentially the set uE is a set containing 2-element subsets of V , called *undirected edges* ue , denoted as $\{u, v\}$ and drawn as segments. By definition an undirected graph does not contain self-loops. We can obtain an undirected graph uG from a directed graph G by removing self-loops, and dropping the order of the nodes in the edges. On the other hand, if we assign an order to every undirected edge of the undirected graph uG we obtain a oriented graph G . We call *biorientation* of uG a graph G obtained from uG by replacing each undirect edge $\{v, w\}$ of uG by either (v, w) or (w, v) or both edges. A graph G is said to be *oriented* if it does not have self-loop and with $u \neq v$, $(u, v) \in E$ implies $(v, u) \notin E$.

Given a graph $G = (V, E)$, the *reverse* graph of G is the graph obtained by reversing all arcs: $\text{rev}(G) = (V, \{(v, w) \in V \times V : (w, v) \in E\})$. A graph such that $\text{rev}(G) = G$ is said to be *symmetric* and has the special property that $(u, v) \in E$ iff $(v, u) \in E$. In drawing a symmetric graph (or symmetric edges) we may use double headed arrows instead of a pair of arrows. Note that a (directed) symmetric graph without self-loop can be seen as the *complete biorientation* of a undirected graph.

Two graphs $G = (V, E)$ and $G' = (V', E')$ are said to be *isomorphic* if there a correspondence between their vertex sets that preserves adjacency. That is if

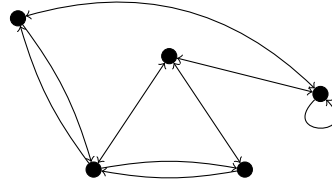


Figure A.2: A symmetric graph.

there exist a bijection $\psi : V \rightarrow V'$ such that $(v, w) \in E$ iff $(\psi(v), \psi(w)) \in E'$. Isomorphic graphs have the same size and number of edges, and only differ in the labeling of vertices. Sometimes it is convenient to identify the vertex set with a set of integers, $V = \{1, \dots, |V|\}$.

The *empty* graph of size n is the trivial graph $E_n = (V, \emptyset)$ with $n = |V|$ vertices and no edges. The *complete* graph of size n is the graph $K_n = (V, \{(u, v) : u \neq v\})$ with $n = |V|$ vertices and $n(n - 1)$ edges. In E_n no vertex is adjacent to any other, while in K_n every two distinct vertices are adjacent.

Consider two graphs $G = (V, E)$ and $G' = (V', E')$: the *intersection* and the *union* of the two graphs are defined by, respectively, $G \cap G' = (V \cap V', E \cap E')$ and $G \cup G' = (V \cup V', E \cup E')$. The graph $G' = (V', E')$ is said to be the *subgraph* of $G = (V, E)$ if $V' \subset V$ and $E' \subset E$, and in such a case we write $G' \subset G$. If the edge set of the subgraph is $E' = \{(u, v) \in E : u, v \in V'\}$, i.e. is the intersection $E \cap V' \times V'$, then G' is said to be the subgraph *induced by* V' and is denoted by $G[V']$. A subgraph G' is said to be *spanning* if $V' = V$.

Given a graph $G = (V, E)$ we can obtain new graphs by removing or adding some vertices or edges. If $W \subset V$, the subgraph $G - W = G[V \setminus W]$ is obtained by removing the vertices in W and every edges with an end-point in W . Similarly, if $E' \subset E$, then $G - E' = (V, E \setminus E')$. If $W = \{w\}$ and $E' = \{(u, v)\}$ the notation is simplified to $G - w$ and $G - (u, v)$.

An *out-neighbor* of the vertex $v \in G$ is a vertex w for which $(v, w) \in E$: the set of out-neighbors of v is $N_v^+ = \{w \in V : (v, w) \in E\}$. Correspondingly an *in-neighbor* of the vertex $v \in G$ is a vertex u for which $(u, v) \in E$: the in-neighbors set of v is $N_v^- = \{u \in V : (u, v) \in E\}$. The *out-degree* and *in-degree* of the vertex v are respectively the number of out-neighbors $d_v = |N_v^+|$ and in-neighbors $k_v = |N_v^-|$. A vertex is called *sink* if without out-neighbors, *source* if without in-neighbors. A graph is said to be *out-regular* (or *d-out-regular*) if the out-degree of every node is d . It is said to be *in-regular* (*k-in-regular*) if every node has in-degree k . If $V = \{1, \dots, n\}$, the collections $(d_i)_1^n$ and $(k_i)_1^n$, of all the out-degrees and in-degrees, are respectively called the *out-degree sequence*

and the *in-degree sequence* of G . Note that for every graph it holds:

$$|E| = \sum_{v \in V} d_v = \sum_{v \in V} k_v.$$

In a symmetric graph the out-neighbors set N_v^+ and the in-neighbors set N_v^- of every node coincide: their elements will be called simply *neighbors*, belonging to a set N_v . The term *degree* correspondingly mean both the out/in-degree, denoted by d_v . In general, for symmetric graphs when it is not necessary to be more specific, we drop the *out/in* prefix and eventual $+/-$ signs from the notation.

Given a graph $G = (V, E)$, consider two vertices u and v . A *path from u to v* is an ordered list of nodes (w_0, \dots, w_ℓ) such that $w_0 = u$, $w_\ell = v$ and, for every $i \in \{0, \dots, \ell - 1\}$, $(w_i, w_{i+1}) \in E$. The edges present in the definition of the path are said to *insist* on the path itself. The *length* of the path (w_0, \dots, w_ℓ) is the number ℓ of edges insisting on the path. If a path from u to v exists we say that the path *starts from u* , *ends at v* and *v is reachable from u* . A path from a node to itself is said to be a *circuit*.

The *distance from u to v* on G is the length of the shortest path from u to v , if any:

$$\text{dist}_G(u, v) = \min\{\ell : \text{exists in } G \text{ a path from } u \text{ to } v \text{ of length } \ell\},$$

provided $u \neq v$. We set $\text{dist}_G(u, u) = 0$ and if v is not reachable from u on G , $\text{dist}_G(u, v) = +\infty$. In general, $\text{dist}_G(\cdot, \cdot)$ is not a metric because it lacks the symmetry property. It is a metric if G is a symmetric graph. Undirected graphs inherit similar notion of path and distance.

With the notion of path we can introduce the following definitions of connectivity. A graph $G = (V, E)$ is said to be:

- *strongly connected* if for every pair of nodes u and v both v is reachable from u and u is reachable from v ;
- *connected* if for every pair of node u, v , either v is reachable from u or u is reachable from v ;
- *weakly connected* if $G \cup \text{rev}(G)$ is strongly connected.

These three notions are all equivalent for symmetric graphs (and for undirected graphs), where we simply use the term *connected*.

Given a graph $G = (V, E)$ the *adjacency matrix* A is the matrix in $\{0, 1\}^{V \times V}$ that fully represent the graph G . The entries of the adjacency matrix A are

$$\begin{cases} A_{uv} = 1 & \text{if } (u, v) \in E \\ A_{uv} = 0 & \text{if } (u, v) \notin E. \end{cases}$$

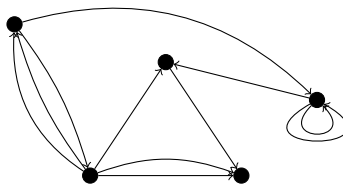


Figure A.3: A multigraph.

We conclude the Section with the definition of weighted graphs and corresponding matrix representation. Given a graph $G = (V, E)$ and a function $c : E \rightarrow \mathbb{R}^+$, a *weighted graph* is a triple $G = (V, E, c)$. The value $c(e)$, with e an edge in E , is called *weight* of e . It is easy to associate to a weighted graph a *weight matrix* $W \in \mathbb{R}_{\geq 0}^{V \times V}$ that generalizes the adjacency matrix:

$$\begin{cases} W_{uv} = c((u, v)) & \text{if } (u, v) \in E \\ W_{uv} = 0 & \text{if } (u, v) \notin E. \end{cases}$$

This weight matrix W defined above fully represents the network, as positive entries are related to weighted edges whereas null entries mean that the edge is not present in the graph. Here we do not consider situations where some edges may have a null weight. Then, we may use directly the weight matrix to define the weighted graph, as in Section 1.4.

A.2 Multigraphs, neighborhoods and trees

In this Section we introduce some advanced definitions, building blocks for the lemmas in the following Section. By definition a graph $G = (V, E)$ does not contain *multiple* (or *parallel*) *edges*, which are multiple copies of the same edge $(u, v) \in E$. In a *multigraph*, E is a *multiset* and multiple edges are allowed. All the above definitions for graphs extend naturally to multigraphs, for example with each copy of an edge contributing once to the out-degree and in-degree of the vertices involved. A graph is always a multigraph, but not the converse. When confusion may arise, standard graphs are also called *simple graphs*.

In the following we define precisely the concept of “neighborhood of depth ℓ ” (adapted from [37]) and “out/in-tree” (from [7]), used in Chapter 2. The definitions are given for multigraphs but work in the same way for simple graphs.

Consider a multigraph $G = (V, E)$ and a given vertex u . The *out-neighborhood of depth ℓ of node u* is the (multi-) subgraph $\mathcal{N}_{u,+}^{\ell} = (V', E') \subset G$ with:

$$\begin{aligned} V' &= \{v \in V : \text{dist}_G(u, v) \leq \ell\} \\ E' &= \{(w, v) \in E : \text{dist}_G(u, w) \leq \ell - 1\}. \end{aligned}$$

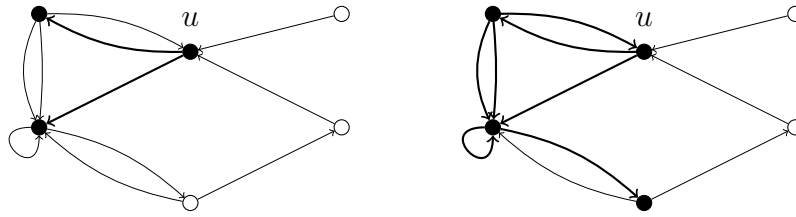


Figure A.4: Out-neighborhood $\mathcal{N}_{u,+}^1$ (left) and $\mathcal{N}_{u,+}^2$ (right) of G .

In the multigraph G consider every path of length ℓ starting from u . The out-neighborhood of depth ℓ of the node u is the multigraph which contains the edges insisting on all such paths, with the corresponding nodes.

Note that in general $\mathcal{N}_{u,+}^\ell$ is not the subgraph induced by V' , but $\mathcal{N}_{u,+}^\ell \subset G[V']$. The induced subgraph $G[V']$ contains those edges which connect nodes already at distance ℓ from u to nodes at distance at most ℓ from u . Such edges (or loops) are not present in E' , therefore $E' \subset E \cap V' \times V'$.

The *in-neighborhood of depth ℓ of node u* is defined similarly as the subgraph $\mathcal{N}_{u,-}^\ell = (V'', E'') \subset G$ with:

$$\begin{aligned} V'' &= \{v \in V : \text{dist}_G(v, u) \leq \ell\} \\ E'' &= \{(v, w) \in E : \text{dist}_G(w, u) \leq \ell - 1\}. \end{aligned}$$

Before introducing the concept of “out/in-tree” for multigraphs, we recall which undirected graphs are called “tree”. A *tree* is a connected undirected graph ${}^uT = (V, {}^uE)$ with $|{}^uE| = |V| - 1$: the number of edges equals the number of vertices minus one. The former definition of a tree is usually regarded as an equivalent characterization (see [11]) but avoids the use of other preliminary concepts. Intuitively, a tree is both the “minimally connected” and “maximal acyclic” undirected graph.

We extend the concept of tree to directed multigraphs by the following definitions. An *out-tree* is a weakly connected graph $T^+ = (V, E)$ which has exactly one source vertex v whereas all the other vertex $u \in V \setminus \{v\}$ have in-degree exactly one, i.e. T^+ is weakly connected, $\exists! v \in T^+ : k_v = 0$ and $\forall u \in T^+ - v, k_u = 1$. The source vertex is called *root* with the out-tree “emerging” from it. Similarly, an *in-tree* is a weakly connected graph $T^- = (V, E)$ such that $\exists! v \in T^- : d_v = 0$ and $\forall u \in T^- - v, d_u = 1$. The sink vertex v is also called *root*.

It is easy to check that an out/in-tree of size n is an oriented graphs with exactly $n - 1$ edges, all oriented “away/towards” the root. Hence, a multigraph with at least a proper multiple edge, or a graph with self loop, can never be a tree. In Figure A.5 it is drawn an out-tree and several modifications which are

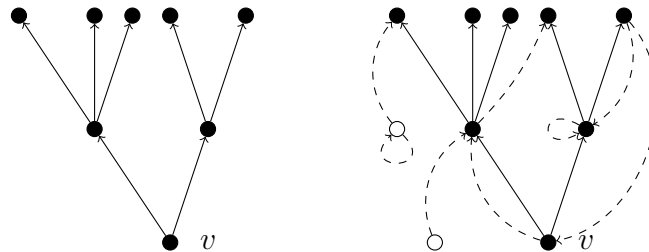


Figure A.5: An out-tree and what is not allowed.

not allowed by the definition. Notice that a graph (u, \emptyset) made of just one node is a (trivial) out/in-tree.

If we drop the the orientation of the edges, we obtain an undirected tree. Notice that it is however possible to obtains undirected trees from many directed graphs which are not necessarily out/in-tree. This has made authors give different names to what we call out/in-tree, e.g. *arborescence*.

A.3 Random Graphs

This Section contains some properties of the random graph model used in Chapter 2. We recall that in Chapter 2 we analyzed the *Linear Threshold Model* (1.1) on a large random network, specifically the *directed* version of the *configuration model* random graph, equipped with state variables and activation thresholds.

The general term *random graph* refers to probability distributions over graphs, and the theory of random graphs lies at the intersection of graph theory and probability theory. An introduction to the random graph theory is far from the scope of this Section. For a discussion of the topic and its application, we refer to the review articles [2, 34] and to the book [16]. Here we just provide a few hint and move to the results used in Chapter 2.

In general, random graphs may be described by a probability distribution over a set of actual graphs, or by the random process which generates them. There are various families and models or random graphs, to model various kind of complex networks encountered in different areas. They are used to understand what happens on the typical network.

The *configuration model* is a random graph model, that can incorporate a specific *degree distribution* i.e. the probability distribution on the node's degree can be specified in advance. The graph used for the analysis of Chapter 2 is a directed version of the configuration model. The graph model is described in Section 2.1.1 and is prepared with a procedure adapted from [16, ch. 3]. We actually specify the *out-degree* and *in-degree* sequences and compute the

statistics afterwards. A weighted version of the configuration model has been used in [4] to model financial networks.

We stress an important aspect related to the degrees of the nodes in a non-regular graph $G = (V, E)$. Suppose that $p_{d,k}$ is the probability than a randomly chosen vertex of V has *out*-degree d and *in*-degree k . If we choose uniformly at random an edge $e = (v, w)$ from E , the *out*-degree and *in*-degree of the terminating vertex w are distributed according to $k p_{d,k} (\sum_{d,k} k p_{d,k})^{-1}$, whereas the *out*-degree and *in*-degree of the starting vertex v are distributed according to $d p_{d,k} (\sum_{d,k} d p_{d,k})^{-1}$.

We get to one important feature of the directed configuration model, heavily used the analysis of Chapter 2. The directed configuration model is *locally like a tree*. Let u be a given vertex in a randomly chosen graph $G = (V, E)$ of the directed configuration model ensemble, with n nodes. Consider a depth L . With this statement above we mean that if n is large, there is a high probability that the *out*-neighborhood of depth L of u is an out-tree. Similarly for the probability of the *in*-neighborhood being an in-tree. The case for the *out*-neighborhood is proven in the following lemma, which suppose the network model has a maximum *out*-degree and a maximum *in*-degree. The proof is inspired by a result in [37].

Lemma A.1. *Let the graph $G = (V, E)$ be a randomly chosen element of the directed configuration model ensemble with n nodes and $|E|$ edges. Let the maximum *out*-degree, *in*-degree and average degree be d_{max} , k_{max} and \bar{d} respectively, with $d_{max} > 1$ and $\bar{d} \geq 1$. Let $u \in V$. For the *out*-neighborhood of depth L of the node u , i.e. $\mathcal{N}_{u,+}^L$, it holds*

$$\mathbb{P}(\mathcal{N}_{u,+}^L \text{ is not an out-tree}) \leq \frac{\gamma}{n}$$

with $\gamma = k_{max} d_{max}^{2L+2}$.

Proof. Consider a depth $l < L$. Let $\mathcal{N}_{u,+}^l$ be the *out*-neighborhood of depth l of the node u . First, we assume $\mathcal{N}_{u,+}^l$ is an out-tree and bound the probability that $\mathcal{N}_{u,+}^{l+1}$ is an out-tree. Let n_l be the number of nodes in $\mathcal{N}_{u,+}^l$ with u included and m_l be the number of *in*-sockets of the n_l nodes in $\mathcal{N}_{u,+}^l$. Both n_l and m_l are random variables satisfying:

$$n_l \leq \sum_{i=0}^l d_{max}^i = \theta_l$$

$$m_l \leq k_{max} \theta_l$$

Suppose that we have revealed h further edges, from nodes at depth l to nodes at depth $l + 1$, and preserved the out-tree structure. Let \tilde{h} be the number of

in-sockets of those newly revealed nodes. There are $|E| - (n_l - 1) - h$ available *in*-sockets to plug the next edge, of which $|E| - m_l - \tilde{h}$ preserve the out-tree structure. Then, the probability to preserve the out-tree structure is:

$$\frac{|E| - m_l - \tilde{h}}{|E| - (n_l - 1) - h} \geq \frac{|E| - k_{max}\theta_l - k_{max}h}{|E|} = 1 - k_{max} \frac{\theta_l + h}{|E|} \geq 1 - \frac{k_{max}\theta_L}{\bar{d}n}$$

There are $n_{l+1} - n_l$ of those edges to be revealed, therefore

$$\mathbb{P}(\mathcal{N}_{u,+}^{l+1} \text{ is an out-tree} | \mathcal{N}_{u,+}^l \text{ is an out-tree}) \geq \left(1 - \frac{k_{max}\theta_L}{\bar{d}n}\right)^{n_{l+1} - n_l}$$

Given $\mathbb{P}(\mathcal{N}_{u,+}^0 \text{ is an out-tree}) = 1$, and iterating, we get

$$\mathbb{P}(\mathcal{N}_{u,+}^L \text{ is an out-tree}) \geq \left(1 - \frac{k_{max}\theta_L}{\bar{d}n}\right)^{n_L} \geq \left(1 - \frac{k_{max}\theta_L}{\bar{d}n}\right)^{\theta_L} \geq 1 - \frac{k_{max}\theta_L^2}{\bar{d}n}$$

by using $(1 - x)^a \geq 1 - ax$ when $a \geq 1$ and $x \in [0, 1]$. When \bar{d} and d_{max} are bigger than one, we have $\theta_L = \sum_{i=0}^L d_{max}^i \leq d_{max}^{L+1}$ and then

$$\mathbb{P}(\mathcal{N}_{u,+}^L \text{ is not an out-tree}) \leq \frac{k_{max}\theta_L^2}{\bar{d}n} \leq \frac{k_{max}d_{max}^{2L+2}}{n}$$

□

The following lemma says that, as the number of vertexes n becomes large, the probability of the *out*-neighborhood of the vertex u being an out-tree is high for depths of order $\log n$.

Lemma A.2. *Let the graph $G = (V, E)$ be a randomly chosen element of the directed configuration model ensemble with n nodes and $|E|$ edges. Let the maximum out-degree, in-degree and average degree be d_{max} , k_{max} and \bar{d} respectively, with $d_{max} > 1$ and $\bar{d} \geq 1$. Let $u \in V$. If the size n grows large (i.e. tends to ∞) and the depth L is of order $\log n$, the *out*-neighborhood of depth L of node u , $\mathcal{N}_{u,+}^L$, is an out-tree with high probability.*

Proof. Given the hypothesis, from lemma A.1, it holds

$$\mathbb{P}(\mathcal{N}_{u,+}^L \text{ is not an out-tree}) \leq \frac{k_{max}d_{max}^{2L+2}}{n}$$

With a depth $L = L(n)$ such that $d_{max}^{2L(n)} = o(n)$ when $n \rightarrow \infty$, the corresponding limit of $\mathbb{P}(\mathcal{N}_{u,+}^L \text{ is not an out-tree})$ tends to zero. Hence, we can set $d_{max}^{2L(n)} = n^\alpha$ with $\alpha < 1$, which gives

$$L(n) = \frac{\alpha}{2 \log d_{max}} \log n.$$

□

Appendix B

Properties of functions in the LMF equations

The Local Mean-Field approach is based on a dynamical system with two recursive equations. To write those equations we introduced in Chapter 2 the family of functions $f_{d,r}(x)$, also used in Chapter 3. In Chapter 4 in addition we introduced the functions $g_{d,r}(x)$. In this Appendix we collect a list of properties of these two family of functions.

B.1 Properties of $f_{d,r}(x)$

The family of functions $f_{d,r} : [0, 1] \rightarrow [0, 1]$, with parameter $d \in \mathbb{N}$ and $r \in \{0, 1, \dots, d + 1\}$ is defined by

$$f_{d,r}(x) := \sum_{i=r}^d \binom{d}{i} x^i (1-x)^{d-i}.$$

The following are the expressions of a few functions in the family:

$$f_{d,0}(x) = 1 \quad \forall x \in [0, 1] \tag{B.1}$$

$$f_{d,1}(x) = 1 - (1-x)^d \tag{B.2}$$

$$f_{d,2}(x) = 1 - (1-x)^{d-1}(1 + (d-1)x) \tag{B.3}$$

$$\vdots \tag{B.4}$$

$$f_{d,d-1}(x) = x^{d-1}(d - (d-1)x) \tag{B.5}$$

$$f_{d,d}(x) = x^d \tag{B.6}$$

$$f_{d,d+1}(x) = 0 \quad \forall x \in [0, 1]. \tag{B.7}$$

A few useful properties are the following:

$$f_{d,r}(0) = \begin{cases} 1 & r = 0 \\ 0 & r \geq 1 \end{cases} \quad (\text{B.8})$$

$$f_{d,r}(1) = \begin{cases} 1 & r \leq 0 \\ 0 & r = d + 1 \end{cases} \quad (\text{B.9})$$

$$\sum_{r=1}^d f_{d,r}(x) = dx \quad (\text{B.10})$$

$$f_{d,r+1}(x) - f_{d,r}(x) = -\binom{d}{r} x^r (1-x)^{d-r} \quad (\text{B.11})$$

$$f_{d,r}(x) = x f_{d-1,r-1}(x) + (1-x) f_{d-1,r}(x). \quad (\text{B.12})$$

The first derivative of $f_{d,r}(x)$, with $d \geq 1$ and $r \in \{1, 2, \dots, d\}$ has general expression:

$$f'_{d,r}(x) = \binom{d}{r} r x^{r-1} (1-x)^{d-r} \quad (\text{B.13})$$

The expression of some specific derivative are:

$$f'_{d,0}(x) = 0 \quad \forall x \in [0, 1] \quad (\text{B.14})$$

$$f'_{d,1}(x) = d(1-x)^{d-1} \quad (\text{B.15})$$

$$f'_{d,2}(x) = d(d-1)x(1-x)^{d-2} \quad (\text{B.16})$$

$$\vdots \quad (\text{B.17})$$

$$f'_{d,d-1}(x) = d(d-1)x^{d-2}(1-x) \quad (\text{B.18})$$

$$f'_{d,d}(x) = dx^{d-1} \quad (\text{B.19})$$

$$f'_{d,d+1}(x) = 0 \quad \forall x \in [0, 1] \quad (\text{B.20})$$

Some properties are:

$$f'_{d,r}(0) = \begin{cases} d & r = 1 \\ 0 & r \neq 1 \end{cases} \quad (\text{B.21})$$

$$f'_{d,r}(1) = \begin{cases} 0 & r \neq d \\ d & r = d \end{cases} \quad (\text{B.22})$$

$$f'_{d,r}(x) > 0 \quad \forall x \in (0, 1) \text{ with } r \neq 0, d+1 \quad (\text{B.23})$$

$$\sum_{r=1}^d f'_{d,r}(x) = d \quad (\text{B.24})$$

$$f'_{d,r+1}(x) - f'_{d,r}(x) = -\frac{x}{d} f'_{d,r}(x). \quad (\text{B.25})$$

We observe that except for the trivial cases with $r = 0, d + 1$, all the functions $f_{d,r}(x)$ are strictly increasing in $[0, 1]$.

In the non trivial cases, the second derivative of $f_{d,r}(x)$ as general expression:

$$f''_{d,r}(x) = \binom{d}{r} r x^{r-2} (1-x)^{d-r-1} (r-1 - (d-1)x) \quad (\text{B.26})$$

Some specific expressions are the following:

$$f''_{d,0}(x) = 0 \quad \forall x \in [0, 1] \quad (\text{B.27})$$

$$f''_{d,1}(x) = -d(d-1)(1-x)^{d-2} \quad (\text{B.28})$$

$$f''_{d,2}(x) = d(d-1)(1-x)^{d-3}(1 - (d-1)x) \quad (\text{B.29})$$

$$\vdots \quad (\text{B.30})$$

$$f''_{d,d-1}(x) = d(d-1)x^{d-3}(d-2 - (d-1)x) \quad (\text{B.31})$$

$$f''_{d,d}(x) = d(d-1)x^{d-2} \quad (\text{B.32})$$

$$f''_{d,d+1}(x) = 0 \quad \forall x \in [0, 1]. \quad (\text{B.33})$$

A few properties are:

$$f''_{d,1}(x) < 0 \quad \forall x \in (0, 1) \text{ with } d \geq 2 \quad (\text{B.34})$$

$$f''_{d,d}(x) > 0 \quad \forall x \in (0, 1) \text{ with } d \geq 2 \quad (\text{B.35})$$

$$\sum_{r=1}^d f''_{d,r}(x) = 0. \quad (\text{B.36})$$

From the analysis of the second derivative, we found that $f_{d,r}(x)$ has an inflection point in

$$\bar{x}_{d,r} = \frac{r-1}{d-1}$$

and it holds:

$$\begin{cases} f''_{d,r}(x) > 0 & x < \bar{x}_{d,r} \\ f''_{d,r}(x) < 0 & x > \bar{x}_{d,r} \end{cases}$$

To following Lemma helps the study the LMF dynamical system when only one function $f_{d,r}(x)$ is present in the convex combination.

Lemma B.1. *Consider the function $\hat{f}(x) = f_{d,r}(x) - x$ in the interval $[0, 1]$. Given a generic d , for $r = 1$ or $r = d$, $\hat{f}(x)$ has two zero, in 0 and 1. For $r \in \{2, 3, \dots, d-1\}$, $\hat{f}(x)$ has three distinct zeros, in 0, \bar{x} and 1.*

Proof. Notice that for every $r \in \{1, 2, \dots, d\}$, we have $\hat{f}(0) = \hat{f}(1) = 0$ and that $\hat{f}(x)$ shares the same convexities as $f_{d,r}(x)$. For $r = 1$, the function $f_{d,1}(x)$ is strictly concave in $(0, 1)$ and then $\hat{f}(x)$ cannot have internal zeros. Similarly for $r = d$, with $f_{d,d}(x)$ convex. For $r \in \{2, 3, \dots, d-1\}$ the function $f_{d,r}(x)$ has an inflection point in $x^* = (r-1)/(d-1)$, being strictly convex in $(0, x^*)$ and strictly concave in $(x^*, 1)$. Notice that $\hat{f}(x)$ is a smooth function and $\hat{f}'(0) = \hat{f}'(1) = -1$, hence $\hat{f}(x)$ need to admit further zeros in $(0, 1)$. Since $\hat{f}(x)$ changes convexity only once, at x^* , the further zero is unique. \square

At the end of Chapter 3 we introduced the function $h_{d,r}(x)$, with $d \in \mathbb{N}$ and $r \in \{1, 2, \dots, d\}$:

$$h_{d,r}(x) = \frac{1 - f_{d,r}(x)}{1 - x} = \sum_{i=0}^{r-1} \binom{d}{i} x^i (1-x)^{d-i-1} \quad (\text{B.37})$$

The first derivative of $h_{d,r}(x)$ is:

$$\begin{aligned} h'_{d,r}(x) &= \sum_{i=0}^{r-1} \binom{d}{i} x^i (1-x)^{d-i-2} - r \binom{d}{r} x^{r-1} (1-x)^{d-r-1} \\ &= \sum_{i=0}^{r-2} \binom{d}{i} x^i (1-x)^{d-i-2} - (d-r) \binom{d}{r-1} x^{r-1} (1-x)^{d-r-1}. \end{aligned}$$

B.2 Properties of $g_{d,r}(x)$

The controlled dynamic (4.11) of the case study in Chapter 4 uses the family of functions $g_{d,r} : [0, 1] \rightarrow [0, 1]$. This family of functions, with parameter $d \in \mathbb{N}$ and $r \in \{0, 1, \dots, d\}$ has general expression

$$g_{d,r}(x) := \binom{d}{r} x^r (1-x)^{d-r} \quad (\text{B.38})$$

which comes from

$$g_{d,r}(x) = -(f_{d,r+1}(x) - f_{d,r}(x)).$$

In the following we study the $g_{d,r}(x)$ as a function of x and the dependence on the parameter d and r . As a function of x , the first derivative of $g_{d,r}(x)$ is:

$$\begin{cases} g'_{d,r}(x) &= \binom{d}{r} x^{r-1} (1-x)^{d-r-1} (r - dx) \\ g'_{d,0}(x) &= -d(1-x)^{d-1} \\ g'_{d,d}(x) &= dx^{d-1}. \end{cases} \quad (\text{B.39})$$

Hence, for fixed d and r the maximum is attained at

$$x_{max} = \frac{r}{d}$$

and this expression holds for all d and r . For $r = 0$, d , $g_{d,r}(x_{max}) = 1$. If $r \neq 0$, d we have

$$g_{d,r}(x_{max}) = \frac{h(r) h(d-r)}{h(d)} \simeq \frac{\sqrt{d}}{\sqrt{2\pi r(d-r)}} \quad (\text{B.40})$$

where, for $n \in \mathbb{N}$, the function $h(n)$ ¹ is :

$$h(n) = \frac{n^n}{n!} \simeq \frac{e^n}{\sqrt{2\pi n}}.$$

To estimate the FWHM², we use the difference $x_2 - x_1$ of the two inflection points of the generic $g_{d,r}$. The generic second derivative is:

$$g_{d,r}''(x) = \binom{d}{r} x^{r-2} (1-x)^{d-r-2} (d(d-1)x^2 - 2r(d-1)x + r(r-1)) \quad (\text{B.41})$$

Solving the second order equation, one has that $x_2 - x_1 = \sqrt{\frac{\Delta}{a}}$, which, using $n-1 = n(1 - \frac{1}{n})$ and $n = (n-1)(1 - \frac{1}{n})^{-1} \simeq (n-1)(1 + \frac{1}{n})$ can be approximated

$$x_2 - x_1 \simeq 2\sqrt{\frac{r(d-r)}{d^3}} \quad (\text{B.42})$$

From a numerical comparison, this estimate is fairly good, underestimating the right quantity of about 10% - 20%.

Next, given an x_0 and d , we study the dependance of $g_{d,r}(x_0)$ from the parameter r . We are mainly interested in finding $r^* = \arg \max_r g_{d,r}(x_0)$. From the definition of $g_{d,r}$,

$$g_{d,r}(x_0) = \binom{d}{r} x_0^r (1-x_0)^{d-r}$$

we consider the expression for $r+1$ and $r-1$:

$$\begin{aligned} g_{d,r+1}(x_0) &= \binom{d}{r+1} x_0^{r+1} (1-x_0)^{d-r-1} \\ g_{d,r-1}(x_0) &= \binom{d}{r-1} x_0^{r-1} (1-x_0)^{d-r+1}. \end{aligned}$$

¹To approximate, use Stirling $n! \simeq \sqrt{2\pi n} n^n e^{-n}$.

²Full Width at Half Maximum.

Then, we study the inequality

$$\begin{aligned} g_{d,r}(x_0) &\geq g_{d,r+1}(x_0) \\ \frac{1-x_0}{d-r} &\geq \frac{x_0}{r+1} \\ r &\geq (d+1)x_0 - 1 \end{aligned}$$

and the inequality

$$\begin{aligned} g_{d,r}(x_0) &\geq g_{d,r-1}(x_0) \\ \frac{x_0}{r} &\geq \frac{1-x_0}{d-r+1} \\ r &\leq (d+1)x_0. \end{aligned}$$

Therefore, r^* is the integer in the interval $[(d+1)x_0 - 1, (d+1)x_0]$.

We do similar computations to study the dependance on d , given x_0 and r . Again, we are interested in $d^* = \arg \max_d g_{d,r}(x_0)$, and we consider $g_{d,r}(x_0)$ with $d+1$ and $d-1$:

$$\begin{aligned} g_{d+1,r}(x_0) &= \binom{d+1}{r} x_0^r (1-x_0)^{d+1-r} \\ g_{d-1,r}(x_0) &= \binom{d-1}{r} x_0^r (1-x_0)^{d-1-r}. \end{aligned}$$

We solve the inequality

$$\begin{aligned} g_{d,r}(x_0) &\geq g_{d+1,r}(x_0) \\ 1 &\geq \frac{d+1}{d+1-r} (1-x_0) \\ (d+1)x_0 &\geq r \end{aligned}$$

and the inequality

$$\begin{aligned} g_{d,r}(x_0) &\geq g_{d-1,r}(x_0) \\ 1 &\geq \frac{d-r}{d(1-x_0)} \\ dx_0 &\leq r. \end{aligned}$$

We obtain that d^* is the integer in the interval $\left[\frac{r}{x_0} - 1, \frac{r}{x_0}\right]$. Notice that that the lower bound on the interval for d^* coincides with the upper bound for r^* .

Given x_0 , Figure B.1 shows the distribution of the values $g_{d,r}(x_0)$; in the example $x_0 = 0.350$. The dots of the 20 largest elements are encircled. We also plot the line with $r/d = x_0$ as a reference.

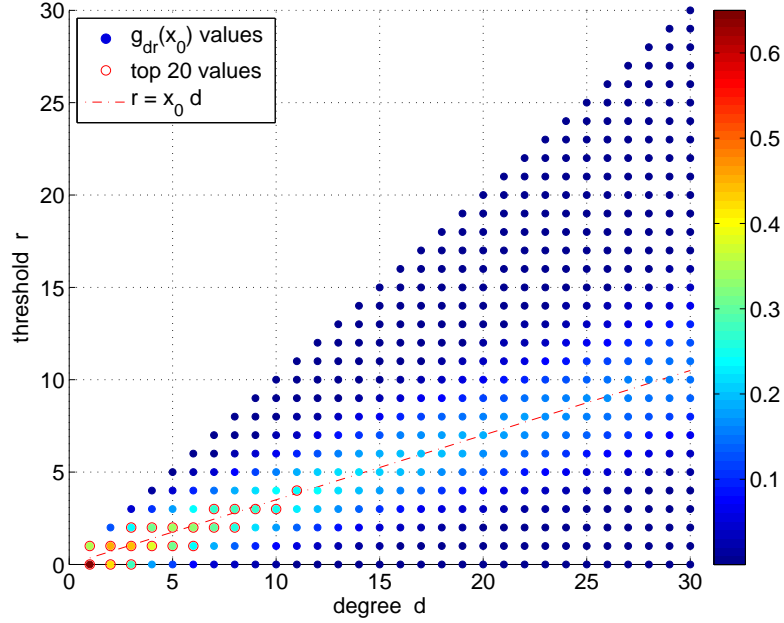


Figure B.1: Colormap plot of the values of $g_{d,r}(x_0)$, with $x_0 = 0.350$, for various pairs (d, r) . The top 20 values are encircled; the dashed line is $r/d = x_0$.

For a specific x , we may be interested in the envelope of the sequence $g_{d,r^*}(x)$ in d , or in the envelope of the sequence $g_{d^*,r}(x)$ in r . We consider the former, as the latter is essentially the same. We use the integer n to rewrite the parameters:

$$\begin{cases} d = n \\ r = \lfloor (n+1)x \rfloor \approx nx \end{cases}$$

Using the substitution above and with the Stirling's approximation,

$$\begin{aligned} g_{n,nx}(x) &= \frac{n!}{(nx)!(n-nx)!} x^{nx} (1-x)^{n-nx} \\ &\approx \frac{\sqrt{2\pi n} n^n e^{-n}}{\sqrt{2\pi nx} (nx)^{nx} e^{-nx} \sqrt{2\pi(n-nx)} (n-nx)^{n-nx} e^{-(n-nx)}} x^{nx} (1-x)^{n-nx} \\ &= \frac{1}{\sqrt{2\pi n x (1-x)}} \end{aligned}$$

which, using $d = n$ gives

$$g_{d,r^*}(x) \approx \frac{1}{\sqrt{2\pi d x (1-x)}} \quad (\text{B.43})$$

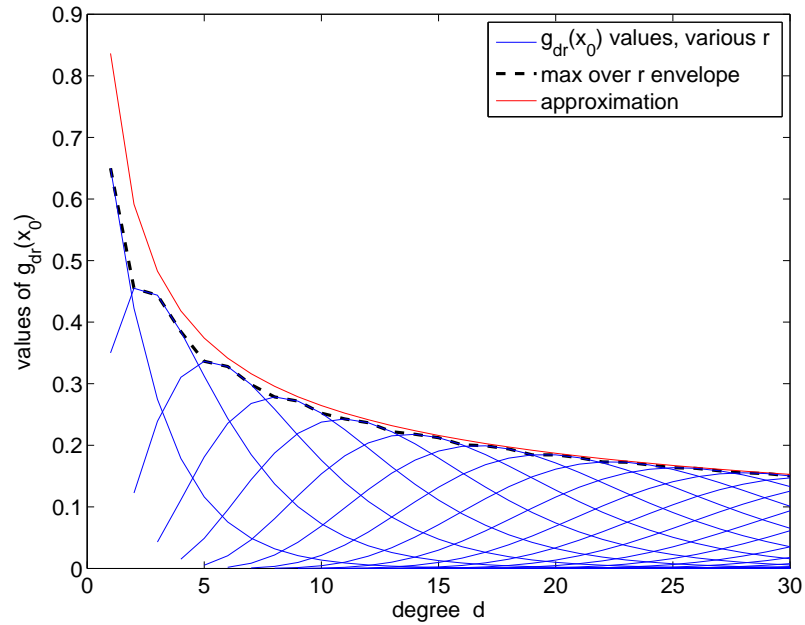


Figure B.2: Envelope of the maximum values over r , for $x_0 = 0.350$

in good agreement with the actual values.

In figure B.2, for $x_0 = 0.350$ and given r , we plot the functions $g_{d,r}(x_0)$ with respect to d . We draw the envelope of the maximum values over r and compare it with the approximation computed above.

Bibliography

- [1] E. Adam, M. Dahleh, and A. Ozdaglar. On the behavior of threshold models over finite networks. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pages 2672–2677, Dec 2012.
- [2] R. Albert and A.-L. Barabási. Statistical mechanics of complex networks. *Rev. Mod. Phys.*, 74:47–97, Jan 2002.
- [3] H. Amini. Bootstrap percolation and diffusion in random graphs with given vertex degrees. *Electronic Journal of Combinatorics*, 17:R25, 2010.
- [4] H. Amini, R. Cont, and A. Minca. Resilience to contagion in financial networks. *Mathematical Finance*, 2013.
- [5] H. Amini, M. Draief, and M. Lelarge. Marketing in a random network. In E. Altman and A. Chaintreau, editors, *Network Control and Optimization*, volume 5425 of *Lecture Notes in Computer Science*, pages 17–25. Springer Berlin Heidelberg, 2009.
- [6] J. Balogh and B. G. Pittel. Bootstrap percolation on the random regular graph. *Random Structures & Algorithms*, 30(1-2):257–286, 2007.
- [7] J. Bang-Jensen and G. Gutin. *Digraphs: Theory, Algorithms and Applications*. Springer Monographs in Mathematics. Springer-Verlag, London, UK, 2 edition, 2009.
- [8] D. P. Bertsekas. *Dynamic programming and optimal control*, volume I. Athena Scientific, Belmont, MA, USA, 3rd edition, 2005.
- [9] D. P. Bertsekas. Dynamic programming and suboptimal control: A survey from ADP to MPC. *European Journal of Control*, 11(4):310–334, 2005.
- [10] D. Bertsimas and J. N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, Nashua, NH, USA, 1997.

- [11] B. Bollobás. *Modern graph theory*, volume 184 of *Graduate texts in mathematics*. Springer-Verlag, New York, NY, USA, 1998.
- [12] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2004.
- [13] M. Cannon. C21 Model Predictive Control, winter 2015. Lecture notes, <http://www.eng.ox.ac.uk/~conmrc/mpc/index.html> (Accessed February 10,2015).
- [14] J. Chalupa, P. L. Leath, and G. R. Reich. Bootstrap percolation on a bethe lattice. *Journal of Physics C: Solid State Physics*, 12(1):L31, 1979.
- [15] R. Cont, A. Moussa, and E. B. e. Santos. Network structure and systemic risk in banking systems. In J.-P. Fouque and J. A. Langsam, editors, *Handbook of Systemic Risk*, chapter 13, pages 327–368. Cambridge University Press, 2013.
- [16] R. Durrett. *Random graph dynamics*. Cambridge university press, New York, NY, USA, 2007.
- [17] D. Easley and J. Kleinberg. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge University Press, New York, NY, USA, 2010.
- [18] H. Elsinger, A. Lehar, and M. Summer. Risk assessment for banking systems. *Management science*, 52(9):1301–1314, 2006.
- [19] F. Fagnani and P. Frasca. Lecture notes on “control of/over networks”. Draft, Apr. 2014.
- [20] D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, Cambridge, MA,USA, 1991.
- [21] F. Giavazzi. 14.02 Principles of Macroeconomics, spring 2014. Lecture notes, Massachusetts Institute of Technology: MIT OpenCouseWare, <http://ocw.mit.edu> (Accessed November 4,2014). License: Creative Commons BY-NC-SA.
- [22] M. Granovetter. Threshold models of collective behavior. *American Journal of Sociology*, 83(6):1420–1443, 1978.
- [23] M. O. Jackson. *Social and Economic Networks*. Princeton University Press, Princeton, NJ, USA, 2008.

- [24] S. Janson and M. J. Luczak. A simple solution to the k -core problem. *Random Structures & Algorithms*, 30(1-2):50–62, 2007.
- [25] S. Janson, T. Łuczak, T. Turova, and T. Vallier. Bootstrap percolation on the random graph $G_{n,p}$. *The Annals of Applied Probability*, 22(5):1989–2047, 2012.
- [26] Y. Kanoria, A. Montanari, et al. Majority dynamics on trees and the dynamic cavity method. *The Annals of Applied Probability*, 21(5):1694–1748, 2011.
- [27] D. Kempe, J. Kleinberg, and E. 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*, KDD '03, pages 137–146, New York, NY, USA, 2003. ACM.
- [28] J. Kleinberg. Cascading behavior in networks: Algorithmic and economic issues. In N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, editors, *Algorithmic game theory*, chapter 24, pages 613–632. Cambridge University Press, 2007.
- [29] M. Lelarge. Efficient control of epidemics over random networks. *SIGMETRICS Performance Evaluation Review*, 37(1):1–12, June 2009.
- [30] M. Lelarge. Diffusion and cascading behavior in random networks. *Games and Economic Behavior*, 75(2):752 – 775, 2012.
- [31] A. Montanari and A. Saberi. Convergence to equilibrium in local interaction games. In *Foundations of Computer Science, 2009. FOCS '09. 50th Annual IEEE Symposium on*, pages 303–312, Oct 2009.
- [32] A. Montanari and A. Saberi. The spread of innovations in social networks. *Proceedings of the National Academy of Sciences*, 107(47):20196–20201, 2010.
- [33] S. Morris. Contagion. *The Review of Economic Studies*, 67(1):57–78, 2000.
- [34] M. Newman. The structure and function of complex networks. *SIAM Review*, 45(2):167–256, 2003.
- [35] M. E. J. Newman, S. H. Strogatz, and D. J. Watts. Random graphs with arbitrary degree distributions and their applications. *Phys. Rev. E*, 64:026118, Jul 2001.

- [36] A. Ozdaglar. Gams Theory with Engineering Applications, spring 2010. Lecture notes, (Massachusetts Institute of Technology: MIT OpenCourseWare), <http://ocw.mit.edu> (Accessed November 4,2014). License: Creative Commons BY-NC-SA.
- [37] T. J. Richardson and R. L. Urbanke. The capacity of low-density parity-check codes under message-passing decoding. *Information Theory, IEEE Transactions on*, 47(2):599–618, Feb 2001.
- [38] J. R. Riehl and M. Cao. Towards control of evolutionary games on networks. In *Proceedings of the 53rd IEEE Conference on Decision and Control*, December 2014. To appear.
- [39] S. H. Strogatz. *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering*. Advanced book program. Perseus Books, Reading, MA, USA, 1994.
- [40] J. Tsitsiklis. 6.041/6.431 Probabilistic Systems Analysis and Applied Probability, fall 2010. Lecture notes, Massachusetts Institute of Technology: MIT OpenCourseWare, <http://ocw.mit.edu> (Accessed November 4,2014). License: Creative Commons BY-NC-SA.
- [41] F. Vega-Redondo. *Complex Social Networks*. Econometric Society Monographs. Cambridge University Press, 2007.
- [42] D. J. Watts. A simple model of global cascades on random networks. *Proceedings of the National Academy of Sciences*, 99(9):5766–5771, 2002.